GLOBAL EDITION

SIXTH EDITION

Designing the User Interface

Strategies for Effective Human-Computer Interaction

Shneiderman • Plaisant • Cohen • Jacobs • Elmqvist

Pearson

DESIGNING THE USER INTERFACE

SHNEIDERMAN • PLAISANT • COHEN • JACOBS • ELMQVIST

SIXTH EDITION
GLOBAL EDITION

DESIGNING THE USER INTERFACE

STRATEGIES FOR EFFECTIVE HUMAN-COMPUTER INTERACTION

Pearson

Boston Columbus Indianapolis New York San Francisco Hoboken
Amsterdam Cape Town Dubai London Madrid Milan Munich Paris Montréal Toronto
Delhi Mexico City São Paulo Sydney Hong Kong Seoul Singapore Taipei Tokyo

Vice President, Editorial Director, ECS: Marcia Horton
Executive Editor: Tracy Johnson
Editorial Assistant: Kristy Alaura
Vice President of Marketing: Christy Lesko
Director of Field Marketing: Tim Galligan
Product Marketing Manager: Bram Van Kempen
Field Marketing Manager: Demetrius Hall
Marketing Assistant: Jon Bryant
Director of Product Management: Erin Gregg
Team Lead, Program and Project Management: Scott Disanno
Program Manager: Camille Trentacosta and Carole Snyder
Project Manager: Katrina Ostler, Ostler Editorial, Inc.
Senior Specialist, Program Planning and Support: Maura Zaldivar-Garcia
Acquisitions Editor, Global Edition: Sourabh Maheshwari
Senior Project Editor, Global Edition: Amrita Naskar
Senior Media Editor, Global Edition: Gargi Banerjee
Senior Manufacturing Controller, Production, Global Edition: Trudy Kimber
Cover Designer: Black Horse Designs and Lumina Datamatics Ltd.
Chapter Opener Designer: Liese Zahabi
Chapter Opener Photos: Jason Cohen, Jon Froehlich, Kotaro Hara, Catherine Plaisant, Ben Shneiderman, and Liese Zahabi
Manager, Rights and Permissions: Ben Ferrini
Project Manager, Rights and Permissions: Tamara Efsen, Aptara Corporation
Inventory Manager: Ann Lam
Cover Images: Ratchanida Thippayos/123RF; Monty Rakusen/Cultura/Getty Images; Monkey Business/Fotolia; RooM the Agency/ Alamy Stock Photo; PhotosIndia.com LLC/ Alamy Stock Photo
Media Project Manager: Steve Wright

Pearson Education Limited
Edinburgh Gate
Harlow
Essex CM20 2JE
England

and Associated Companies throughout the world

Visit us on the World Wide Web at:
www.pearsonglobaleditions.com

ISBN 10: 1-292-15391-1
ISBN 13: 978-1-292-15391-9

British Library Cataloguing-in-Publication Data
A catalogue record for this book is available from the British Library

10 9 8 7 6 5 4 3 2 1

Typeset by Cenveo® Publisher Services

Printed and bound by Vivar in Malaysia

To Jenny, Anna, and Sara
Ben

To John, Thomas, Grace, and Peter
Catherine

To Arnold, Jason, Rochelle, Joe, and Sara
Maxine

To Sheila, Heather, Lindsey, and Andrew
Steve

To Helene, Keegan, and Corrie
Niklas

To Risa and Minny
Nick

This page intentionally left blank

Preface

Designing the User Interface is written for students, researchers, designers, managers, and evaluators of interactive systems. It presents a broad survey of how to develop high-quality user interfaces for interactive systems. Readers with backgrounds in computer science, engineering, information science/studies/systems, business, psychology, sociology, education, and communications should all find fresh and valuable material. Our goals are to encourage greater attention to user experience design issues and to promote further scientific study of human-computer interaction, including the huge topic of social media participation.

Since the publication of the first five editions of this book in 1986, 1992, 1998, 2005, and 2010, HCI practitioners and researchers have grown more numerous and influential. The quality of interfaces has improved greatly, while the community of users and its diversity have grown dramatically. Researchers and designers deserve as much recognition as the Moore's Law community for bringing the benefits of information and communications technologies to more than 6 billion people. In addition to desktop computers, designers now must accommodate web-based services and a diverse set of mobile devices. User-interface and experience designers are moving in new directions. Some innovators provoke us with virtual and augmented realities, whereas others offer alluring scenarios for ubiquitous computing, embedded devices, and tangible user interfaces.

These innovations are important, but much work remains to be done to improve the experiences of novice and expert users who still struggle with too many frustrations. These problems must be resolved if we are to achieve the goal of universal usability, enabling all citizens in every country to enjoy the benefits of these new technologies. This book is meant to inspire students, guide designers, and provoke researchers to seek those solutions.

Keeping up with the innovations in human-computer interaction is a demanding task, and requests for an update begin arriving soon after the publication of each edition. The expansion of the field led the single author of the first three editions, Ben Shneiderman, to turn to Catherine Plaisant, a long-time valued research partner, for coauthoring help with the fourth and fifth editions. In addition, two contributing authors lent their able support to the fifth edition: Maxine S. Cohen and Steven M. Jacobs have long experience teaching with earlier editions of the book and provided fresh perspectives that improved the quality for all readers and instructors. In preparing for this sixth edition, the team expanded again to include Niklas Elmqvist and Nick Diakopoulos, who are both new colleagues at the University of Maryland. We harvested information from books and journals, searched the World Wide Web, attended conferences, and consulted with colleagues. Then we returned to our keyboards to write, producing first drafts that served as a starting point to generate feedback from each other as well as external colleagues, HCI practitioners, and students. The work that went into the final product was intense but satisfying. We hope you, the readers, will put these ideas to good use and produce more innovations for us to report in future editions.

New in the Sixth Edition

Readers will see the dynamism of human-computer interaction reflected in the substantial changes to this sixth edition. The good news is that most universities now offer courses in this area, and some require it in computer science, information schools, or other disciplines. Courses and degree programs in human-computer interaction, human-centered computing, user experience design, and others are a growing worldwide phenomenon at every educational level. Although many usability practitioners must still fight to be heard, corporate and government commitments to usability engineering grow stronger daily. The business case for usability has been made repeatedly, and dedicated websites describe numerous projects demonstrating strong return on investment for usability efforts.

Comments from instructors who used the previous editions were influential in our revisions. The main changes were (1) to include more on design methods with case study examples and (2) to totally revise our coverage of social media participation and user-generated content, especially from mobile devices. We made major revisions to every chapter, changing almost every figure and substantially updating the references.

The first chapter more boldly recognizes the success story of HCI and user experience design. The growing issue of universal usability for increasingly diverse users of interactive systems became a separate chapter. The next chapters present design guidelines, principles, and theories that have been substantially updated to reflect new ways of thinking. Part 2 covers refinements to development methodologies and evaluation techniques. Part 3 explores progress in direct manipulation and its extensions such as virtual and augmented reality as well as changes to menus, form fill-in, and command languages brought about by the new platforms (especially mobile devices). Since collaboration and social media participation have become so central, that chapter has been heavily expanded and updated. Part 4 emphasizes high-quality and timely user experiences. The chapter on user manuals has been thoroughly revised to reflect the importance of well-designed documentation and user support in serving the goal of universal usability. Finally, information search and visualization have their own chapters because each of these topics has grown dramatically in importance.

We strive to give balanced presentations on controversial topics such as 3-D, speech, and natural language interfaces. Philosophical controversies such as the degree of human control and the role of animated characters are treated carefully to present fairly the viewpoints that differ from our own. We gave colleagues a chance to comment on these sections and made a special effort to provide a balanced presentation while making our own opinions clear. Readers will have to judge for themselves whether we succeeded.

Instructors wanted guidelines and summary tables; these elements are shown in boxes throughout the book. The Practitioner Summaries and Researcher Agendas remain popular; they have been updated. The references have been expanded and freshened with many new sources and with classic papers still included. We worked hard to select references that were widely available and often web-accessible. Figures, especially those showing screen designs, age quickly, so many new user interfaces are shown. Printing in full color makes these figures valuable as a record of contemporary design styles.

Ways to Use This Book

We hope that students, practitioners, and researchers who read this book will want to keep it on their shelves or their electronic book readers to consult when they are working on new topics or seeking pointers to the literature.

Instructors may choose to assign the full text in the order that we present it or to make selections from it. The opening chapter is a good starting point for most students, the second chapter was written as a strong foundation for understanding the challenges of universal usability, and the third chapter covers basic guidelines, principles, and theories. We think all readers should start with these foundations. From there, instructors may take different paths depending on their disciplines. For example, instructors might emphasize the following chapters, listed by area:

- Computer science: 4, 5, 7, 8, 9, 10, 15, 16
- Psychology and sociology: 5, 9, 10, 11, 12
- Industrial engineering: 4, 5, 11, 13, 16
- Library and information studies: 5, 8, 9, 11, 12, 15, 16
- Business and information systems: 4, 5, 6, 9, 10, 12, 13, 14
- Education technology: 4, 5, 11, 12, 14
- Communication arts and media studies: 4, 5, 7, 11, 12, 13, 14
- Technical writing and graphic design: 4, 5, 6, 12, 14, 16

Companion Website www.pearsonglobaleditions.com/shneiderman

The presence of the World Wide Web has a profound effect on researchers, designers, educators, and students. We want to encourage intense use of the web by members of all these groups, but the volatility of the web is not in harmony with the permanence of printed books. Publishing numerous website addresses in the book would have been risky because changes are made daily, but we included key websites in a box at the end of each chapter. To provide more website addresses and keep them current, we have established a Companion Website for this book. We hope that interested readers will visit the site and send us ideas for improving it.

In addition to pointers to current web resources, a variety of supplemental materials for this text are available at the book's Companion Website. The following are accessible to all readers who register using the prepaid access card in the front of this book:

- Links to hundreds of human-computer interaction resources, examples, and research studies that enhance and expand on the material in each chapter
- Chapter/section summaries
- Self-test questions for each chapter
- Homework assignments and projects

PowerPoint lecture slides are also available from Instructor Resource Center (www.pearsonglobaleditions.com/shneiderman). For information about accessing these instructor's supplements, visit the Instructor Resource Center or get in touch with your Pearson representative.

Acknowledgments

Writing is a lonely process; revising is a social one. We are grateful to the many colleagues and students who have made suggestions for improvements to prior editions. After one two-day kickoff meeting, we collaborated smoothly by using e-mail, Dropbox for sharing drafts, Google Docs for group-edited task lists, and hour-long phone conference calls every one to three weeks. Capable coauthors with cooperative personalities made the hard work for this massive project possible even with tight time constraints. We are grateful to Nick Diakopoulos for writing the chapter on communication and collaboration, which provided a fresh perspective on this vital topic, and for reviewing draft chapters.

We've greatly appreciated the comments on draft chapters from many colleagues, including: Sriram Karthik Badam, Gilles Bailly, Andy Cockburn, Jim Dabrowski, Eck Doerry, Steven Dow, Laura Downey, Pierre Dragicevic, Laurie Dringus, Cody Dunne, Jason Dykes, Massoud Ghyam, Marti Hearst, Harold Henke, Pourang Irani, Jonathan Lazar, Clayton Lewis, Kurt Luther, Ethan Munson, Alan Newell, Whitney Quesenbery, Karthik Ramani, Dan Russell, Helen Sarid, Cees Snoek, Erik Stolterman, Pawel Wozniak, and Adil Yalcin. We also thank all the colleagues and industry contacts who have helped us assemble the 170+ figures of the book.

We appreciate our close daily collaborations with faculty at the University of Maryland Human-Computer Interaction Lab, many of whom provided comments on drafts. This supportive community has a profound influence on our work: June Ahn, Ben Bederson, Marshini Chetty, Allison Druin, Leah Findlater, Jon Froehlich, Jen Golbeck, Kent Norman, Doug Oard, Jennifer Preece, and Jessica Vitak. Thanks also to Liese Zahabi, creator of the chapter-opening graphics, which add an uplifting elegance to our work. We also appreciate the undergraduate and graduate students who provide encouraging feedback and challenging questions plus the motivation to keep updating this book.

The publisher's editorial and production staff were involved in this book from the start. We appreciate the contributions of Kristy Alaura, Scott Disanno, Tracy Johnson, Timothy Nicholls, Carole Snyder, Camille Trentacoste, and Katie Ostler. We apologize if we have left out any other contributors. Finally, further thanks goes to the students and professionals from around the world who have sent us inspiring comments and helpful suggestions.

This book is written for the future students and professionals who will continue the remarkable work in human-computer interaction and user experience design that has helped bring the benefits of information and communications technologies to billions of users.

Ben Shneiderman
Catherine Plaisant
Maxine Cohen
Steven Jacobs
Niklas Elmqvist
Nicholas Diakopoulos

Global Edition Acknowledgments

The publishers would like to thank the following for their contribution to the Global Edition:

Contributor
Wricha Mishra, MAEER's MIT Institute of Design

Reviewers
Yogesh D. Deshpande, Vishwakarma Institute of Information Technology
Ashutosh Mishra, Indian Institute of Information Technology, Allahabad

About the Authors

BEN SHNEIDERMAN (http://www.cs.umd.edu/~ben) is a Distinguished University Professor in the Department of Computer Science, Founding Director (1983-2000) of the Human-Computer Interaction Laboratory (http://www.cs.umd.edu/hcil/), and a Member of the UM Institute for Advanced Computer Studies (UMIACS) at the University of Maryland. He is a Fellow of the AAAS, ACM, IEEE, NAI, and SIGCHI Academy and a Member of the National Academy of Engineering, in recognition of his pioneering contributions to human-computer interaction and information visualization.

CATHERINE PLAISANT (hcil.umd.edu/catherine-plaisant) is a Senior Research Scientist at the University of Maryland Institute for Advanced Computer Studies and Associate Director of Research of the Human-Computer Interaction Lab. Catherine Plaisant earned her Ph.D. in Industrial Engineering at the Université Pierre et Marie Curie - Paris VI, France. She was elected to the ACM SIGCHI Academy in 2015 for her contributions to the field of human-computer interaction, medical informatics and information visualization.

MAXINE COHEN (http://cec.nova.edu/faculty/cohen.html) is a Professor in the College of Engineering and Computing at Nova Southeastern University in Fort Lauderdale, FL. She teaches graduate courses (on campus and online) in Human-Computer Interaction, Interaction Design, and Social Media and advises doctoral students. Previously she worked for IBM (Endicott, NY and Boca Raton, FL) and taught at the Watson School of Engineering at Binghamton University. She has served as a meta-reviewer for ACM Computing Reviews for over 20 years. She earned her Ph.D. and M.S. from Binghamton University and her B.A. from the University of Vermont. She is a member of ACM, IEEE, and UPE.

STEVEN JACOBS (http://cefns.nau.edu/~smj93/) retired from the aerospace industry and is now a lecturer in the School of Informatics, Computing, and Cyber Systems as well as University College Faculty Fellow at Northern Arizona University, Flagstaff, Arizona. He was with Northrop Grumman Information Systems (formerly TRW) in Carson, California for 25 years. He was also Adjunct Assistant Professor at the University of Southern California Department of Computer Science for 17 years, where he developed and taught their graduate courses in user interface design and human performance engineering. He received his M.S.C.S. from UCLA and B.A. in Mathematics from Monmouth University (N.J.). Mr. Jacobs is a Senior Member of ACM.

NIKLAS ELMQVIST (http://sites.umiacs.umd.edu/elm/) is an Associate Professor in the College of Information Studies at University of Maryland, College Park, with affiliate appointments in the Department of Computer Science and the UM Institute for Advanced Computer Studies (UMIACS). He is also a member of the Human-Computer Interaction Laboratory (HCIL). Previously a faculty member at Purdue University, he received his Ph.D. from Chalmers University of Technology in Gothenburg, Sweden. He is a Senior Member of the ACM and the IEEE.

NICHOLAS DIAKOPOULOS (http://www.nickdiakopoulos.com/) is an Assistant Professor at the University of Maryland, College Park Philip Merrill College of Journalism with courtesy appointments in the College of Information Studies and Department of Computer Science. He is a member of the Human-Computer Interaction Lab (HCIL) and is director of the Computational Journalism Lab where he researches algorithmic accountability, narrative data visualization, and social computing in the news. He received his Ph.D. in Computer Science from the School of Interactive Computing at the Georgia Institute of Technology.

Brief Contents

This page intentionally left blank

Contents

DESIGNING THE USER INTERFACE

PART 1

Introduction

PART OUTLINE

This first set of chapters provides a broad introduction to user interface design and interactive systems. Chapter 1 covers usability goals, measures, and motivations as well as general goals for the HCI profession. A rich set of resources is available at the end of the chapter, listing important books, guidelines, and relevant journals and professional organizations.

Chapter 2 discusses universal usability and exposure to the diversity of users. This includes the challenges posted by physical, cognitive, perceptual, personality, and cultural differences. Chapter 3 reviews the guidelines, principles, and theories of the field to help facilitate good design.


GV


score:
estat:
hbond:
vdw:
ds:

CHAPTER 1

Usability of Interactive Systems

> "Designing an object to be simple and clear takes at least twice as long as the usual way. It requires concentration at the outset on how a clear and simple system would work, followed by the steps required to make it come out that way—steps which are often much harder and more complex than the ordinary ones. It also requires relentless pursuit of that simplicity even when obstacles appear which would seem to stand in the way of that simplicity."
>
> **T. H. Nelson**
> *The Home Computer Revolution, 1977*

CHAPTER OUTLINE

1.1 Introduction

User-interface designers are the heroes of a profound transformation. Their work turned personal computers into today's wildly successful mobile devices, enabling users to communicate and collaborate in remarkable ways. The desktop applications that once served the needs of professionals have increasingly given way to powerful social tools that deliver compelling user experiences to global communities. These invigorated communities conduct business, communicate with family, get medical advice, and create user-generated content that can be shared with billions of connected users.

These life-changing shifts were made possible because researchers and user-interface designers harnessed technology to serve human needs. Researchers created the interdisciplinary design science of *human-computer interaction* by applying the methods of experimental psychology to the powerful tools of computer science. Then they integrated lessons from educational and industrial psychologists, instructional and graphic designers, technical writers, experts in human factors or ergonomics, and growing teams of anthropologists and sociologists. As the impact of these mobile social tools and services spreads, researchers and designers are gathering still fresher insights from sustainability activists, consumer advocates, citizen scientists, and humanitarian disaster response teams.

User experience designers produce business success stories, Hollywood heroes, and Wall Street sensations. They also produce intense competition, copyright-infringement suits, intellectual-property battles, mega-mergers, and international partnerships. Crusading Internet visionaries, like Google's Eric Schmidt, promote a world with free access to information and entertainment, while equally devoted protectors of creative artists, like singer Taylor Swift, argue for fair payments. User interfaces are also controversial because of their central role in personal identification, national defense, crime fighting, electronic health records, and so on.

At an individual level, effective user experiences change people's lives: Doctors can make more accurate diagnoses, and pilots can fly airplanes more safely; at the same time, children can learn more effectively, users with disabilities can lead more productive lives, and graphic artists can explore more creative possibilities. Some changes, however, are disruptive, reducing the need for telephone operators, typesetters, and travel agents. Too often, users must cope with frustration, fear, and failure when they encounter excessively complex menus, incomprehensible terminology, or chaotic navigation paths.

At a societal level, connected communities open up new forms of collective action and policy engagement. Having more informed citizens may lead to better decisions, more transparent governance, and greater equity when facing legal, health, or civic challenges. But there may be increased dangers from extreme groups who promote terrorism, oppressive social policies, or racial hatred. The increased power of social media and collaboration technologies means that there must be a new balance of legal protections, police powers, and privacy.

The steadily growing interest in human-computer interaction stems from the designers' desire to improve the users' experience (Figs. 1.1 to 1.3 show some popular applications). In business settings, better decision-support and document-sharing tools support entrepreneurs, while in-home settings, digital photo libraries, and internet conferencing enhance family and personal relationships. Millions of people take advantage of the World Wide Web's extraordinary educational and cultural heritage resources, which provide access to everything from outstanding art objects from China to music from Indonesia, sports from Brazil, and entertainment from Hollywood or Bollywood

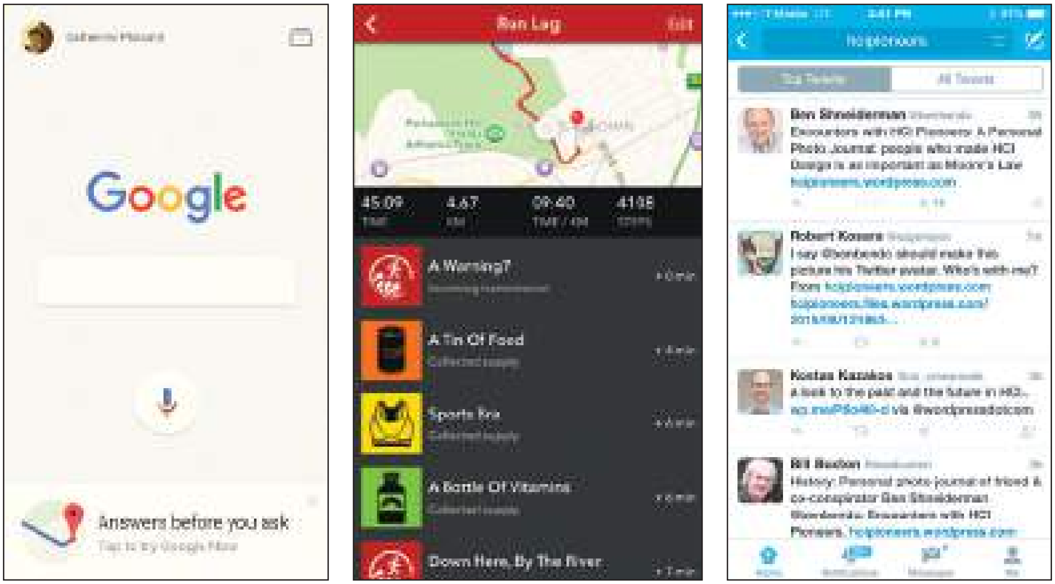


FIGURE 1.1

Smartphones have high-quality displays, provide fast Internet connections, include many sensors, and support a huge variety of applications.
Left: Google Now for searching, reviewing notification cards, and speaking commands.
Center: Zombies, Run! is an immersive running game and audio adventure which encourages runners to run as if pursued by zoombies, and to collect goods to help their community survive.
Right: A Twitter feed lists the top tweets after Ben Shneiderman announced the release of the HCI Pioneers website.

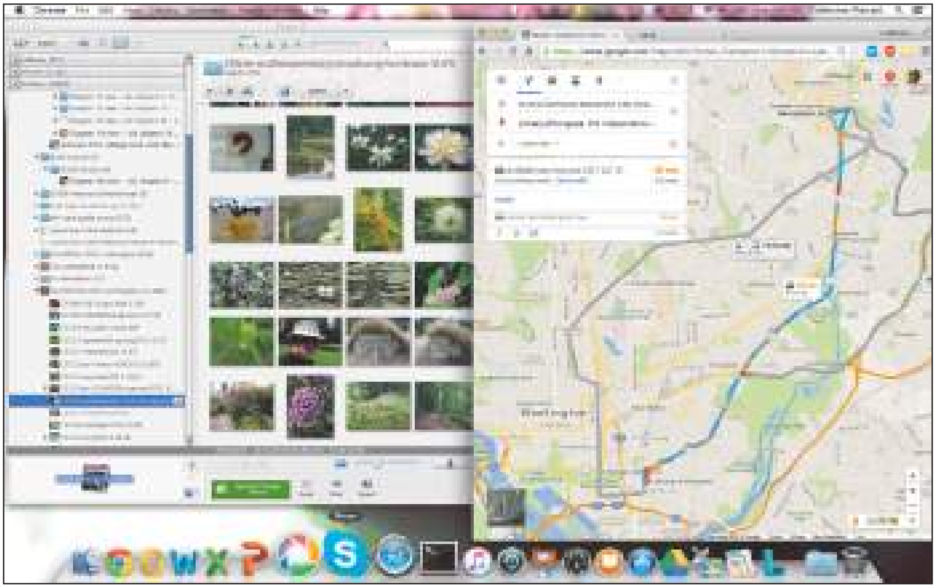


FIGURE 1.2

Apple® Mac OS X®. showing Picasa for photo browsing and Google Maps in a web browser. The bottom of the screen also shows the Dock, a menu of frequently accessed items whose icons grow larger on mouse-over.

(Figs. 1.4 to 1.5 show examples of popular websites). Mobile devices enrich daily life for many users, including those with disabilities, limited literacy, and low incomes. On a worldwide scale, promoters and opponents of globalization debate the role of technology in international development, while activists work to attain the United Nations Sustainable Development Goals.

FIGURE 1.3

Ben Shneiderman at a standing desk with two high-resolution screens. We can see a MS Word document (with six pages visible), two web browsers, and the Outlook e-mail application in a Windows environment.

The remarkably rapid and widespread adoption of mobile devices (including smartphones, tablets, game devices, fitness trackers, etc.) supports personal communication, collaboration, and content creation. The proliferation of such devices in developed as well as developing

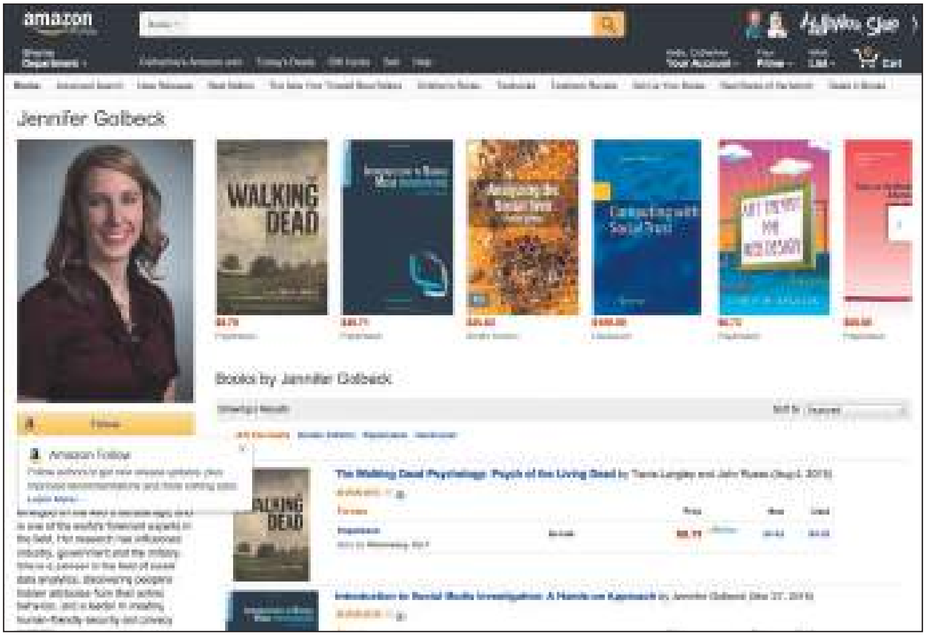


FIGURE 1.4
The Amazon.com website (http://www.amazon.com/) showing the books published by Jen Golbeck. Facebook will make book and product recommendations based on a user's personal history with the site.

nations has been astonishing. Economists see a direct linkage between cell-phone dissemination and economic growth since communications facilitate e-commerce and stimulate entrepreneurial ventures. Mobile devices also promote wellness, enable timely medical care, and provide life-saving disaster response services.

Similarly, explosive growth is the appropriate description for what's happening in the realms of social networking and user-generated content. Older media, such as newspapers and television, have lost audiences in favor of social media such as Facebook, Twitter, YouTube, and Wikipedia (all of which are among the top 10 most visited services). These leading websites are just a taste of what is to come, as entrepreneurs trigger ever more social media involvement accessible through web-based applications and small mobile devices.

Designers enable users to create, edit, and distribute 3-D printed objects, immersive virtual reality games, interactive animations, and increasingly high-definition music, voice, and videos. The result is ever-richer experiences and a creative outpouring of user-generated content available, even on mobile devices.

FIGURE 1.5

YouTube showing a video showing NASA TV and other available related videos on the side. The NASA video shows an example of a control center with multiple large wall displays and workstations.

Sociologists, anthropologists, policymakers, and managers are studying how social media are changing education, family life, shopping, and services such as medical care, financial advice, and political organizations. They are also dealing with issues of organizational impact, job redesign, distributed teamwork, work-at-home scenarios, and long-term societal changes. As face-to-face interaction gives way to screen-to-screen, how can personal trust and organizational loyalty be preserved? How can empathy be conveyed and civic participation be enhanced?

Designers face the challenge of providing services on small-, wall-, and mall-sized displays, ranging from jewelry, clothing (Fig. 1.6), smartphones, and tablets (Fig. 1.7) to large panels, projected displays, and illuminated buildings. When the *plasticity* of their designs provides smooth conversion across different display sizes, consumers take pleasure; when conversions are difficult, consumers take notice. But the malleability of user interfaces has to extend to translation into multiple languages, accessibility support for users with disabilities, and accommodation for varying network bandwidths.

Some innovators promise that desktop computers and their user interfaces will disappear, as new interfaces become ubiquitous, pervasive, invisible, and embedded in the surrounding environment. They believe that novel devices will be context-aware, attentive, and perceptive, sensing users' needs and providing feedback through ambient displays that glow, hum, change shape, or blow air.

FIGURE 1.6
Two children learn about the human body using a wearable, e-textile shirt displaying real-time visualizations of how the body works via "organs" with embedded LED lights and sound (Norooz et al., 2015).

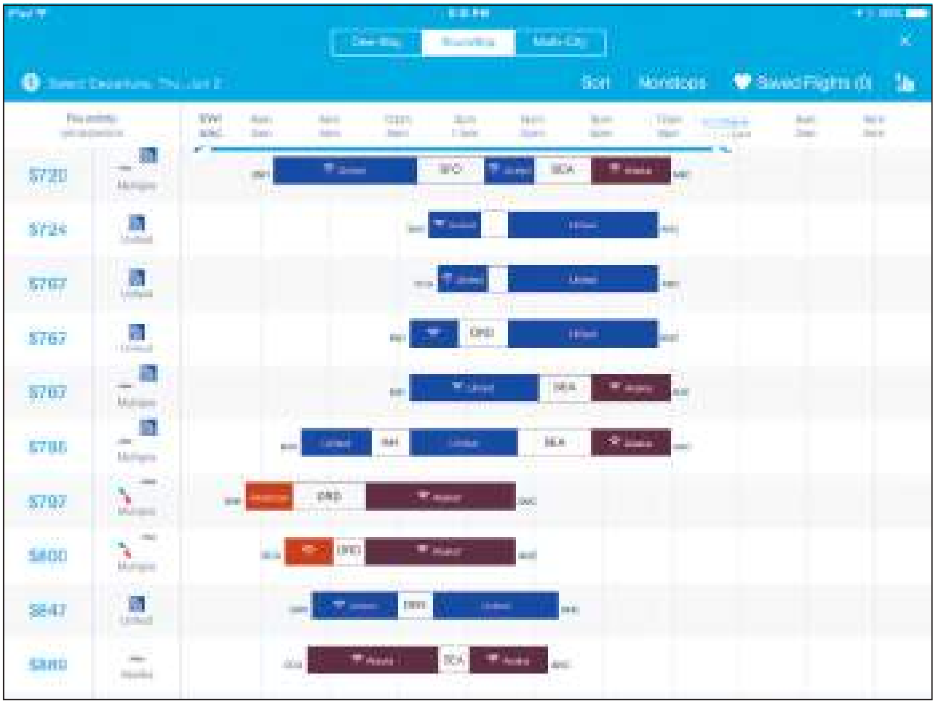


FIGURE 1.7
The HIPMUNK travel search shows available flights visually as seen on a Apple iPad tablet. The slider at the top allows users to narrow down the results. Here we see only the flights landing before 10:25 P.M.

Designers are already offering interfaces that are wearable or control implanted (under-the-skin) devices, such as pacemakers, insulin pumps, and varied biomonitors. Other kinds of sensors already track FedEx packages, users entering buildings, or cars at tollbooths, but they will expand into elaborate sensor nets that follow crowds, epidemics, and pollution.

Other designers promote persuasive technologies that change users' behavior, multi-modal or gestural interfaces that facilitate use, and affective interfaces that respond to the user's emotional state.

We are living in an exciting time for designers of user interfaces. The inspirational pronouncements from technology prophets can be thrilling, but rapid progress is more likely to come from those who do the hard work of tuning designs to genuine human needs. These designers will rigorously evaluate actual use with eager early adopters, as well as reluctant late adopters, and seriously study the resistant non-users. This book's authors believe that the next phase of human-computer interaction will be strongly influenced by those who are devoted to broadening the community of users by promoting universal usability and facilitating many forms of social media participation. User interfaces that deliver excellent user experiences will be a key component in improving healthcare, creating sustainable economies, protecting natural resources, and resolving conflicts (Froehlich et al., 2010; Friedman et al., 2014).

This first chapter gives a broad overview of human-computer interaction from practitioners' and researchers' perspectives. It lays out usability goals, measures, and motivations in Sections 1.2 and 1.3 and closes with a statement of goals for our profession. Specific references cited in the chapter appear at the end, followed by a set of general references. Lists of relevant books, guidelines documents, journals, professional organizations, and video collections give readers starting points for further study.

The second chapter takes on universal usability, reminding readers of the opportunities to reach diverse users with tailored materials that serve the needs of young and old, high and low literacy users, diverse international users, and users with varying disabilities.

The third chapter reviews the guidelines, principles, and theories that will be drawn on and refined throughout the book. Chapters 4–6 introduce design processes and evaluation methods, with case study examples to demonstrate the processes and methods. Chapters 7–9 cover interaction styles that range from graphical direct manipulation to speech control and their implementation using common interaction devices. Collaboration is included in this part to emphasize the need for every designer to go beyond the personal computer and consider the many forms of social computing. Chapters 10–16 address the critical design decisions that often determine the success or failure of products and that may lead to breakthroughs that open the way to new possibilities. The Afterword reflects on the societal and individual impacts of technology.

1.2 Usability Goals and Measures

Every designer wants to develop high-quality user experiences that are admired by colleagues, celebrated by users, and imitated by competitors. But getting such attention takes more than flamboyant promises and stylish advertising; it's earned by providing quality features such as usability, universality, and usefulness. These goals are achieved by thoughtful planning, sensitivity to user needs, devotion to requirements analysis, and diligent testing, all while keeping within budget and on schedule.

Managers who pursue user-interface excellence first select experienced designers and then prepare realistic schedules that include time for requirements gathering, guidelines preparation, and repeated testing. The designers begin by determining user needs, generating multiple design alternatives, and conducting extensive evaluations (Chapters 4–6). Modern user-interface-building tools then enable implementers to quickly build working systems for further testing.

Successful designers go beyond vague notions of "user friendliness," "intuitive," and "natural," doing more than simply making checklists of subjective guidelines. They have a thorough understanding of the diverse community of users and the tasks that must be accomplished. They study evidence-based guidelines and pursue the research literature when necessary. Great designers are deeply committed to enhancing the user experience, which strengthens their resolve when they face difficult choices, time pressures, and tight budgets. Great designers are also aware of the importance of eliciting emotional responses, attracting attention with animations, and playfully surprising users.

When managers and designers have done their jobs well, their interfaces generate positive feelings of success, competence, and mastery among users. The users have a clear mental model of the interface that enables them to confidently predict what will happen in response to their actions. In the best cases, the interface almost disappears, enabling users to concentrate on their work, exploration, or pleasure. This kind of calming environment gives users the feeling that they are "in the flow," operating at their peak, while attaining their goals.

Close interaction with the user community leads to a well-chosen set of benchmark tasks that is the basis for usability goals and measures. For each user type and each task, precise measurable objectives guide the designer through the testing process. The ISO 9241 standard *Ergonomics of Human-System Interaction* (ISO, 2013) focuses on admirable goals—*effectiveness*, *efficiency*, and *satisfaction*—but the following usability measures, which focus on the latter two goals, lead more directly to practical evaluation:

1. *Time to learn.* How long does it take for typical members of the user community to learn how to use the actions relevant to a set of tasks?

2. *Speed of performance.* How long does it take to carry out the benchmark tasks?
3. *Rate of errors by users.* How many and what kinds of errors do people make in carrying out the benchmark tasks? Although time to make and correct errors might be incorporated into the speed of performance, error handling is such a critical component of interface usage that it deserves extensive study.
4. *Retention over time.* How well do users maintain their knowledge after an hour, a day, or a week? Retention may be linked closely to time to learn, and frequency of use plays an important role.
5. *Subjective satisfaction.* How much did users like using various aspects of the interface? The answer can be ascertained by interviews or by written surveys that include satisfaction scales and space for free-form comments.

Every designer would like to succeed in every measure, but there are often forced tradeoffs. If lengthy learning is permitted, task-performance times may be reduced by use of abbreviations, hidden shortcuts, and compact designs that minimize scrolling. If the rate of errors is to be kept extremely low, speed of performance may have to be sacrificed. In some applications, subjective satisfaction may be the key determinant of success; in others, short learning times or rapid performance may be paramount. Project managers and designers who are aware of the tradeoffs can be more effective if they make their choices explicit and public. Requirements documents and marketing brochures that make clear which goals are primary are more likely to be valued.

After multiple design alternatives have been raised, the leading possibilities should be reviewed by designers and users. Low-fidelity paper mockups are useful, but high-fidelity interactive prototypes create a more realistic environment for expert reviews and usability testing. The user training and supporting materials such as online help can be produced before the implementation to provide another review and a new perspective on the design. Next, the implementation can be carried out with proper software tools; this task should be a modest one if the design is complete and precise. Then, acceptance testing certifies that the delivered interface meets the goals of the designers and customers. Finally, continuous evaluation and improvement have become common practices. These design processes, evaluation procedures, and software tools are described more fully in Chapters 4–6.

The business case for usability is strong and has been made repeatedly (Bias and Mayhew, 2005; Tullis and Albert, 2013). User-interface design success stories can also be managerial success stories for projects that are on budget and on schedule. A thoroughly documented set of user needs clarifies the design process, and a carefully tested prototype generates fewer changes during implementation while avoiding costly updates after release. Thorough acceptance testing of the implementation produces robust interfaces that are aligned with user needs. Then continuous evaluation based on usage logs and user comments guide evolutionary refinements.

1.3 Usability Motivations

The enormous interest in interface usability arises from the demonstration of the benefits that come from well-designed user interfaces. This increased motivation emanates from designers and managers of consumer electronics who produce mobile devices, e-commerce websites, and social media where excellent user experiences are necessary to succeed in large, highly competitive markets. Usability has gone from desirable to necessary for survival. Similarly, the huge interest in games and entertainment has raised the performance of devices, networks, and user interfaces. The goals are to ensure that game playing is fluid and vivid; that photo, music, and video streaming is fast; and that sharing is graceful and simple. Strong motivations for usability quality come from high-functioning professionals who demand excellence in environments such as life-critical systems, industrial plants, legal offices, and police agencies. The spirit of usability excellence is also expected by users of exploratory, creative, and collaborative interfaces as well as diverse sociotechnical systems.

1.3.1 Consumer electronics, e-commerce, and social media

User experience designers have played a key role in the dramatic growth of consumer electronics by providing effective and satisfying designs that have become widely adopted for personal communications, education, healthcare, and much more. The annual Consumer Electronics Show, now replicated in many locations around the world, brings tens of thousands of exhibitors and hundreds of thousands of attendees who are eager to try the latest products from leading vendors.

Product announcements trigger worldwide media coverage, with Hollywood or sports personalities celebrating the newest products. Similarly, famed musicians, supermodels, and other luminaries contribute to the media hype while making everyone aware of the latest designs, appealing features, and must-have capabilities. Heroes such as Apple's Chief Design Officer Jony Ive have become celebrities who are knighted by the Queen of England and pestered by interviewers to reveal the secrets of the next product release.

The transformative power of consumer electronics has been celebrated by those who see improved family communication, better healthcare, thriving businesses, and wider access to education. The social media applications, dominated by Facebook, and user-generated content such as online restaurant, film, or product reviews have become part of daily life for many users. For these interfaces, ease of learning, low error rates, and subjective satisfaction are paramount because use is discretionary and competition is fierce. If the users cannot succeed quickly, they will give up or try a competing supplier. Critics raise concerns about reduced privacy, dangers in distracted driving, and declining quality of interpersonal relationships.

1.3.2 Games and entertainment

The rapid expansion of home and entertainment applications is a further source of interest in usability. Personal-computing applications include e-mail clients, search engines, cellphones, digital cameras, and music players. Entertainment applications have flourished, making computer games a larger industry than Hollywood, while game input devices like the Nintendo® Wii™ and *the Microsoft Kinect's*™ controller-free gameplay (Fig. 1.8) open up entirely new possibilities in areas ranging from sports to education to rehabilitation.

Choosing the right functionality while keeping costs low is difficult. Novices are best served by a constrained, simple set of actions, but as users' experience increases, so does their desire for more extensive functionality and rapid performance. A layered or level-structured design is one approach to facilitating graceful evolution from novice to expert usage: Users can move up to higher layers when they need additional features or have time to learn them. A simple

FIGURE 1.8

Dance Central, a highly successful dance-playing franchise of games in which users dance to popular songs and earn points for how well they keep up. The Dance Central website allows users to purchase additional songs and also hosts livestream events and community forums.

example is the design of search engines, which almost always have basic and advanced interfaces (Chapter 15). Another approach to winning novice users is to carefully trim the features to make a simple device or application so users can get started easily.

1.3.3 Professional environments

Most consumer electronics users also benefit from interfaces in professional environments from supermarkets to space stations. Life-critical systems include those that control air traffic, nuclear reactors, power utilities, police or fire dispatch, military operations, and clinical care (Fig. 1.9). In these applications, high costs are expected, but they should yield high reliability and effectiveness. Lengthy training periods are acceptable to obtain rapid, error-free performance, even when the users are under stress. Subjective satisfaction is less of an issue because the users are well-motivated professionals. Retention is obtained by frequent use of common functions and practice sessions for emergency actions.

Typical industrial and commercial uses include interfaces for banking, insurance, production management, airline and hotel reservations, utility billing, and point-of-sale terminals. In these cases, costs shape many judgments. Operator training time is expensive, so ease of learning is important. Since many businesses are international, translation to multiple languages and adaptations to local cultures are necessary. The tradeoffs between speed of performance and error rates are governed by the total cost over the system's lifetime (Chapter 12). Subjective satisfaction is of modest importance; retention is obtained by frequent use. Speed of performance is central for most of these applications because of the high volume of transactions, but operator fatigue, stress, and burnout are legitimate concerns. Trimming 10% off the mean transaction time could mean 10% fewer operators, 10% fewer workstations, and a 10% reduction in hardware costs.

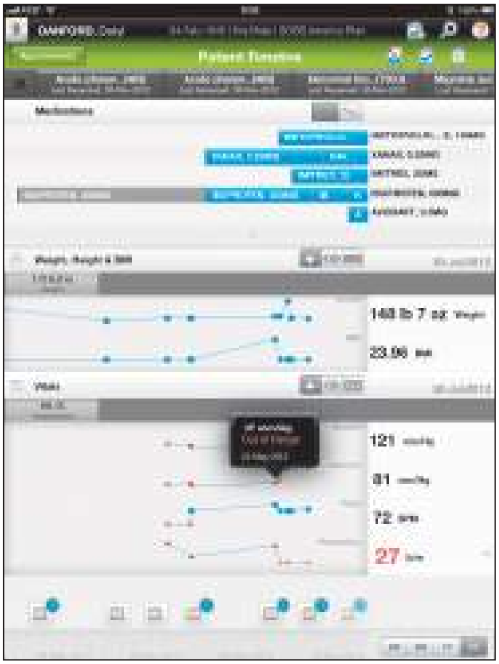


FIGURE 1.9

The Wand timeline view of a patient record in Allscript's ambulatory Electronic Health Record iPad application.

1.3.4 Exploratory, creative, and collaborative interfaces

An increasing fraction of computer use is dedicated to supporting open-ended exploration that promotes human creativity while lowering barriers to collaboration. Exploratory applications include web browsers, search engines, data visualization, and team collaboration support. Creative applications include design environments (Fig. 1.10), music-composition tools, animation builders, and video-editing systems. Collaborative interfaces enable two or more people to work together (even if the users are separated by time and space) through use of text, voice, and video; through systems that facilitate face-to-face meetings; through large audience participation in webinars; or through sharing tools that enable remote collaborators to work concurrently on a document, map, calendar, or image.

In these exploratory, creative, and collaborative environments, the users may be knowledgeable in the task domains but novices in the underlying computer concepts. Their motivation is often high, but so are their expectations. Benchmark tasks are more difficult to describe because of the exploratory nature of these applications, and usage can range from occasional to frequent. In short, it is difficult to design and evaluate these systems. Designers can pursue the goal of having the

FIGURE 1.10

Sketchbook™, a design tool for digital artists, from Autodesk™. A large number of tools and options are available through a rich set of menus and tool palettes (http://www.sketchbook.com).

computer "vanish" as users become completely absorbed in their task domains. This goal seems to be met most effectively when the computer provides a direct-manipulation representation of the world of action (Chapter 7), supplemented by keyboard shortcuts. Then tasks are carried out by rapid familiar selections or gestures with immediate feedback and new sets of choices. Users can keep their focus on the task with minimal distraction caused by operating the interface.

1.3.5 Sociotechnical systems

A growing domain for usability is in social systems that involve many people over long time periods, such as healthcare, citizen science, disaster response, and community crime reporting. Interfaces for these systems, often created by governmental organizations, have to deal with trust, privacy, and responsibility as well as limiting the harmful effects of malicious tampering, deception, and incorrect information. Users will want to know whom to turn to when things go wrong—and maybe whom to thank when things go right (Whitworth and de Moor, 2009).

For example, in electronic voting systems (Jones and Simons, 2012), citizens need to have reassuring feedback that their votes are correctly recorded, possibly by having a printed receipt. In addition, government officials and professional observers from opposing parties need to have ways of verifying that the votes from each district and regional aggregations are correctly reported (Fig. 1.11). If complaints are registered, investigators need tools to review procedures at every stage.

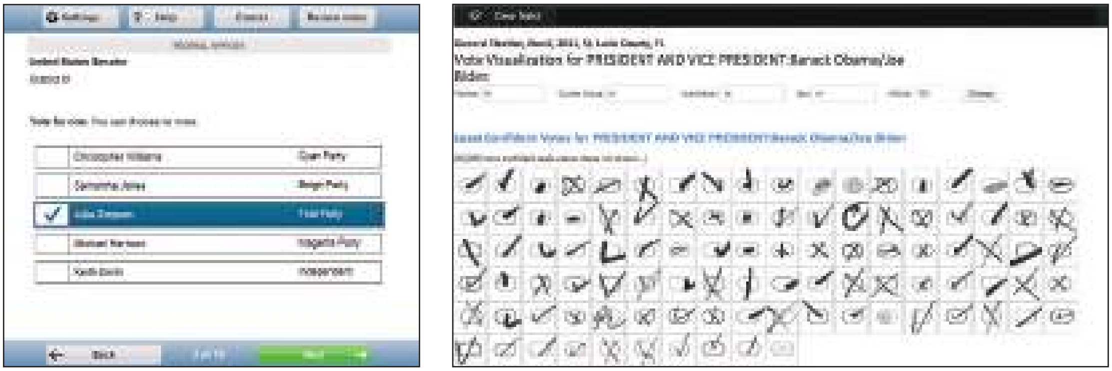


FIGURE 1.11

On the left we see an example of a touchscreen voting kiosk interface (Summers et al., 2014). We see contest number 2 out of 10 and the five candidates. The selected candidate is clearly marked. Some voting jurisdictions use paper ballots that are then digitized. The interface on the right allows rapid review of all the handwritten marks. Courtesy of Clear Ballot (http://www.clearballot.com).

Designers of sociotechnical systems have to take into consideration the diverse levels of expertise of users with different roles. Successful designs for the large number of novice and first-time users emphasize ease of learning and provide the feedback that builds trust. Designs for professional administrators and seasoned investigators enable rapid performance of complex procedures, perhaps with visualization tools to spot unusual patterns or detect fraud in usage logs.

1.4 Goals for Our Profession

Clear goals are useful not only for interface design but also for educational and professional enterprises. Three broad goals seem attainable: (1) influencing academic and business researchers; (2) providing tools, techniques, and knowledge for commercial designers; and (3) raising the user-interface consciousness of the general public.

1.4.1 Influencing academic and business researchers

Researchers in human-computer interaction are prolific as they produce more than 10,000 papers per year. Their research include traditional controlled experimentation in laboratory settings, but increasingly researchers conduct online testing with real users, ethnographic observations in users' homes or workplaces, and long-term, in-depth case studies of users (Chapter 5).

Newer research methods include crowd-sourced user studies that invite thousands of users to participate or pay users through systems such as Amazon's Mechanical Turk. Another innovation is the use of user log data, observations, and interviews to provide complementary strategies that reveal actual performance in live settings. The combination of methods often leads to a deeper understanding of the fundamental principles of human interaction with computers.

The classic scientific method for interface research, which is based on controlled experimentation, has this basic outline:

- Understanding of a practical problem and related theory
- Lucid statement of a testable hypothesis
- Manipulation of a small number of independent variables
- Measurement of specific dependent variables
- Careful selection and assignment of subjects
- Control for bias in subjects, procedures, and materials
- Application of statistical tests
- Interpretation of results, refinement of theory, and guidance for experimenters

When experimental materials and methods are tested by pilot studies and results validated by replication in various situations, then the recommendations are more likely to be reliable.

Of course, the scientific method based on controlled experimentation has its weaknesses. It may be difficult or expensive to find adequate subjects, and laboratory conditions may distort the situation so much that the conclusions have little value. Controlled experiments typically deal with short-term usage, so understanding long-term consumer behavior or experienced user strategies is difficult. Since controlled experiments emphasize statistical aggregation, extremely good or poor performance by individuals may be overlooked. Furthermore, anecdotal evidence or researcher insights may be given too little emphasis because of the authoritative influence of statistics.

Because of these concerns, researchers balance controlled experimentation with ethnographic observation methods and long-term, in-depth case studies. Anecdotal experiences and subjective reactions are recorded, think-aloud approaches are employed, and field or case studies can be carried out. Other research methods include crowd-sourced user studies, analysis of user logs, surveys, focus groups, and interviews.

Within computer science and information studies, there is a growing awareness of the need for greater attention to usability issues. Courses on human-computer interaction are required for some undergraduate degrees, and interface design issues are being added to many curricula. Researchers who propose new programming languages, privacy-protection schemes, or network services are more aware of the need to align with human cognitive skills and preferences. Designers of advanced graphics systems, 3-D printing tools, or consumer products increasingly recognize that their success depends on the construction of effective user interfaces and creation of appealing user experiences.

There is a grand opportunity to apply the knowledge and techniques of traditional psychology (and of subfields such as cognitive and social psychology) to the study of human-computer interaction. Psychologists are investigating human problem solving and creativity with user interfaces to gain an understanding of cognitive processes and social dynamics. The benefit to psychology is great, but psychologists also have a golden opportunity to dramatically influence an important and widely used technology. Similarly, sociologists and communications theorists are now actively participating in human-computer interaction research.

Researchers in business, management, education, sociology, anthropology, and other disciplines are benefiting from and contributing to the study of human-computer interaction. There are many fruitful directions for research, but here are a few:

- *Reduced anxiety and fear of computer usage.* Although computers are widely used, some otherwise competent people resist using e-mail and engaging in e-commerce because they are anxious about—or even fearful of—breaking the device, making an embarrassing mistake, or having their privacy violated. Fear of scams and frustration with e-mail spam could also be reduced by improved designs that promote security and privacy while increasing the users' control over their experiences.

- *Graceful evolution.* Although novices may begin their interactions with a computer by using just a few features, they may later wish to move up to more powerful facilities. Refined multi-layer interface designs, preference settings, and training materials are needed to smooth the transition from novice to knowledgeable user to expert. The differing requirements of novices and experts in terms of prompting, error messages, online assistance, display complexity, pacing, and informative feedback all need investigation. Users may be allowed to customize their interfaces far beyond changing backgrounds, font sizes, and ring tones, but methods for guiding users through such processes are an open topic.
- *Social media.* The remarkable spread of social media is an indicator of larger changes to come. Enabling sharing of user-generated content, especially from mobile devices, is widespread; much work remains to be done in raising the quality of what is produced, enabling effective annotations, making these materials accessible, and facilitating reuse in ways that protect users' desires for privacy or profit.
- *Input devices.* The plethora of input devices presents opportunities and challenges to interface designers (Chapter 10). There are heated discussions about the relative merits of multi-touch screens, voice, gestures, and haptic feedback. Such conflicts could be resolved through experimentation with multiple tasks and users. Underlying issues include speed, accuracy, fatigue, error correction, and subjective satisfaction.
- *Information exploration.* As navigation, browsing, and searching in multimedia digital libraries and the World Wide Web become more common, the pressure for more effective strategies and tools has increased (Chapter 15). Users will want to filter, select, and restructure their information rapidly with minimum effort and without fear of getting lost or finding misleading information. Large databases of text, images, graphics, sound, video, and scientific data, commonly called *big data*, are becoming easier to explore with information visualization and visual analytic tools.

1.4.2 Providing tools, techniques, and knowledge for commercial designers

User-interface design and development are hot topics, and international competition is lively. Employers who used to see usability as a secondary topic are increasingly hiring user experience designers, information architects, mobile app implementers, and usability testers. These employers recognize the competitive advantage from high-quality consumer interfaces and from improving the performance of their employees. There is a great thirst for knowledge about software tools, design guidelines, and testing techniques. User-interface–building tools provide support for rapid prototyping and interface development while aiding design consistency, supporting universal usability, and simplifying evolutionary refinement.

Guidelines documents have been written for general and specific audiences (see the list at end of this chapter). Most projects take the productive route of writing their own guidelines, which are tied to the problems of their application environments and users. These guidelines are constructed from experience with existing interfaces, research results, and knowledgeable guesswork.

Iterative usability testing and expert reviews are appropriate during interface design. Once the initial interface is available, continuous refinements can be made on the basis of observations, surveys, interviews, usage log analysis, or more controlled empirical tests of novel strategies (Chapter 5). Agile processes emphasize lively design studio critiques of proposals and rapid trials of multiple alternatives to guide designers.

Feedback from users during the design process and for continuous refinement can provide useful insights and guidance. E-mail, web-based tools, and text messaging allow users to send comments directly to the designers, while logs of user behaviors provide designers with further evidence of what needs fixing. While searchable databases of user questions can often resolve problems and guide designers, online user consultants and fellow users can provide assistance and supportive encouragement.

1.4.3 Raising the user-interface consciousness of the general public

The media are so filled with stories about user interfaces that raising public consciousness of these tools may seem unnecessary. However, many people are still uncomfortable with the technologies they use. When they use a bank machine, a cell phone, or e-mail, they may feel fearful of making mistakes, anxious about damaging the equipment, worried about feeling incompetent, or threatened by the computer "being smarter than I am." These fears are generated, in part, by poor designs that have complex features, inconsistent terminology, confusing error messages, and tortuous sequences of actions.

One of our goals is to encourage users to translate their internal fears into outraged action. Instead of feeling guilty when they get a message such as DATA ERROR, users should express their anger at the user-interface designer who was so inconsiderate and thoughtless. Instead of feeling inadequate or foolish because they cannot remember a complex sequence of actions, they should complain to the designer who did not provide a more convenient mechanism or should seek another product that does.

Usability ultimately becomes a question of national priorities. Advocates of electronic voting and other services and promoters of e-healthcare and e-learning increasingly recognize the need to influence allocation of government resources and commercial research agendas. Policymakers and industry leaders become heroes when they facilitate access and promote quality, but they become villains when failures threaten children, disrupt travel, or menace consumers.

As examples of successful and satisfying interfaces become more visible, the crude designs appear archaic and will become commercial failures. As designers improve the user experience, some users' fears will recede, and the positive experiences of their competence, mastery, and satisfaction will flow in.

Practitioner's Summary

When designers of interactive systems conduct thorough user and task analyses, they are more likely to gain insights that will lead them to a proper functional design. They are more likely to have positive outcomes if they pay attention to reliability, availability, security, integrity, standardization, portability, integration, and the administrative issues of schedules and budgets. As design alternatives are proposed, evaluations can lead to shorter learning times, more rapid task performance, lower error rates, easier retention, and higher user satisfaction. Designers who accommodate the needs of children, older adults, and users with disabilities can improve the quality for all users. As designs are refined and implemented, evaluation by pilot studies, expert reviews, usability tests, user observations, user log analysis, and acceptance tests can accelerate improvement. Success in product design is measured in terms of evidence that universal usability is being attained (rather than testimonials from a few enthusiastic users). The proliferating literature and evidence-based guidelines will be of assistance in designing projects while accommodating the increasingly diverse and growing community of users.

Researcher's Agenda

The criteria for success in research favor innovations that work for broad communities of users performing useful tasks over longer time periods. At the same time, researchers are struggling to understand what kinds of imaginative consumer products will attract, engage, and satisfy diverse populations. The opportunities for researchers are unlimited. There are so many interesting, important, and doable projects that it may be hard to choose a direction. The goal of universal usability through plasticity of interface designs will keep researchers busy for years. Getting past vague promises and measuring user performance with alternate interfaces will be central to rapid progress. Each study has two parents: the practical problems facing designers and the fundamental theories based on principles of human behavior and interface design. Begin by proposing a lucid, testable hypothesis. Then consider the appropriate research methodology, conduct the study, collect the data, and analyze the results. Each study also has three children: specific recommendations for the practical problem, refinements of theories, and guidance for future researchers.

WORLD WIDE WEB RESOURCES

www.pearsonglobaleditions.com/shneiderman

This book is accompanied by a website (www.pearsonglobaleditions.com/shneiderman) that includes pointers to additional resources tied to the contents of each chapter. In addition, this website contains information for instructors, students, practitioners, and researchers. The links for Chapter 1 include pointers to general resources on human-computer interaction, such as professional societies, government agencies, companies, bibliographies, and guidelines documents.

Readers seeking references to scientific journals and conferences can consult the online searchable bibliography for human-computer interaction (http://www.hcibib.org/). Maintained since 1989, under the heroic leadership of Gary Perlman, the HCI Bibliography makes available more than 120,000 journal, conference, and book abstracts plus link collections on many topics, including consulting companies, education, history, and international development.

Some wonderful World Wide Web resources are:

- Resource on usability methods and guidelines from the U.S. government: http://www.usability.gov/
- IBM's extensive guide to user-centered design methods: http://www.ibm.com/design/
- Interaction Design Foundation's free online educational materials: https://www.interaction-design.org/
- Diamond Bullet Design: http://www.usabilityfirst.com/

E-mail lists for announcements and discussion lists are maintained by ACM SIGCHI (http://www.acm.org/sigchi/) and by the British HCI Group (http://www.bcs-hci.org.uk/), which also sponsors the frequently updated Usability News (http://usabilitynews.bcs.org/).

Discussion Questions

1. Devise an outline, consistent with the scientific method, which interface researchers should follow to validate their designs.
2. List some characteristics of successful user-interface designers with respect to their approach to solving UI problems.
3. As noted in this chapter, some skeptics feel that accommodating diversity requires dumbing-down or lowest-common-denominator strategies. However, the authors claim that in their experience, rethinking interface designs to accommodate these diversity situations will result in a better product for all users. Give an example of a product that meets the specific needs of a certain group of people, yet gives all users a better experience.
4. How can designers encourage novice users to use a system?
5. Suggest three usability measures that can be directly used to produce a practical evaluation of a system. Keep the goals of efficiency and satisfaction in mind with these measures.

References

Specialized references for this chapter appear here; general information resources are listed in the following section.

Bias, Randolph and Mayhew, Deborah (Editors), *Cost-Justifying Usability: An Update for the Internet Age*, 2nd Edition, Morgan Kaufmann, San Francisco, CA (2005).

Center for Information Technology Accommodation, Section 508: The road to accessibility, General Services Administration, Washington, DC (2015). Available at http://www.section508.gov.

Friedman, C., Rubin, J., Brown, J., Buntin, M., Corn, M., Etheredge, L., Gunter, C., Musen, M., Platt, R., Stead, W., Sullivan, K., and Van Houweling, D., Toward a science of learning systems: A research agenda for the high-functioning learning health system, *Journal of the American Medical Informatics Association* (2014), 1–6.

Froehlich, Jon, Findlater, Leah, and Landay, James, The design of eco-feedback technology, *Proceedings CHI 2010 Conference: on Human Factors in Computing Systems*, ACM Press, New York (2010), 1999–2008.

Jones, Douglas W., and Simons, Barbara, *Broken Ballots: Will Your Vote Count?* Center for the Study of Language and Information (2012).

Norooz, L., Mauriello, M., McNally, B., Jorgenson, A., and Froehlich, J., BodyVis: A new approach to body learning through wearable sensing and visualization, *Proceedings CHI 2015 Conference: Human Factors in Computing Systems, ACM Press, New York* (2015), 1025–1034.

Summers, K., Chisnell, D., Davies, D., Alton, N., and McKeever, M., Making voting accessible: Designing digital ballot marking for people with low literacy and mild cognitive disabilities. *USENIX Journal of Election Technology and Systems (JETS) 2,* 2 (2014).

Tullis, Thomas, and Albert, William, *Measuring the User Experience: Collecting, Analyzing, and Presenting Usability Metrics,* 2nd Edition, Morgan Kaufmann (2013).

Whitworth, Brian and De Moor, Aldo (Editors), *Handbook of Research on Socio-Technical Design and Social Networking Systems*, IGI Global, Hershey, PA(2009).

General information resources

Primary journals include the following:

ACM Interactions: A Magazine for User Interface Designers, ACM Press

ACM Transactions on Accessible Computing, ACM Press

ACM Transactions on Computer-Human Interaction (TOCHI), ACM Press

AIS Transactions on Human-Computer Interaction, AIS

Behaviour & Information Technology (BIT), Taylor & Francis Ltd.

Computer Supported Cooperative Work, Springer

Human-Computer Interaction, Taylor & Francis Ltd.

Information Visualization, Sage

Interacting with Computers, Oxford University Press

International Journal of Human-Computer Interaction, Taylor & Francis Ltd.

International Journal of Human-Computer Studies, Elsevier

Journal of Usability Studies, User Experience Professionals Association

Universal Access in the Information Society, Springer

Other journals that regularly carry articles of interest include:

ACM: Communications of the ACM (CACM)

ACM Transactions on Graphics

ACM Transactions on Information Systems

ACM Transactions on Interactive Intelligent Systems

ACM Transactions on the Web

Cognitive Science

Computers in Human Behavior

Ergonomics

Human Factors

IEEE Computer

IEEE Computer Graphics and Applications

IEEE Transactions on Human-Machine Systems

IEEE Transactions on Visualization and Computer Graphics

Journal of Computer-Mediated Communication

Journal of Visual Languages and Computing

Personal and Ubiquitous Computing

Presence

Psychology

Technical Communication

User Modeling and User-Adapted Interaction

Virtual Reality

The Association for Computing Machinery (ACM) has a Special Interest Group on Computer-Human Interaction (SIGCHI), which holds regularly scheduled conferences. ACM also publishes the highly regarded *Transactions on Human-Computer Interaction* and the lively magazine *Interactions*. Other ACM Special Interest Groups, such as Graphics and Interactive Techniques (SIGGRAPH), Accessible Computing (SIGACCESS), Multimedia (SIGMM), and Hypertext and the Web (SIGWEB), also produce conferences and newsletters. Other relevant ACM groups are Computers and Society (SIGCAS), Design of Communication (SIGDOC), Groupware (SIGGROUP), Information Retrieval (SIGIR), and Mobility of Systems, Users, Data, and Computing (SIGMOBILE).

The IEEE Computer Society, through its many conferences, transactions, and magazines, covers user-interface issues. Similarly, the business-oriented Association for Information Systems (AIS) has a SIGHCI that publishes a journal and runs sessions at several conferences. The long-established Human Factors & Ergonomics Society also runs annual conferences and has a Computer Systems Technical Group with a newsletter. Additionally, the Society for Technical Communications (STC), the American Institute of Graphic Arts (AIGA), the International Ergonomics Association, and the Ergonomics Society increasingly focus on user interfaces. The influential business-oriented User Experience Professionals Association (UXPA) publishes the *UX—User Experience* magazine and the online *Journal of Usability Studies*. The UXPA also spawned the annual World Usability Day with hundreds of events around the world each November.

The International Federation for Information Processing has a Technical Committee (TC.13) and Working Groups on Human-Computer Interaction. The British Computer Society Human-Computer Interaction Group has held an international conference since 1985. The French Association Francophone pour l'Interaction Homme-Machine (AFIHM), the Spanish Asociación Interacción Persona-Ordenador (AIPO), and other associations promote HCI within their language communities. Other groups conduct important events in South Africa, Australia/New Zealand, Scandinavia, Asia, Latin America, and elsewhere.

Conferences—such as the ones held by the ACM (especially SIGCHI), IEEE, Human Factors & Ergonomics Society, and IFIP—often have relevant papers presented and published in the proceedings. INTERACT, Human-Computer Interaction International, and Work with Computing Systems are conference series that cover user-interface issues broadly. Many specialized conferences may also be of interest: for example, User Interfaces Software and Technology, Hypertext, Computer-Supported Cooperative Work, Intelligent User Interfaces, Computers and Accessibility, Ubiquitous Computing, Computers and Cognition, Designing Interactive Systems, and more.

Brad Myers's brief history of HCI (*ACM Interactions*, March 1998) is one starting point for those who want to study the emergence and evolution of this field. James Martin provided a thoughtful and useful survey of interactive systems in his 1973 book *Design of Man-Computer Dialogues*. Ben Shneiderman's 1980 book *Software Psychology: Human Factors in Computer and Information Systems* promoted the use of controlled experimental techniques and scientific research methods. Rubinstein and Hersh's *The Human Factor: Designing Computer Systems for People* (1984) offered an appealing introduction to computer-system design and many useful guidelines. The first edition of this book, published in 1987, reviewed critical issues, offered guidelines for designers, and suggested research directions.

A steady flow of influential books has stimulated widespread media and public attention about usability issues, including Nielsen's *Usability Engineering* (1993), Landauer's *The Trouble with Computers* (1995), and Nielsen's *Designing Web Usability* (1999). Don Norman's 1988 book *The Psychology of Everyday Things* (reprinted and revised in 2013 as *The Design of Everyday Things*) is a refreshing look at the psychological issues involved in the design of the everyday technology that surrounds us.

As the field matured, subgroups and publications centered around specialized topics emerged; this happened with mobile computing, web design, online communities, information visualization, virtual environments, and so on. The following list of guidelines documents and books is a starting point to an exploration of the large and growing literature.

Guidelines documents

Apple Computer, Inc., *Human Interface Guidelines, Version for the Mac OS X, iPhone, iPad, and Apple Watch*, Apple, Cupertino, CA (April 2015). Available at http://developer.apple.com/.

—Explains how to design consistent visual and behavioral properties for Apple products.

International Organization for Standardization, *ISO 9241 Ergonomics of Human-System Interaction*, Geneva, Switzerland (updated 2013). Available at http://www.iso.org/.

—Thorough general introduction, covering dialog principles, guidance on usability, presentation of information, user guidance, menu dialogs, command dialogs, direct-manipulation dialogs, form-filling dialogs, and much more. This is an important source for many countries and companies.

Microsoft, Inc., *The Microsoft Windows User Experience Interaction Guidelines*, Redmond, WA (2015). Available at https://msdn.microsoft.com/.

—Describes design principles, controls, text, interaction, windows, and aesthetics.

United Kingdom Health & Social Care Information Centre, *User Interface Guidance* (June 2015). Available at http://systems.hscic.gov.uk/data/cui/uig.

—Detailed guidelines oriented to medical systems.

United Kingdom Ministry of Defence, *Human Factors for Designers of Systems*, Defence Standard 00-250 (June 2013). Available at http://www.dstan.mod.uk/data/00/250/00000100.pdf.

—Describes human factors, integration processes, requirements, and acceptance testing.

U.S. Dept. of Defense, *Human Engineering Design Criteria Standard*, Military Standard MIL-STD–1472G, U.S. Government Printing Office, Washington, DC (2012).

—Covers traditional ergonomic and anthropometric issues. Later editions pay increasing attention to user-computer interfaces. Interesting and thought-provoking reminder of many human-factors issues.

U.S. Federal Aviation Administration, *The Human Factors Design Standard*, Atlantic City, NJ (updated May 2012). Available at http://hf.tc.faa.gov/hfds/.

—Extensive compilation of human-factors standards for contractors to follow, especially relevant to aircraft and air-traffic control.

U. S. National Cancer Institute, *Research-based Web Design and Usability Guidelines*, Dept. of Health & Human Services, National Institutes of Health (2006, updated on the web 2015). Available at http://guidelines.usability.gov/.

—Authoritative and packed with numerous full-color examples of information-oriented websites.

World Wide Web Consortium's Web Accessibility Initiative, *Web Content Accessibility Guidelines 2.0* (2008). Available at http://www.w3.org/WAI/.

—Practical, implementable three-level prioritization of web design guidelines for users with disabilities. The Web Accessibility Initiative (WAI) develops strategies, guidelines, and resources to help make the web accessible to people with disabilities. Four principles are offered: Perceivable, Operable, Understandable, and Robust.

World Wide Web Consortium, *Web Accessibility Evaluation Tools* (2014). Available at http://www.w3.org/WAI/ER/tools/.

—An occasionally updated list of software tools related to accessibility; demonstrates lively activity.

Books

Allen, J., and Chudley, J., *Smashing UX Design: Foundations for Designing Online User Experiences*, Wiley, Chichester (2012).

Anderson, S., *Seductive Interaction Design: Creating Playful, Fun, and Effective User Experiences,* New Riders (2011).

Barnum, Carol M., *Usability Testing Essentials: Ready, Set… Test!* Morgan Kaufmann (2011).

Baxter, Kathy, and Courage, Catherine, *Understanding Your Users: A Practical Guide to User Requirements Methods, Tools, and Techniques*, 2nd Edition, Morgan Kaufmann (2015).

Bell, Genevieve, and Dourish, Paul, *Divining a Digital Future: Mess and Mythology in Ubiquitous Computing*, MIT Press (2011).

Berkman, Eric, and Hoober, Steven, *Designing Mobile Interfaces,* O'Reilly, Sebastopol, CA (2011).

Boy, Guy, *Handbook of Human-Machine Interaction*, Ashgate (2011).

Boy, Guy, *Orchestrating Human-Centered Design*, Springer (2013).

boyd, danah, *It's Complicated: The Social Lives of Networked Teens*, Yale University Press (2014).

Buley, Leah, *The User Experience Team of One: A Research and Design Survival Guide*, Rosenfeld Media (2013).

Cairns, P., and Cox, A. L. (Editors), *Research Methods for Human-Computer Interaction*, Cambridge University Press (2008, reprinted 2011).

Calvo, Rafael A., and Peters, Dorian, *Positive Computing: Technology for Wellbeing and Human Potential*, MIT Press (2014).

Carroll, J. (Editor), *Creativity and Rationale: Enhancing Human Experience by Design*, Springer (2013).

Chandler, C., and Unger, R., *A Project Design to UX Design: For User Experience Designers in the Field or in the Making*, 2nd Edition, New Riders (2012).

Chandler, C., and van Slee, A., *Adventures in Experience Design*, New Riders (2014)

Chapman, C., *The Smashing Idea Book: From Inspiration to Application*, Wiley (2011).

Colborne, Giles, *Simple and Usable Web, Mobile, and Interaction Design (Voices That Matter)*, New Riders, Berkeley, CA (2011).

Cooper, A., Reimann, R., Cronin, D., and Noessel, C., *About Face: The Essentials of Interface Design*, 4th Edition, John Wiley & Sons (2104).

Crabtree, Andrew, Rouncefield, Mark, and Tolmie, Peter, *Doing Design Ethnography*, Springer, London (2012).

Craig, Alan B., *Understanding Augmented Reality: Concepts and Applications*, Morgan Kaufmann, San Francisco, CA (2013).

Cunningham, Katie, *Accessibility Handbook*, O'Reilly Publishing (2012).

Dannen, Chris, *iPhone Design Award-Winning Projects (The Definitive Guide)*, Apress, Berkeley, CA (2010).

Ferster, Bill, *Interactive Visualization: Insight through Inquiry*, MIT Press (2012).

Garrett, J. J., *The Elements of User Experience: User-centered Design for the Web and Beyond*, 2nd Edition, New Riders Press (2010).

Goodman, E., Kuniavsky, M., and Moed, A., *Observing the User Experience*, 2nd Edition, Morgan Kaufmann (2012).

Gothelf, J., and Seiden, J. (Editors), *Lean UX: Applying Lean Principles to Improve User Experience*, O'Reilly Media (2013).

Greenberg, S., Carpendale, S., Marquardt, N., and Buxton, B., *Sketching User Experiences: The Workbook*, Morgan Kaufmann (2012).

Harris, D., *Writing Human Factors Research Papers: A Guidebook*. Ashgate Publishing (2012).

Hartson, R., and Pyla, P., *The UX Book: Process and Guidelines for Ensuring a Quality User Experience*, Morgan Kaufmann (2012).

Hearst, Marti A., *Search User Interfaces*, Cambridge University Press, New York, NY (2009).

Hinman, Rachel, *The Mobile Frontier: A Guide for Designing Mobile Experiences*, Rosenfeld Media, Brooklyn, NY (2012).

Holtzblatt, Karen and Beyer, Hugh, *Contextual Design Evolved*, Morgan & Claypool (2014).

Horton, Sarah and Quesenbery, Whitney, *A Web for Everyone: Designing Accessible User Experiences*, Rosenfeld Media (2015).

Johnson, Jeff, *Designing with the Mind in Mind: Simple Guide to Understanding User Interface Design Rules*, 2nd Edition, Morgan Kaufmann, San Francisco, CA (2014).

Jones, P., *Design for Care: Innovating the Healthcare Experience*, Rosenfeld Media (2013).

Katz, J., *Designing Information: Human Factors and Common Sense in Information Design*. John Wiley (2012).

Kim, Gerard Jounghyun, *Human-Computer Interaction: Fundamentals and Practice,* CRC Press (2015).

Kipper, Greg, and Rampolla Joseph, *Augmented Reality: An Emerging Technologies Guide to AR,* Syngress (2012).

Klimczak, E., *Design for Software,* Wiley (2013).

Kolko, J., *Thoughts on Interaction Design*, 2nd Edition, Morgan Kaufmann (2011).

Koskinen, I., Zimmerman, J., Binder, T., Redstrom, J., and Wensveen, S., *Design Research through Practice from the Lab, Field, and Showroom,* Morgan Kaufmann (2011).

Kraft, C., *User Experience Innovation: User Centered Design That Works*, Apress (2012).

Krug, Steve, *Rocket Surgery Made Easy*: *The Do-it-yourself Guide to Finding and Fixing Usability Problems*, New Riders (2010).

Krug, Steve, *Don't Make Me Think, Revisited: A Common Sense Approach to Web Usability*: 3rd Edition *(Voices That Matter),* New Riders, Berkeley, CA (2014).

Lazar, J., Feng, J. H., and Hochheiser, H., *Research Methods in Human-Computer Interaction*, Wiley (2010).

Lazar, Jonathan, Goldstein, Daniel F., and Taylor, Anne, *Ensuring Digital Accessibility through Process and Policy*, Morgan Kaufmann (2015).

Levin, Michal, *Designing Multi-Device Experiences: An Ecosystem Approach to User Experiences across Devices,* O'Reilly Media, Sebastopol, CA (2014).

Lund, A., *User Experience Management: Essential Skills for Leading Effective UX Teams,* Morgan Kaufmann (2011).

MacKenzie, I. Scott, *Human-Computer Interaction: An Empirical Research Perspective,* Morgan Kaufmann, San Francisco, CA (2013).

Mariani, Joseph, Rosset, Sophie, Garnier-Rizet, and Devillers, Laurence (Editors), *Natural Interaction with Robots, Knowbots and Smartphones: Putting Spoken Dialog Systems into Practice*, Springer (2014).

McKay, E., *UI Is Communication: How to Design Intuitive, User-centered Interfaces by Focusing on Effective Communication*, Morgan Kaufmann (2013).

Moffett, Jack, *Bridging UX and Web Development*, Morgan Kaufmann (2014).

Monk, Andrew (Editor), *Fundamentals of Human-Computer Interaction*, Morgan Kaufmann/Academic Press (2014).

Nagel, Wolfram, *Multiscreen UX Design: Developing for a Multitude of Devices,* Morgan Kaufmann (2015).

Nakano, Y., Conati, C., Bader, T. (Editors), *Eye Gaze in Intelligent User Interfaces: Gaze-based Analyses, Models, and Applications*, Springer (2013).

Neil, Theresa, *Mobile Design Pattern Gallery: UI Patterns for Smartphone Apps*, 2nd Edition, O'Reilly (2014).

Nielsen, Jakob, and Pernice, Kara, *Eyetracking Web Usability,* New Riders, Berkeley, CA (2010).

Nielsen, Jakob, and Budiu, Raluca, *Mobile Usability,* New Riders, Berkeley, CA (2012).

Nielsen, Lene, *Personas: User Focused Design,* Springer, London (2013).

Norman, D., *The Design of Everyday Things: Revised and Expanded Edition,* Basic Books (2013).

Nudelman, G., *Android Design Patterns*, John Wiley (2013). Available at http://www.androiddesignbook.com.

Olson, J. S., and Kellogg, W. A. (Editors), *Ways of Knowing in HCI*, Springer (2014).

Oviatt, Sharon, and Cohen, Philip, *The Paradigm Shift to Multimodality in Contemporary Computer Interfaces*, Morgan & Claypool (2015).

Parush, Avi, *Conceptual Design for Interactive Systems: Designing for Performance and User Experience,* Elsevier/Morgan Kaufmann (2015).

Pratt, A., and Nunes, J., *Interactive Design: An Introduction to the Theory and Application of User-Centered Design*, Rockport Publishers (2012).

Preece, J., Rogers, Y., and Sharp, H., *Interaction Design: Beyond Human-Computer Interaction*, 4th Edition, Wiley, New York, NY (2015).

Purchase, Helen, *Experimental Human-Computer Interaction: A Practical Guide with Visual Examples*, Cambridge University Press (2012).

Quesenbery, Whitney, and Szuc, Daniel, *Global UX: Design and Research in a Connected World*, Morgan Kaufmann (2011).

Redish, Janice, *Letting Go of the Words: Writing Web Content That Works (Interactive Technologies),* 2nd Edition, Morgan Kaufmann, San Francisco, CA (2012).

Reiss, E., *Usable Usability: Simple Steps for Making Stuff Better,* Wiley (2012).

Ritter, Frank E., Baxter, Gordon D., and Churchill, Elizabeth F., *Foundations for Designing User-Centered Systems: What System Designers Need to Know about People*, Springer, London (2014).

Robinson, Simon, Jones, Matt, and Marsden, Gary, *There's Not an App for That: Mobile User Experience Design for Life*, Morgan Kaufmann (2015).

Salgado, L. C. C., Leitão, C. F., and de Souza, C. S., *A Journey through Cultures: Metaphors for Guiding the Design of Cross-Cultural Interactive Systems*, Springer (2012).

Sauro, J., and Lewis, J., *Quantifying the User Experience: Practical Statistics for User Research,* Morgan Kaufmann (2012).

Schlatter, T., and Levinson, D., *Visual Usability: Principles and Practices for Designing Digital Applications*, Morgan Kaufmann (2013).

Sharon, Tomer, *It's Our Research: Getting Stakeholder Buy-in for User Experience Research Projects*, Morgan Kaufmann (2012).

Stephanidis, Constantine, *The Universal Access Handbook (Human Factors and Ergonomics)*, CRC Press, Boca Raton, FL (2009).

Tullis, Thomas, and Albert, William, *Measuring the User Experience: Collecting, Analyzing, and Presenting Usability Metrics,* 2nd Edition, Morgan Kaufmann (2013).

Weinschenk, S., *100 Things Every Designer Needs to Know about People*, New Riders (2011).

Wigdor, Daniel, and Wixon, Dennis, *Brave NUI World: Designing Natural User Interfaces for Touch and Gesture*, Morgan Kaufmann, San Francisco, CA (2011).

Wilson, Chauncey, *Brainstorming and Beyond: A User-Centered Design Method,* Elsevier/Morgan Kaufmann, Burlington (2013).

Wilson, Chauncey, *Interview Techniques for UX Practitioners: A User-Centered Design Method*, Elsevier/Morgan Kaufmann, Burlington (2013).

Wilson, Chauncey, *User Interface Inspection Methods,* Elsevier/Morgan Kaufmann, Amsterdam (2014).

Wilson, C. (Editor), *User Experience Re-Mastered: Your Guide to Getting the Right Design*, Morgan Kaufmann (2010).

Wilson, Max L., *Search User Interface Design (Synthesis Lectures on Information Concepts, Retrieval, and Services),* Morgan & Claypool Publishers, San Rafael, CA (2011).

Videos

Video is an effective medium for presenting the dynamic, graphical, and interactive nature of modern user interfaces. A wonderful set of lectures from Stanford University's CS547 Human-Computer Interaction Seminar can be found at http://hci.stanford.edu/courses/cs547/.

Inspirational videos from the annual Technology, Entertainment & Design (TED) Conference, which covers a wide range of topics including visionary user-interface themes, are found at http://www.ted.com/index.php/talks/. Another exceptional resource is YouTube (http://www.youtube.com/), where a search on "user interfaces" produces a list of hundreds of recent product demonstrations, research reports, and some clever and funny technology demonstrations.

This page intentionally left blank

CHAPTER 2

Universal Usability

> "Social scientists have shown that teams and organizations whose members are heterogeneous in meaningful ways, for example, in skill set, education, work experiences, perspectives on a problem, cultural orientation, and so forth, have a higher potential for innovation than teams whose members are homogeneous."
>
> **Beryl Nelson**
> *Communications of the ACM*, November 2014

> "I feel . . . an ardent desire to see knowledge so disseminated through the mass of mankind that it may, at length, reach even the extremes of society: beggars and kings."
>
> **Thomas Jefferson**
> Reply to American Philosophical Society, 1808

CHAPTER OUTLINE

2.1 Introduction

The remarkable diversity of human abilities, backgrounds, motivations, personalities, cultures, and work styles challenges interface designers. A young female designer in India with computer training and a desire for rapid interaction using densely packed displays may have a hard time designing a successful interface for older male artists in France with a more leisurely and free-form work style. Understanding the physical, intellectual, and personality differences among users is vital for expanding market share, supporting required government services, and enabling creative participation by the broadest possible set of users. As a profession, we will be remembered for how well we meet our users' needs. That's the ultimate goal: addressing the needs of all users (Fig. 2.1).

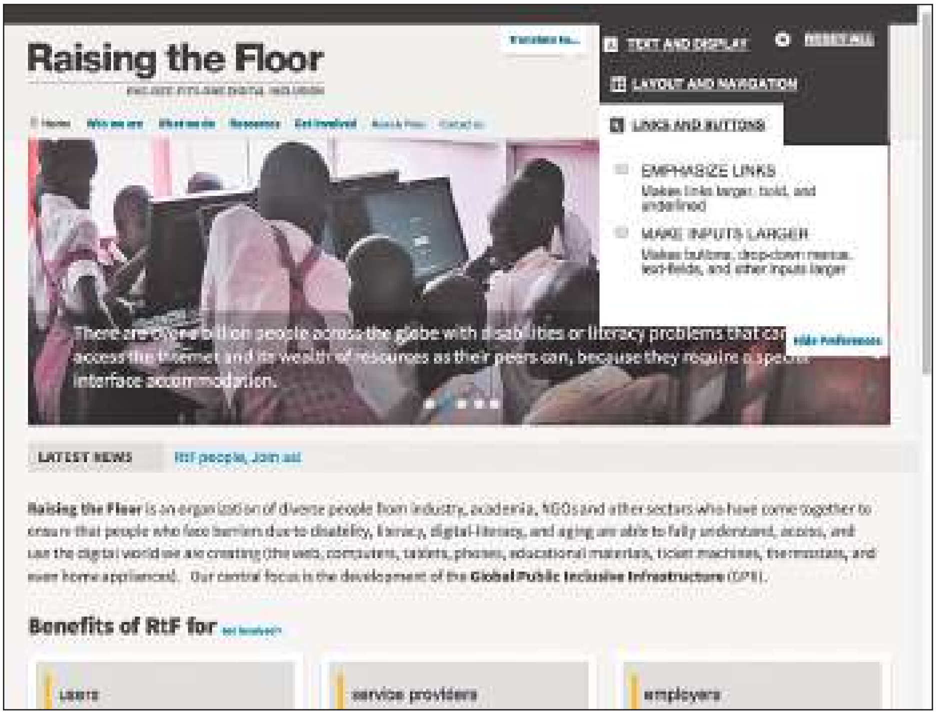


FIGURE 2.1

The website of Raising the Floor includes universal accessibility features such as options for emphasizing the links or making buttons larger, offering several font sizes, contrast, text descriptions of photos, translation services, and so on (http://www.raisingthefloor.net).

The huge international consumer market in mobile devices has raised the pressure for designs that are universally usable. While skeptics suggest that accommodating diversity requires dumbing-down or lowest-common-denominator strategies, our experience is that rethinking interface designs for differing situations often results in a better product for all users. Measures to accommodate the special needs of one group, such as curb cuts in sidewalks for wheelchair users, often have payoffs for many groups, such as parents with baby strollers, skateboard riders, travelers with wheeled luggage, and delivery people with handcarts. With this in mind, this chapter introduces the challenges posed by physical, cognitive, perceptual, personality, and cultural differences. It covers considerations for users with disabilities, older adults, and young users, ending with a discussion of hardware and software diversity. The important issues of different usage profiles (novice, intermittent, and expert), wide-ranging task profiles, and multiple interaction styles are covered in Chapter 3.

2.2 Variations in Physical Abilities and Physical Workplaces

Accommodating diverse human perceptual, cognitive, and motor abilities is a challenge to every designer. Fortunately, *ergonomics* researchers and practitioners have gained substantial experience from design projects with automobiles, aircraft, cellphones, and so on. This experience can be applied to the design of user interfaces and mobile devices.

Basic data about human dimensions comes from research in *anthropometry* (Preedy, 2012). Thousands of measures of hundreds of features of people—male and female, young and adult, European and Asian, underweight and overweight, tall and short—provide data to construct 5- to 95-percentile design ranges. Head, mouth, nose, neck, shoulder, chest, arm, hand, finger, leg, and foot sizes have been carefully cataloged for a variety of populations. The great diversity in these static measures reminds us that there can be no image of an "average" user and that compromises must be made or multiple versions of a system must be constructed.

Cellphone keypad design parameters—placement, size, distance between keys, and so forth (Section 10.2)—evolved to accommodate differences in users' physical abilities. People with especially large or small hands may have difficulty using standard cellphones or keyboards, but a substantial fraction of the population is well served by one design. On the other hand, since screen-brightness preferences vary substantially, designers often enable users to control this parameter. Similarly, controls for chair seat and back heights and for display angles allow individual adjustment. When a single design cannot accommodate a large fraction of the population, multiple versions or adjustment controls are helpful.

Physical measures of static human dimensions are not enough. Measures of dynamic actions—such as reach distance while seated, speed of finger presses, or strength of lifting—are also necessary.

Since so much of work is related to perception, designers need to be aware of the ranges of human perceptual abilities, especially with regard to vision (Ware, 2012). For example, researchers consider human response time to varying visual stimuli or time to adapt to low or bright light. They examine human capacity to identify an object in context or to determine the velocity or direction of a moving point. The visual system responds differently to various colors, and some people have color deficiencies, either permanently or temporarily (due to illness or medication). People's spectral range and sensitivity vary, and peripheral vision is quite different from the perception of images in the fovea (the central part of the retina). Designers need to study flicker, contrast, motion sensitivity, and depth perception as well as the impact of glare and visual fatigue. Finally, designers must consider the needs of people who wear corrective lenses, have visual impairments, or are blind.

Other senses are also important: for example, touch for keyboard or touchscreen entry and hearing for audible cues, tones, and speech input or output (Chapter 10). Pain, temperature sensitivity, taste, and smell are rarely used for input or output in interactive systems, but there is room for imaginative applications.

These physical abilities influence elements of the interactive-system design. They also play a prominent role in the design of the workplace or workstation (or playstation). The *Human Factors Engineering of Computer Workstations* standard (HFES, 2007) lists these concerns:

- Worktable and display-support height
- Clearance under work surface for legs
- Work-surface width and depth
- Adjustability of heights and angles for chairs and work surfaces
- Posture—seating depth and angle, backrest height, and lumbar support
- Availability of armrests, footrests, and palmrests
- Use of chair casters

Workplace design is important in ensuring high job satisfaction, good performance, and low error rates. Incorrect table heights, uncomfortable chairs, or inadequate space to place documents can substantially impede work. The standards document also addresses such issues as illumination levels (200 to 500 lux); glare reduction (antiglare coatings, baffles, mesh, positioning); luminance balance and flicker; equipment reflectivity; acoustic noise and vibration; air temperature, movement, and humidity; and equipment temperature.

The most elegant screen design can be compromised by a noisy environment, poor lighting, or a stuffy room, and that compromise will eventually lower performance, raise error rates, and discourage even motivated users. Thoughtful designs, such as workstations that provide wheelchair access and good lighting, will be even more appreciated by users with disabilities and older adults.

Another physical-environment consideration involves room layout and the sociology of human interaction. With multiple workstations in a classroom or office, different layouts can encourage or limit social interaction, cooperative work, and assistance with problems. Because users can often quickly help one another with minor problems, there may be an advantage to layouts that group several terminals close together or that enable supervisors or teachers to view all screens at once from behind. On the other hand, programmers, reservations clerks, or artists may appreciate the quiet and privacy of their own workspaces.

Mobile devices are increasingly being used while walking or driving and in public spaces, such as restaurants or trains where lighting, noise, movement, and vibration are part of the user experience. Designing for these more fluid environments presents opportunities for design researchers and entrepreneurs.

2.3 Diverse Cognitive and Perceptual Abilities

A vital foundation for interactive-system designers is an understanding of the cognitive and perceptual abilities of the users (Radvansky and Ashcraft, 2013). The journal *Ergonomics Abstracts* offers this classification of human cognitive processes:

- Short-term and working memory
- Long-term and semantic memory
- Problem solving and reasoning
- Decision making and risk assessment
- Language communication and comprehension
- Search, imagery, and sensory memory
- Learning, skill development, knowledge acquisition, and concept attainment

It also suggests this set of factors affecting perceptual and motor performance:

- Arousal and vigilance
- Fatigue and sleep deprivation
- Perceptual (mental) load
- Knowledge of results and feedback
- Monotony and boredom
- Sensory deprivation
- Nutrition and diet
- Fear, anxiety, mood, and emotion
- Drugs, smoking, and alcohol
- Physiological rhythms

These vital issues are not discussed in depth in this book, but they have a profound influence on the design of user interfaces. The term *intelligence* is not included in this list because its nature is controversial and measuring different forms of intelligence is difficult.

In any application, background experience and knowledge in the task and interface domains play key roles in learning and performance. Task- or computer-skill inventories can be helpful in predicting performance.

2.4 Personality Differences

Some people are eager to use computers and mobile devices, while others find them frustrating. Even people who enjoy using these technologies may have very different preferences for interaction styles, pace of interaction, graphics versus tabular presentations, dense versus sparse data presentation, and so on. A clear understanding of personality and cognitive styles can be helpful in designing interfaces for diverse communities of users.

One evident difference is between men and women, but no clear pattern of gender-related preferences in interfaces has been documented. While the majority of video-game players and designers are young males, some games (such as *The Sims*™, *Candy Crush Saga*, and *Farmville*) draw ample numbers of female players. Designers can get into lively debates about why many women prefer certain games, often speculating that women prefer less violent action and quieter soundtracks. Other conjectures are that women prefer social games, characters with appealing personalities, softer color patterns, and a sense of closure and completeness. Can these informal conjectures be converted to measurable criteria and then validated?

Turning from games to productivity tools, there is also a range of reactions to violent terms such as KILL a process or ABORT a program. These and other potentially unfortunate mismatches between the user interface and the users might be avoided by more thoughtful attention to individual differences among users.

Unfortunately, there is no simple taxonomy of user personality types. A popular, but controversial, technique is the Big Five Test, based on the OCEAN model (Wiggins, 1996): Openness to Experience/Intellect (closed/open), Conscientiousness (disorganized/organized), Extraversion (introverted/extraverted), Agreeableness (disagreeable/agreeable), and Neuroticism (calm/nervous). There are hundreds of other psychological scales, including risk taking versus risk avoidance; internal versus external locus of control; reflective versus impulsive behavior; convergent versus divergent thinking; high versus low anxiety; tolerance for stress; tolerance for ambiguity, motivation, or compulsiveness; field dependence versus independence; assertive versus passive personality; and left- versus right-brain orientation. As designers explore computer applications for the home, education, art, music, and entertainment, they may benefit from paying greater attention to personality types. Consumer-oriented researchers are especially aware of the personality distinctions across market segments, so as to tune their advertising for niche products designed for tech-savvy youngsters versus family-oriented parents.

Another approach to personality assessment is by studying user behavior. For example, some users file thousands of e-mails in a well-organized hierarchy of folders, while others keep them all in the inbox, using search strategies to find what they want later. These distinct approaches may well relate to personality variables, giving designers the clear message that multiple requirements must be satisfied by their designs.

2.5 Cultural and International Diversity

Another perspective on individual differences has to do with cultural, ethnic, racial, or linguistic background (Quesenbery and Szuc, 2011; Marcus and Gould, 2012; Salgado, 2012). Users who were raised learning to read Japanese or Chinese will scan a screen differently from users who were raised learning to read English or French. Users from reflective or traditional cultures may prefer interfaces with stable displays from which they select a single item, while users from action-oriented or novelty-based cultures may prefer animated screens and multiple clicks. Preferred content of webpages also varies; for example, university home pages in some cultures emphasize their impressive buildings and respected professors lecturing to students, while others highlight student team projects and a lively social life. Mobile device preferences also vary across cultures that lead to rapidly changing styles in successful apps, which may include playful designs, music, and game-like features.

More and more is being learned about computer users from different cultures, but user experience designers are still struggling to establish guidelines that are appropriate across multiple languages and cultures (Sun, 2012; Pereira and Baranauskas, 2015). The growth of a worldwide computer and mobile device market means that designers must prepare for internationalization. Software architectures that facilitate customization of local versions of user interfaces offer a competitive advantage (Reinecke and Bernstein, 2013). For example, if all text (instructions, help, error messages, labels, and so on) is stored in files, versions in other languages can be generated with little or no additional programming. Hardware issues include character sets, keyboards, and special input devices. User-interface design concerns for internationalization include the following:

- Characters, numerals, special characters, and diacriticals
- Left-to-right versus right-to-left versus vertical input and reading
- Date and time formats
- Numeric and currency formats
- Weights and measures
- Telephone numbers and addresses
- Names and titles (*Mr., Ms., Mme., M., Dr.*)
- Social Security, national identification, and passport numbers
- Capitalization and punctuation
- Sorting sequences
- Icons, buttons, and colors
- Pluralization, grammar, and spelling
- Etiquette, policies, tone, formality, and metaphors

The list is long and yet incomplete. Recent studies of consumer use show performance and preference differences for information density, animation, cute characters, eagerness for timely updates, incentives for social participation, and game-like features. Whereas early designers were often excused from cultural and linguistic slips, the current highly competitive atmosphere means that more effective localization may produce a strong advantage. To develop effective designs, companies run usability studies with users from different countries, cultures, and language communities.

The role of information technology in international development is steadily growing, but much needs to be done to accommodate the diverse needs of users with vastly different language skills and technology access. To promote international efforts to foster successful implementation of information technologies, representatives from around the world meet regularly for the United Nations World Summit on the Information Society. They declared their

> desire and commitment to build a people-centered, inclusive and development-oriented Information Society, where everyone can create, access, utilize and share information and knowledge, enabling individuals, communities and peoples to achieve their full potential in promoting their sustainable development and improving their quality of life, premised on the purposes and principles of the Charter of the United Nations and respecting fully and upholding the Universal Declaration of Human Rights.

The plan calls for applications to be "accessible to all, affordable, adapted to local needs in languages and culture, and [to] support sustainable development." The UN Sustainability Development Goals include eradicate extreme poverty and hunger; reduce child mortality; combat HIV/AIDS, malaria, and other diseases; and ensure environmental sustainability. Information and communications technologies can play important roles in developing the infrastructure that is needed to achieve these goals (Fig. 2.2).

FIGURE 2.2

Designing for cellphones can open the door to a wider audience (Medhi et al., 2011), for example, in developing countries where feature phones often are the only way to access the internet, literacy may be an issue, and users have very low monthly limits on the data volume they can use.

2.6 Users with Disabilities

When digital content and services can be flexibly presented in different formats, all users benefit (Horton and Quesenbery, 2014). However, flexibility is most appreciated by users with disabilities who now can access content and services using diverse input and output devices. Blind users may utilize screen readers (speech output such as JAWS or Apple's VoiceOver) or refreshable braille displays, while low-vision users may use magnification. Users with hearing impairments may need captioning on videos and transcripts of audio, and people with limited dexterity or other motor impairments may utilize speech recognition, eye-tracking, or alternative keyboards or pointing devices (Fig. 2.3). Increasingly, especially on Apple products, these alternate forms of input or output are integrated into technology out of the box (other laptops, tablets, and smartphones have add-on screen reader and magnification capability, and a small number of laptops have built-in eye tracking).

There is a long history of research on how users with perceptual or motor impairments (such as those described above) interact with technology, and research on intellectual or cognitive impairments is now also increasing (Blanck, 2014; Chourasia et al., 2014). In some cases, people with intellectual impairments

FIGURE 2.3

A young man uses a wheelchair-mounted augmentative communication and control device to control a standard television. New universal remote console standards can allow people to use communication aids and other personal electronics as alternate interfaces for digital electronics in their environments (http://trace.wisc.edu).

need transformation of content, but in other cases, no modifications or assistive technologies are needed. Designing for accessibility helps everyone. The same captioning on video that is utilized by users with hearing impairments is also used by users watching video in noisy locations, such as gyms, bars, and airports. Many accessibility features help with graceful presentation of content in multiple formats, allowing for flexibility in presentation on small screens of mobile devices or with audio output instead of visual output. As users are increasingly on the go and experience "situational impairments," these accessibility features help all users, who may be in situations where they cannot see their screen (e.g., they are driving a car) or cannot play audio out loud (e.g., on a plane).

For interfaces to be accessible for people with disabilities, they generally need to follow a set of design guidelines for accessibility. The international standards for accessibility come from the Web Accessibility Initiative, a project of the World Wide Web Consortium. The best-known standards are the Web Content Accessibility Guidelines (WCAG); the current version is WCAG 2.0 (since 2008, http://www.w3.org/TR/WCAG20/). There are also other guidelines such as the Authoring Tool Accessibility Guidelines (ATAG) for developer tools and the User Agent Accessibility Guidelines (UAAG) for browsers. Other guidelines, such as EPUB3, exist for ebooks. Because WCAG 2.0 is the best-known, best-understood, and most-documented set of accessibility guidelines in the world, there is a companion guide, known as Guidance on Applying WCAG 2.0 to Non-Web Information and Communications Technologies (WCAG2ICT), for utilizing WCAG concepts in non-web technologies (Cunningham, 2012).

These concepts of digital accessibility are not new. The first version of WCAG came out in 1999, and captioning of video has existed for more than 30 years. The accessibility features are not technically hard to accomplish. WCAG requires, for instance, that all graphics have ALT text describing the image, that a webpage not have flashing that could trigger seizures, that tables and forms be marked up with appropriate labels (such as first name, last name, street address instead of FIELD1, FIELD2, FIELD3) to allow for identification. Another WCAG requirement is that all content on a page can be accessed even if you cannot use a pointing device through keyboard access. Creating accessible digital content is simply good coding, and it doesn't change, in any way, how information is visually presented.

Similar concepts apply for creating accessible word-processing documents, presentations, and PDF files—appropriate labeling and descriptions ensure that a document or presentation will be accessible. Multiple approaches for accomplishing a task allow for successful task completion for a diverse population of users. Even when properly utilizing guidelines such as WCAG 2.0, it is a good idea to evaluate for success by usability testing with people with disabilities, expert reviews, and automated accessibility testing.

The Web Content Accessibility Guidelines form the basis for many of the laws and regulations around the world. Section 508 of the Rehabilitation Act in the United States requires that when the federal government develops, procures, maintains, or uses electronic and information technology, that technology must be accessible for employees and members of the general public who have disabilities. This applies to procurement of both hardware and software technology as well as ensuring that websites are accessible (Lazar and Hochheiser, 2013; Lazar et al., 2015).

The Americans with Disabilities Act, as interpreted by federal courts and the U.S. Department of Justice, also requires accessibility of state and local government websites as well as those of private companies and organizations that are considered "public accommodations" (stores, museums, hotels, video rental, etc.). The U.S. Department of Justice is also enforcing accessibility of websites and instructional materials at universities. Lawsuits such as those against Target, Netflix, Harvard University, and MIT highlight the increasing importance and expectations of digital accessibility.

The European Union Mandate 376 (http://www.mandate376.eu/) will require procurement and development of accessible technologies by EU governments and will coordinate with U.S. Section 508, utilizing WCAG 2.0 and enabling developers to easily satisfy both U.S. and EU legal requirements. Prior to EU Mandate 376, many European countries, such as the UK, Italy, and Germany, and other countries around the world, including Australia and Canada, also had information technology accessibility requirements. The coverage (only government technology or also public accommodations), required reporting requirements, and penalties for noncompliance differ from country to country.

The United Nations Convention on the Rights of Persons with Disabilities (CRPD, http://www.un.org/disabilities/convention/conventionfull.shtml), an international human rights agreement, also addresses accessible technology. Article 9 of the CRPD calls upon countries to "Promote access for persons with disabilities to new information and communications technologies and systems, including the Internet," and article 21 encourages countries to "[provide] information intended for the general public to persons with disabilities in accessible formats and technologies appropriate to different kinds of disabilities."

Accessibility is a core feature of contemporary information systems, baked into development from the start. Programmers who follow coding standards and guidance from WCAG 2.0 add minimal cost in development yet provide valuable services to all users. By contrast, implementers who seek to retrofit for accessibility find that their effort is much greater (Wentz et al., 2011).

Increasingly, a person's economic success depends on equal access to digital content and services. University classes take place online, job postings are made online, and job applications must be submitted online. Prices are often lower when using a company website instead of calling the company on the phone. When people with disabilities have equal access to digital content and services, they have access to the full range of economic opportunities. The good news is that computer scientists, software engineers, developers, designers, and user experience professionals have the opportunity, through good design, appropriate coding standards, and proper testing and evaluation, to ensure equal access.

2.7 Older Adult Users

Seniority offers many pleasures and all the benefits of experience, but aging can also have negative physical, cognitive, and social consequences. Understanding the human factors of aging can help designers to create user interfaces that facilitate access by older adult users (Fig. 2.4). The benefits include improved chances for productive employment and opportunities to use writing, e-mail, and other computer tools plus the satisfactions of education, entertainment, social interaction, and challenge (Newell, 2011; Czaja and Lee, 2012). Older adults are particularly active participants in health support groups. The benefits to society include increased access to older adults, which is valuable for their experience and the emotional support they can provide to others.

FIGURE 2.4

HomeAssist is an assisted living platform for older adults installed in homes is Bordeaux, France. The tablet is used to show alerts (e.g., when the front door was left opened) and reminders but also to run a slide show of photographs when not in use (http://phoenix.inria.fr/research-projects/homeassist).

The National Research Council's report *Human Factors Research Needs for an Aging Population* describes aging as

> a nonuniform set of progressive changes in physiological and psychological functioning. . . . Average visual and auditory acuity decline considerably with age, as do average strength and speed of response. . . . [People experience] loss of at least some kinds of memory function, declines in perceptual flexibility, slowing of "stimulus encoding," and increased difficulty in the acquisition of complex mental skills, . . . visual functions such as static visual acuity, dark adaptation, accommodation, contrast sensitivity, and peripheral vision decline, on average, with age. (Czaja, 1990)

This list has its discouraging side, especially since older adults may have multiple impairments, but many older adults increasingly experience only moderate effects, allowing them to be active participants, even throughout their nineties.

The further good news is that interface designers can do much to accommodate older adult users (Chisnell et al., 2006). Improved user experiences give older adults access to the beneficial aspects of computing and network communication, thus bringing many societal advantages. How many young people's lives might be enriched by e-mail access to grandparents or great-grandparents? How many businesses might benefit from electronic consultations with experienced older adults? How many government agencies, universities, medical centers, or law firms could advance their goals from easily available contact with knowledgeable, older adult citizens? As a society, how might we all benefit from the continued creative work of older adults in literature, art, music, science, or philosophy?

As the world's population ages, designers in many fields are adapting their work to serve older adults, which can benefit all users. Baby boomers have already begun to push for larger street signs, brighter traffic lights, and better nighttime lighting to make driving safer for drivers and pedestrians. Similarly, desktop, web, and mobile devices can be improved for all users by providing users with control over font sizes, display contrast, and audio levels. Interfaces can also be designed with easier-to-use pointing devices, clearer navigation paths, and consistent layouts to improve access for older adults and every user (Hart et al., 2008; Czaja and Lee, 2012).

Considering older and disabled users during the design process often produces novel designs (Newell, 2011), such as ballpoint pens (for people with impaired dexterity), cassette tape recorders (for blind users to listen to audiobooks), and auto-completion software (to reduce keystrokes). Texting interfaces that suggest words or web-address completion were originally designed to ease data input for older and disabled users but have become expected conveniences or all users of mobile devices and web browsers. These conveniences, which reduce cognitive load, perceptual difficulty, and motor control demands, become vital in difficult environments, such as while traveling, injured, stressed, or under pressure for rapid correct completion. Similarly, subtitles (closed captioning) and user-controlled font sizes were designed for users with hearing and visual difficulties, but they benefit many users.

Researchers and designers are actively working on improving interfaces for older adults (Czaja and Lee, 2012). In the United States, the AARP's Older Wiser Wired initiatives provide education for older adults and guidance for designers. The European Union also has multiple initiatives and research support for computing for older adults.

Networking projects, such as the San Francisco–based SeniorNet, are providing adults over the age of 50 with access to and education about computing and the Internet "to enhance their lives and enable them to share their knowledge and wisdom" (http://www.seniornet.org/). Computer games are attractive for older adults, as shown by the surprising success of Nintendo's Wii, because they stimulate social interaction, provide practice in sensorimotor skills such as eye-to-hand coordination, enhance dexterity, and improve reaction time. In addition, meeting a challenge and gaining a sense of accomplishment and mastery are helpful in improving self-image for anyone.

In our experiences in bringing computing to two residences for older adults, we also encountered residents' fear of computers and belief that they were incapable of using computers. These fears gave way quickly after a few positive experiences. The older adults, who explored e-mail, photo sharing, and educational games, felt quite satisfied with themselves and were eager to learn more. Their newfound enthusiasm encouraged them to try automated bank machines and supermarket touchscreen kiosks. Suggestions for redesigns to meet the needs of older adults (and possibly other users) also emerged—for example, the appeal of high-precision touchscreens compared with the mouse was highlighted (Chapter 10).

In summary, making computing more attractive and accessible to older adults enables them to take advantage of technology, enables others to benefit from their participation, and can make technology easier for everyone. For more information on this topic, check out the Human Factors & Ergonomics Society (http://www.hfes.org), which has an Aging Technical Group that publishes a newsletter and organizes sessions at conferences.

2.8 Children

Another lively community of users is children, whose uses emphasize entertainment and education (Hourcade, 2015). Even pre-readers can use computer-controlled toys, music generators, and art tools. As they mature, begin reading, and gain limited keyboard skills, they can use a wider array of desktop applications, web services, and mobile devices (Foss and Druin, 2014). When they become teenagers, they may become highly proficient users who often help their parents or other adults. This idealized growth path is followed by many children who have easy access to technology and supportive parents and peers. However, many children without financial resources or supportive learning environments struggle to gain access to technology. They are often frustrated with its use and are endangered by threats surrounding privacy, alienation, pornography, unhelpful peers, and malevolent strangers.

The noble aspirations of designers of children's software include educational acceleration, facilitating socialization with peers, and fostering the self-confidence that comes from skill mastery (Fig. 2.5). Advocates of educational games promote intrinsic motivation and constructive activities as goals, but opponents often complain about the harmful effects of antisocial and violent games.

For teenagers, the opportunities for empowerment are substantial. They often take the lead in employing new modes of communication, such as text messaging on cellphones, and in creating cultural or fashion trends that surprise even the designers (for example, playing with simulations and fantasy games and participating in web-based virtual worlds).

Appropriate design principles for children's software recognize young people's intense desire for the kind of interactive engagement that gives them control with appropriate feedback and supports their social engagement with peers (Bruckman et al., 2012; Fails et al., 2014). Designers also have to find the balance between children's desire for challenge and parents' requirements for safety.

Children can deal with some frustrations and with threatening stories, but they also want to know that they can clear the screen, start over, and try again without severe penalties. They don't easily tolerate patronizing comments or

FIGURE 2.5

Using Digital Mysteries on a tablet, two elementary school children work together to read information slips, group them, and create a sequence to answer the question "Who killed King Ted?" The blue pop-up pie menu allows the selection of tools. A larger tabletop version allows larger groups to collaborate (http://www.reflectivethinking.com).

inappropriate humor, but they like familiar characters, exploratory environments, and the capacity for repetition. Younger children will sometimes replay a game, reread a story, or replay a music sequence dozens of times, even after adults have tired of it. While too much "screen time" can interfere with childhood development, well-designed applications can help children with physical, relationship, and emotional problems (Börjesson et al., 2015).

Some designers work by observing children and testing software with children, while the innovative approach of "children as our technology-design partners" engages them in a long-term process of cooperative inquiry during which children and adults jointly design novel products and services. A notable successful product of working with children as design partners is the International Children's Digital Library, which offers 4500-plus of the world's best children's books in 50-plus languages using an interface in 19 languages while supporting low-and high-speed networks.

Designing for younger children requires attention to their limitations. Their evolving dexterity means that mouse dragging, double-clicking, and small targets cannot always be used; their emerging literacy means that written instructions and error messages are not effective; and their low capacity for abstraction means that complex sequences must be avoided unless an adult is involved. Other concerns are short attention spans and limited capacity to work with multiple concepts simultaneously. Designers of children's software also have a responsibility to attend to dangers, especially in web-based environments, where parental control over access to violent, racist, or pornographic materials is unfortunately necessary. Appropriate information for the education of children about privacy issues and threats from strangers is also a requirement.

The capacity for playful creativity in art, music, and writing and the value of educational activities in science and math remain potent reasons to pursue children's software. Enabling them to make high-quality images, photos, songs, or poems and then share them with friends and family can accelerate children's personal and social development. Offering access to educational materials from libraries, museums, government agencies, schools, and commercial sources enriches their learning experiences and serves as a basis for children to construct their own web resources, participate in collaborative efforts, and contribute to community-service projects.

Providing programming tools, such as the Scratch project (https://scratch.mit.edu/), and simulation-building tools enables older children to take on complex cognitive challenges and construct ambitious artifacts for others to use. These and other opportunities have motivated efforts (such as One Laptop Per Child, http://one.laptop.org/) to bring low-cost computers to children around the world. Advocates point to enthusiastic adoption and tell stories of individual enablement. However, critics encourage a shift from the technology-centered goals to greater attention to rich content, social engagement, parental guidance materials, and effective teacher training.

2.9 Accommodating Hardware and Software Diversity

In addition to accommodating different classes of users and skill levels, designers need to support a wide range of hardware and software platforms. The rapid progress of technology means that newer systems may have a hundred or a thousand times greater storage capacity, faster processors, and higher-bandwidth networks. However, designers need to accommodate older devices and deal with newer mobile devices that may have low-bandwidth connections and small screens (Fig. 2.2).

The challenge of accommodating diverse hardware is coupled with the need to ensure access through many generations of software. New operating systems, web browsers, e-mail clients, and application programs should provide backward compatibility in terms of their user-interface design and file structures. Skeptics will say that this requirement can slow innovation, but designers who plan ahead carefully to support flexible interfaces and self-defining files will be rewarded with larger market shares.

For at least the next decade, three of the main technical challenges will be:

- *Producing satisfying and effective Internet interaction on high-speed (broadband) and slower (dial-up and some wireless) connections.* Some technological breakthroughs have already been made in compression algorithms to reduce file sizes for images, music, animations, and even video, but more are needed. New technologies are needed to enable pre-fetching or scheduled downloads. User control of the amount of material downloaded for each request could also prove beneficial (for example, allowing users to specify that a large image should be reduced to a smaller size, sent with fewer colors, converted to a simplified line drawing, replaced with just a text description, or downloaded at night when Internet charges are perhaps lower).
- *Responsive design enabling access to web services from large displays (3200 × 2400 pixels or larger) and smaller mobile devices (1024 × 768 pixels and smaller).* Rewriting each webpage for different display sizes may produce the best quality, but this approach is probably too costly and time-consuming for most web providers. Software tools such as Cascading Style Sheets (CSS) allow designers to specify their content in a way that enables automatic conversions for an increasing range of display sizes.
- *Supporting easy maintenance of or automatic conversion to multiple languages.* Commercial operators recognize that they can expand their markets if they can provide access in multiple languages and across multiple countries. This means isolating text to allow easy substitution, choosing appropriate

metaphors and colors, and addressing the needs of diverse cultures (Section 2.5).

Practitioner's Summary

The good news is that when designers think carefully about the needs of diverse users, they are likely to come up with desktop, laptop, web, and mobile device designs that are better for all users. A frequent path to success is through participatory methods that bring designers in close and continuing contact with their intended users. In some cases, improved tools and designs mean that one design can be made so flexible that it can be presented automatically in text (with a wide range of font sizes, colors, and contrast ratios), in speech (with male or female styles and at varying volumes and speeds), and in a wide range of display sizes. Adjustments for different cultures, personalities, disabilities, ages, input devices, and preferences may take more design effort, but the payoffs are in larger markets and more satisfied users. As for costs, with appropriate software tools, e-commerce providers are finding that a small additional effort can expand markets by 20% or more. Although it can require additional effort, designing for diverse users is cost effective and sometimes leads to major breakthroughs.

Researcher's Agenda

While market forces provide incentives for changes, additional legal and policy interventions could speed progress in ensuring that desktop, laptop, web, and mobile device user interfaces continue to be accessible to all. The expanding worldwide research community, especially the ACM Special Interest Group on Accessible Computing (SIGACCESS), hosts international conferences, publishes journals, and encourages further research.

Research on diversity often brings innovations for all users; for example, input devices for users with poor motor control can often help all passengers in rough riding cars, buses, trains, or planes. Improved automated assistance for conversions to diverse languages and cultures would improve designer productivity and facilitate changes to prices, dimensions, colors, and so on. Research on cultural diversity is still needed about the acceptability by differing user groups of novel features like emoticons, animation, personalization, gamification, and musical accompaniments.

WORLD WIDE WEB RESOURCES

www.pearsonglobaleditions.com/shneiderman

Major suppliers offer diverse accessibility tools:

- Apple: https://www.apple.com/accessibility/
- Microsoft: http://www.microsoft.com/enable/
- Google: https://www.google.com/accessibility/

And many consumer-oriented and government groups provide assistance, such as:

- AARP: http://www.aarp.org/home-family/personal-technology/
- Older Adults Technology Services: http://oats.org/
- U.S. Section 508: http://www.section508.gov/
- Resource list from Trace Center: http://trace.wisc.edu/resources/

Discussion Questions

1. Describe three populations of users with special needs. For each of these populations, suggest three ways current interfaces could be improved to better serve them.
2. Suppose you need to design a system for users in two countries that are very different from each other culturally. What are some of the design concerns that you should be aware of to create a successful design?
3. In certain interfaces, it is necessary to inform users of an abnormal condition or time-dependent information. It is important that the display of this information catches the user's attention. Suggest five ways a designer can successfully attract attention.
4. Name a piece of software you often use where it is easy to produce an error. Explain ways you could improve the interface to better prevent errors.
5. What factors should designers consider to address the needs of individuals with different physical abilities?

References

Blanck, P., *eQuality: The Struggle for Web Accessibility by Persons with Cognitive Disabilities*, Cambridge University Press (2014).

Börjesson, P., Barendregt, W., Eriksson, E., and Torgersson, O., Designing technology for and with developmentally diverse children: A systematic literature review, *Proceedings of ACM SIGCHI Interaction Design and Children Conference,* ACM Press, New York (2015), 79–88.

Bruckman, Amy, Bandlow, Alisa, Dimond, Jill, and Forte, Andrea, Human-computer interaction for kids, in Jacko, Julie (Editor), *The Human-Computer Interaction Handbook,* 3rd Edition, CRC Press (2012), 841–862.

Center for Information Technology Accommodation, Section 508: The road to accessibility, General Services Administration, Washington, DC (2015). Available at http://www.section508.gov/.

Chisnell, Dana E., Redish, Janice C., and Lee, Amy, New heuristics for understanding older adults as web users, *Technical Communications 53*, 1 (February 2006), 39–59.

Chourasia, A., Nordstrom, D., and Vanderheiden, G., State of the science on the cloud, accessibility, and the future, *Universal Access in the Information Society 13*, 4 (2014), 483–495.

Cunningham, Katie, *Accessibility Handbook,* O'Reilly Publishing (2012).

Czaja, S. J. (Editor), *Human Factors Research Needs for an Aging Population*, National Academy Press, Washington, DC (1990).

Czaja, S. J., and Lee, C. C., Older adults and information technology: Opportunities and challenges, in Jacko, Julie (Editor), *The Human-Computer Interaction Handbook,* 3rd Edition, CRC Press (2012), 825–840.

Fails, J. A., Guha, M. L., and Druin, A., Methods and techniques for involving children in the design of new technology, *Foundations and Trends in Human-Computer Interaction 6*, 2, Now Publishers Inc., Hanover (2014), 85–166.

Foss, E., and Druin, A., *Children's Internet Search: Using Roles to Understand Youth Search Behavior*, Morgan & Claypool Publishers (2014).

Hart, T. A., Chaparro, B. S., and Halcomb, C. G., Evaluating websites for older adults: adherence to "senior-friendly" guidelines and end-user performance, *Behavior & Information Technology 27,* 3 (May 2008), 191–199.

Horton, Sarah, and Quesenbery, Whitney, *A Web for Everyone: Designing Accessible User Experiences,* Rosenfeld Media (2014).

Hourcade, J. P., *Child-Computer Interaction,* CreateSpace Independent Publishing (2015). Available at http://homepage.divms.uiowa.edu/~hourcade/book/index.php.

Human Factors & Ergonomics Society, *ANSI/HFES 100–2007 Human Factors Engineering of Computer Workstations*, Santa Monica, CA (2007).

Lazar, Jonathan, Goldstein, Daniel F., and Taylor, Anne, *Ensuring Digital Accessibility through Process and Policy*, Morgan Kaufmann (2015).

Lazar, J., and Hochheiser, H., Legal aspects of interface accessibility in the U.S., *Communications of the ACM 56*, 12 (2013), 74–80.

Marcus, Aaron, and Gould, Emile W., Globalization, localization and cross-cultural user-interface design, in Jacko, Julie (Editor), *The Human-Computer Interaction Handbook,* 3rd Edition, CRC Press (2012), 341–366.

Medhi, I., Patnaik, S., Brunskill, E., Gautama, N., Thies, W., and Toyama, K., Designing mobile interfaces for novice and low-literacy users, *ACM Transactions on Computer-Human Interaction 18*, 1 (2011), Article 2, 28 pages.

Newell, Alan, *Design and the Digital Divide: Insights from 40 Years in Computer Support for Older and Disabled People*, Synthesis Lectures on Assistive, Rehabilitative, and Health-Preserving Technologies (Ron Baecker, Editor), Morgan & Claypool Publishers (2011).

Pereira, Roberto, and Baranauskas, Maria C. C., A value-oriented and culturally informed approach to the design of interactive systems, *International Journal of Human-Computer Systems 80* (2015), 66–82.

Preedy, V. R. (Editor), *Handbook of Anthropometry: Handbook of Human Physical Form in Health and Disease*, Springer Publishers (2012).

Quesenbery, Whitney, and Szuc, Daniel, *Global UX: Design and Research in a Connected World*, Morgan Kaufmann (2011).

Radvansky, Gabriel A., and Ashcraft, Mark H., *Cognition,* 6th Edition, Pearson (2013).

Reinecke, Katharina, and Bernstein, Abraham, Knowing what a user likes: A design science approach to interfaces that automatically adapt to culture, *MIS Quarterly 37,* 2 (2013), 427–453.

Salgado, L. C. C., Leitão, C. F., and de Souza, C.S., *A Journey through Cultures: Metaphors for Guiding the Design of Cross-Cultural Interactive Systems*, Springer (2012).

Sun, Huatong, *Cross-Cultural Technology Design*, Oxford University Press (2012).

Ware, Colin, *Information Visualization: Perception for Design,* 3rd Edition, Morgan Kaufmann Publ., San Francisco, CA (2012).

Wentz, B., Jaeger, P., and Lazar, J., Retrofitting accessibility: The inequality of after-the-fact access for persons with disabilities in the United States, *First Monday 16*, 11 (2011).

Wiggins, J. S., *The Five-Factor Model of Personality: Theoretical Perspectives,* Guilford Press (1996).

This page intentionally left blank


MARKER BEACON
ALT
CLIMB

CHAPTER 3

Guidelines, Principles, and Theories

> We want principles, not only developed—the work of the closet—but applied, which is the work of life.
>
> **Horace Mann**
> *Thoughts,* 1867

> There never comes a point where a theory can be said to be true. The most that anyone can claim for any theory is that it has shared the successes of all its rivals and that it has passed at least one test which they have failed.
>
> **A. J. Ayer**
> *Philosophy in the Twentieth Century,* 1982

CHAPTER OUTLINE

3.1 Introduction

User-interface designers have accumulated a wealth of experience and researchers have produced a growing body of empirical evidence and theories, all of which can be organized into:

1. *Guidelines*. Low-level focused advice about good practices and cautions against dangers.
2. *Principles*. Middle-level strategies or rules to analyze and compare design alternatives.
3. *Theories*. High-level widely applicable frameworks to draw on during design and evaluation as well as to support communication and teaching. Theories can also be predictive, such as those for pointing times by individuals or posting rates for community discussions.

In many contemporary systems, designers have a grand opportunity to improve the user interface by applying established guidelines to clean up cluttered displays, inconsistent layouts, and unnecessary text. These sources of debilitating stress and frustration can lead to poorer performance, minor slips, and serious errors, all contributing to job dissatisfaction and consumer resistance.

Guidelines, principles, and theories, which offer preventive medicine and remedies for these problems, have matured in recent years (Grudin, 2012). Reliable methods for predicting pointing and input times (Chapter 10), better social persuasion principles (Chapter 11), and helpful cognitive or perceptual theories (Chapter 13) now shape research and guide design. International or national standards, which could be described as commonly accepted and precisely defined so as to be enforceable, are increasingly influential (Carroll, 2014).

This chapter begins with a sampling of guidelines for navigating, organizing displays, getting user attention, and facilitating data entry (Section 3.2). Then Section 3.3 covers some fundamental principles of interface design, such as coping with user skill levels, task profiles, and interaction styles. It presents the Eight Golden Rules of Interface Design, explores ways of preventing user errors, and closes with a section on ensuring human control while increasing automation. Section 3.4 reviews micro-HCI and macro-HCI theories of interface design.

3.2 Guidelines

From the earliest days of computing, interface designers have written down guidelines to record their insights and to try to guide the efforts of future designers. The early Apple and Microsoft guidelines, which were influential for

FIGURE 3.1
Example of Apple guidelines for designing menus for the iWatch.

desktop-interface designers, have been followed by dozens of guidelines documents for the web and mobile devices (Fig. 3.1) (see the list at the end of Chapter 1). A guidelines document helps by developing a shared language and then promoting consistency among multiple designers in terminology usage, appearance, and action sequences. It records best practices derived from practical experience or empirical studies, with appropriate examples and counterexamples. The creation of a guidelines document engages the design community in lively discussions about input and output formats, action sequences, terminology, and hardware devices (Lynch and Horton, 2008; Hartson and Pyla, 2012; Johnson, 2014).

Critics complain that guidelines can be too specific, incomplete, hard to apply, and sometimes wrong. Proponents argue that building on experience from design leaders contributes to steady improvements. Both groups recognize the value of lively discussions in promoting awareness.

The following four sections provide examples of guidelines, and Section 4.3 discusses how they can be integrated into the design process. The examples address some key topics, but they merely sample the thousands of guidelines that have been written.

3.2.1 Navigating the interface

Since navigation can be difficult for many users, providing clear rules is helpful. The sample guidelines presented here come from the U.S. government's efforts to promote the design of informative webpages (National Cancer Institute, 2006), but these guidelines have widespread application. Most are stated positively ("reduce the user's workload"), but some are negative ("do not display unsolicited windows or graphics"). The 388 guidelines, which offer cogent examples and impressive research support, cover the design process, general principles, and specific rules. This sample of the guidelines gives useful advice and a taste of their style:

> *Standardize task sequences.* Allow users to perform tasks in the same sequence and manner across similar conditions.
>
> *Ensure that links are descriptive.* When using links, the link text should accurately describe the link's destination.
>
> *Use unique and descriptive headings.* Use headings that are distinct from one another and conceptually related to the content they describe.
>
> *Use radio buttons for mutually exclusive choices.* Provide a radio button control when users need to choose one response from a list of mutually exclusive options.
>
> *Develop pages that will print properly.* If users are likely to print one or more pages, develop pages with widths that print properly.
>
> *Use thumbnail images to preview larger images.* When viewing full-size images is not critical, first provide a thumbnail of the image.

Guidelines to promote accessibility for users with disabilities were included in the U.S. Rehabilitation Act. Its Section 508, with guidelines for web design, is published by the Access Board (http://www.access-board.gov/508.htm), an independent U.S. government agency devoted to accessibility for people with disabilities. The World Wide Web Consortium (W3C) adapted these guidelines (http://www.w3.org/TR/WCAG20/) and organized them into three priority levels, for which it has provided automated checking tools. A few of the accessibility guidelines are:

> *Text alternatives*. Provide text alternatives for any non-text content so that it can be changed into other forms people need, such as large print, braille, speech, symbols, or simpler language.
>
> *Time-based media*. Provide alternatives for time-based media (e.g., movies or animations). Synchronize equivalent alternatives (such as captions or auditory descriptions of the visual track) with the presentation.
>
> *Distinguishable.* Make it easier for users to see and hear content, including separating foreground from background. Color is not used as the only visual means of conveying information, indicating an action, prompting a response, or distinguishing a visual element.
>
> *Predictable*. Make Web pages appear and operate in predictable ways.

The goal of these guidelines is to have webpage designers use features that permit users with disabilities to employ screen readers or other special technologies to give them access to webpage content.

3.2.2 Organizing the display

Display design is a large topic with many special cases. An early influential guidelines document (Smith and Mosier, 1986) offers five high-level goals for data display:

1. *Consistency of data display.* During the design process, the terminology, abbreviations, formats, colors, capitalization, and so on should all be standardized and controlled by use of a dictionary of these items.
2. *Efficient information assimilation by the user.* The format should be familiar to the operator and should be related to the tasks required to be performed with the data. This objective is served by rules for neat columns of data, left justification for alphanumeric data, right justification of integers, lining up of decimal points, proper spacing, use of comprehensible labels, and appropriate measurement units and numbers of decimal digits.
3. *Minimal memory load on the user.* Users should not be required to remember information from one screen for use on another screen. Tasks should be arranged such that completion occurs with few actions, minimizing the chance of forgetting to perform a step. Labels and common formats should be provided for novice or intermittent users.
4. *Compatibility of data display with data entry.* The format of displayed information should be linked clearly to the format of the data entry. Where possible and appropriate, the output fields should also act as editable input fields.
5. *Flexibility for user control of data display.* Users should be able to get the information from the display in the form most convenient for the task on which they are working. For example, the order of columns and sorting of rows should be easily changeable by the users.

This compact set of high-level objectives is a useful starting point, but each project needs to expand these into application-specific and hardware-dependent standards and practices.

3.2.3 Getting the user's attention

Since substantial information may be presented to users, exceptional conditions or time-dependent information must be presented so as to attract attention (Wickens et al., 2012). These guidelines detail several techniques for getting the user's attention:

- *Intensity.* Use two levels only, with limited use of high intensity to draw attention.
- *Marking.* Underline the item, enclose it in a box, point to it with an arrow, or use an indicator such as an asterisk, bullet, dash, plus sign, or X.
- *Size.* Use up to four sizes, with larger sizes attracting more attention.
- *Choice of fonts.* Use up to three fonts.

- *Blinking.* Use blinking displays (2–4 Hz) or blinking color changes with great care and in limited areas, as it is distracting and can trigger seizures.
- *Color.* Use up to four standard colors, with additional colors reserved for occasional use.
- *Audio.* Use soft tones for regular positive feedback and harsh sounds for rare emergency conditions.

A few words of caution are necessary. There is a danger of creating cluttered displays by overusing these techniques. Some web designers use blinking advertisements or animated icons to attract attention, but users almost universally disapprove. Animation is appreciated primarily when it provides meaningful information, such as for a progress indicator or to show movement of files.

Novices need simple, logically organized, and well-labeled displays that guide their actions. Expert users prefer limited labels on fields so data values are easier to extract; subtle highlighting of changed values or positional presentation is sufficient. Display formats must be tested with users for comprehensibility.

Similarly highlighted items will be perceived as being related. Color-coding is especially powerful in linking related items, but this use makes it more difficult to cluster items across color codes (Section 12.5). User control over highlighting is much appreciated, for example, allowing cellphone users to select the color for contacts that are close family members or for meetings that are of high importance.

Audio tones, like the clicks in keyboards or cellphone ring tones, can provide informative feedback about progress. Alarms for emergency conditions do alert users rapidly, but a mechanism to suppress alarms must be provided. If several types of alarms are used, testing is necessary to ensure that users can distinguish between the alarm levels. Prerecorded or synthesized voice messages are a useful alternative, but since they may interfere with communications between operators, they should be used cautiously (Section 9.3).

3.2.4 Facilitating data entry

Data-entry tasks can occupy a substantial fraction of users' time and can be the source of frustrating and potentially dangerous errors. Smith and Mosier (1986) offer five high-level objectives as part of their guidelines for data entry (Courtesy of MITRE Corporate Archives: Bedford, MA):

1. *Consistency of data-entry transactions.* Similar sequences of actions speed learning.
2. *Minimal input actions by user*. Fewer input actions mean greater operator productivity and—usually—fewer chances for error. Making a choice by a single mouse selection or finger press, is preferred over typing in a lengthy string of characters. Selecting from a list of choices eliminates

the need for memorization, structures the decision-making task, and eliminates the possibility of typographic errors.

A second aspect of this guideline is that redundant data entry should be avoided. It is annoying for users to enter the same information in two locations, such as entering the billing and shipping addresses when they are the same. Duplicate entry is perceived as a waste of effort and an opportunity for error.

3. *Minimal memory load on users*. When doing data entry, users should not be required to remember lengthy lists of codes.
4. *Compatibility of data entry with data display*. The format of data-entry information should be linked closely to the format of displayed information, such as dashes in telephone numbers.
5. *Flexibility for user control of data entry*. Experienced users prefer to enter information in a sequence that they can control, such as selecting the color first or size first, when clothes shopping.

Guidelines documents are a wonderful starting point to give designers the benefit of experience (Fig. 3.2), but they will always need processes to facilitate education, enforcement, exemption, and enhancement (Section 4.3).

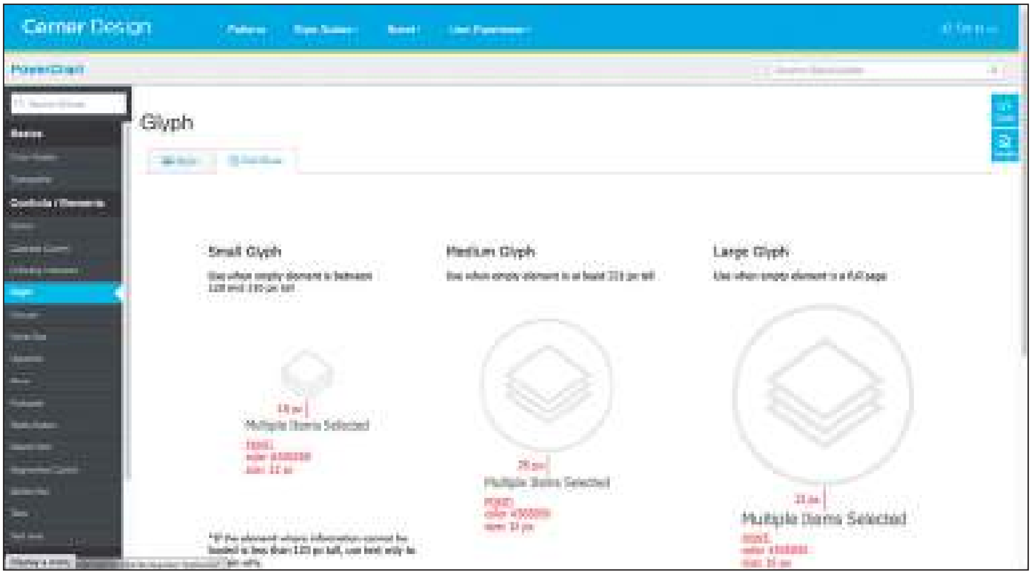


FIGURE 3.2

The guidelines website for Cerner designers and developers. This particular guideline describes the three sizes of icons, or glyphs, that should be used in all electronic health record products (each consisting of hundreds of screens). The 100-plus programmers who develop the user interfaces can also access code samples. Each guideline includes references (http://design.cerner.com).

3.3 Principles

While guidelines are low-level and narrowly focused, principles are more fundamental, widely applicable, and enduring. However, they also tend to need more clarification. For example, the principle of recognizing user diversity makes sense to every designer, but it must be thoughtfully interpreted. A preschooler playing a computer game is a long way from a legal librarian searching for precedents for anxious and hurried lawyers. Similarly, a grandmother sending a text message is a long way from a highly trained and experienced air-traffic controller. These examples highlight the differences in users' background knowledge, frequency of use, and goals as well as in the impact of user errors. Since no single design is ideal for all users and situations, designers who characterize their users and the context of use are more likely to produce successful products.

Chapter 2 introduced the individual differences that designers must address in pursuing universal usability. This section focuses on a few fundamental principles, beginning with accommodating user skill levels and profiling tasks and user needs. It discusses the five primary interaction styles (direct manipulation, menu selection, form fill-in, natural language, and command language) and the Eight Golden Rules of Interface Design, followed by a section on error prevention. Finally, it covers controversial strategies for ensuring human control while increasing automation.

3.3.1 Determine users' skill levels

Learning about the users is a simple idea but a difficult and, unfortunately, often undervalued goal. No one would argue against this principle, but many designers simply assume that they understand the users and the users' tasks. Successful designers are aware that people learn, think, and solve problems in different ways. Some users prefer to deal with tables rather than graphs, with words instead of numbers, or with rigid structures rather than open-ended forms.

All design should begin with an understanding of the intended users, including population profiles that reflect their age, gender, physical and cognitive abilities, education, cultural or ethnic backgrounds, training, motivation, goals, and personality. There are often several communities of users for an interface, especially for web applications and mobile devices, so the design effort is multiplied. Typical user personas—such as nurses, doctors, storekeepers, high school students, or children—can be expected to have various combinations of knowledge and usage patterns. User groups from different countries may each deserve special attention, and regional differences often exist within countries. Other variables that characterize user personas include location (for example, urban versus rural), economic profile, disabilities, and attitudes toward using technology. Users with poor reading skills, limited education, and low motivation require special attention.

In addition to these personas, an understanding of users' skills with interfaces and with the application domain is important. Users might be tested for their familiarity with interface features, such as traversing hierarchical menus or drawing tools. Other tests might cover domain-specific abilities, such as knowledge of airport city codes, stockbrokerage terminology, or map icons.

The process of getting to know the users is never-ending because there is so much to know and because the users keep changing. However, every step toward understanding the users and recognizing them as individuals with outlooks different from the designer's own is likely to be a step closer to a successful design.

For example, a generic separation into novice or first-time, knowledgeable intermittent, and expert frequent users might lead to these differing design goals:

1. *Novice or first-time users.* True novice users—for example, bank customers making their first cellphone check deposit—are assumed to know little of the task or interface concepts. By contrast, first-time users are often professionals who know the task concepts well but have shallow knowledge of the interface concepts (for example, a business traveler using a new rental car's navigation system). Overcoming their uncertainties, via instructions or dialog boxes, is a serious challenge to interface designers. Restricting vocabulary to a small number of familiar, consistently used concept terms is essential. The number of actions should also be small so that novice and first-time users can carry out simple tasks successfully, which reduces anxiety, builds confidence, and provides positive reinforcement. Informative feedback about the accomplishment of each task is helpful, and constructive, specific error messages should be provided when users make mistakes. Carefully designed video demonstrations and online tutorials may be effective.

2. *Knowledgeable intermittent users*. Many people are knowledgeable but intermittent users of a variety of systems (for example, business travelers filing for travel reimbursements). They have stable task concepts and broad knowledge of interface concepts, but they may have difficulty retaining the structure of menus or the location of features. The burden on their memories will be lightened by orderly structure in the menus, consistent terminology, and interfaces that emphasize recognition rather than recall. These features will also help novices and some experts, but the major beneficiaries are knowledgeable intermittent users. Protection from danger is necessary to support relaxed exploration of features and usage of partially forgotten action sequences. These users will benefit from context-dependent help to fill in missing pieces of task or interface knowledge.

3. *Expert frequent users.* Expert "power" users are thoroughly familiar with the task and interface concepts and seek to get their work done quickly. They demand rapid response times, brief and non-distracting feedback, and the shortcuts to carry out actions with just a few clicks or gestures.

Designing for one class of users is relatively easy; designing for several is much more difficult. When multiple user classes must be accommodated, the basic strategy is to permit a *multi-layer* (sometimes called *level-structured* or *spiral*) approach to learning. Novices can be taught a minimal subset of objects and actions with which to get started. They are most likely to make correct choices when they have only a few options and are protected from making mistakes—that is, when they are given a *training-wheels* interface. After gaining confidence from hands-on experience, these users can choose to progress to ever-greater levels of task concepts and the accompanying interface concepts. The learning plan should be governed by the users' progress through the task concepts, enabling users to take on new interface concepts when they are needed to support more complex tasks. For users with strong knowledge of the task and interface concepts, rapid progress is possible.

For example, novice users of a cellphone can quickly learn to make/receive calls first, then to use the menus to change ring tones, and later to reset the privacy protections. The multi-layer approach helps users with different skill levels and promotes universal usability (Shneiderman, 2003).

Another option for accommodating different user classes is to permit users to personalize the menu contents. A third option is to permit users to control the density of informative feedback that the user interface provides. Novices want more informative feedback to confirm their actions, whereas frequent users want less distracting feedback. Similarly, frequent users like displays to be more densely packed than do novices. Finally, the pace of interaction may be varied from slow for novices to fast for frequent users.

3.3.2 Identify the tasks

After carefully drawing the user profile, the designers identify the tasks to be carried out. Every designer would agree that the tasks must be identified first, but too often, the task analysis is done informally or incompletely. Task analysis has a long history (Hackos and Redish, 1998; Wickens et al., 2012), but successful strategies usually involve long hours of observing and interviewing users. This helps designers to understand task frequencies and sequences and make the tough decisions about what tasks to support. Some implementers prefer to include all possible actions in the hope that some users will find them helpful, but this causes clutter. However, mobile app designers are successful because they ruthlessly limited functionality (for example, calendar, contacts, and to-do list) to guarantee simplicity.

High-level task actions can be decomposed into multiple middle-level task actions, which can be further refined into atomic actions that users execute with a single menu selection or other action. Choosing the most appropriate set of atomic actions is a difficult task. If the atomic actions are too small, the users will become frustrated by the large number of actions necessary to accomplish a higher-level task. If the atomic actions are too large and elaborate, the users will need special options to get what they want from the user interface.

The relative task frequencies are important in shaping a menu tree. Frequent tasks should be near the top and therefore quick to carry out, while rare tasks are deeper down. Relative frequency of use is one of the bases for making architectural design decisions. For example, in a text editor:

- Frequent actions might be performed by pressing special keys, such as the four arrow keys, Insert, and Delete.
- Less frequent actions might be performed by pressing a single letter plus the Ctrl key or by a selection from a pull-down menu—examples include underscore, bold, and save.
- Infrequent actions or complex actions might require going through a sequence of menu selections or form fill-ins—for example, to change the margins or to revise default printers.

Similarly, cellphone users can assign their close friends and family members to speed-dial numbers so that frequent calls can be made easily by pressing a single key

Creating a matrix of users and tasks can help designers sort out these issues (Fig. 3.3). In each box, the designer can put a check mark to indicate that this user carries out this task. A more precise analysis would include frequencies instead of just simple check marks. Such user-needs assessment clarifies what tasks are essential for the design and which ones could be left out to preserve system simplicity and ease of learning.

3.3.3 Choose an interaction style

When the task analysis is complete and the task objects and actions have been identified, the designer can choose from these five primary interaction styles: direct manipulation, menu selection, form fill-in, command language, and natural language (Box 3.1 and Box 3.2). Chapters 7 through 9 explore these styles in detail; this summary gives a brief comparative overview.

Direct manipulation When designers can create a visual representation of the world of action, the users' tasks can be greatly simplified because direct manipulation of familiar objects is possible. Examples of such visual and tangible user interfaces include the desktop metaphor, drawing tools, photo editing, and games. By pointing at visual representations of objects and actions, users can

	TASK				
Job Title	**Query by Patient**	**Update Data**	**Query across Patients**	**Add Relations**	**Evaluate System**
Nurse	**	**			
Physician	**	*			
Supervisor	*	*	**		
Appointment personnel	****				
Medical-record maintainer	**	**	*	*	
Clinical researcher			***		*
Database programmer		*	**	**	*

FIGURE 3.3

Frequency of Task by Job Title

Hypothetical frequency-of-use of data for a medical clinic information system. Answering queries from appointment personnel about individual patients is the highest-frequency task (****), and lower-frequency use is shown with ***, **, or *.

carry out tasks rapidly and can observe the results immediately (for example, dragging and dropping an icon into a trash can). Context-aware, embedded, natural, and wearable user interfaces often extend the capacity of direct manipulation designs by allowing users to gesture, point, move, or even dance to achieve their goals. Direct manipulation is appealing to novices, is easy to remember for intermittent users, and, with careful design, can be rapid for frequent users. Chapter 7 describes direct manipulation and its application.

Navigation and menu selection In navigation and menu-selection systems, users review choices, select the one most appropriate to their task, and observe the effect. If the terminology and the meaning of the items are understandable and distinct, users can accomplish their tasks with little learning or memorization and just a few actions. The greatest benefit may be that there is a clear structure to decision making, since all possible choices are presented at one time. This interaction style is appropriate for novice and intermittent users and can be appealing to frequent users if the display and selection mechanisms are rapid. For designers, menu-selection systems require careful task analysis to ensure that all functions are supported conveniently and that terminology is chosen carefully and used consistently. User-interface–building tools that support menu selection provide an enormous benefit by ensuring consistent screen design, validating completeness, and supporting maintenance. Navigation and menu selection is discussed in Chapter 8.

BOX 3.1

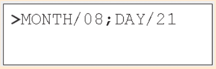
Advantages and disadvantages of the five primary interaction styles.

Advantages	Disadvantages
Direct manipulation	
• Visually presents task concepts • Allows easy learning • Allows easy retention • Allows errors to be avoided • Encourages exploration • Affords high subjective satisfaction	• May be hard to program • Accessibility requires special attention
Navigation and menu selection	
• Shortens learning • Reduces keystrokes • Structures decision making • Permits use of dialog-management tools • Allows easy support of error handling	• Presents danger of many menus • May slow frequent users • Consumes screen space • Requires rapid display rate
Form fill-in	
• Simplifies data entry • Enables convenient assistance • Permits use of form-management tools	• Consumes screen space
Command language	
• Powerful • Allows easy scripting and history keeping	• Requires learning and retention • Error-prone
Natural language	
• Relieves burden of learning syntax	• Requires clarification dialog • May not show context • May require more keystrokes • Unpredictable

BOX 3.2

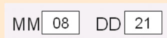
Spectrum of directness.

An example of progression toward more direct manipulation: less recall/more recognition, fewer keystrokes/fewer clicks, less capability to make errors, and more visible context.

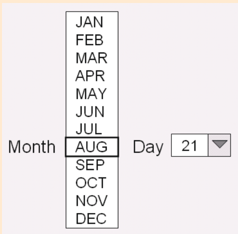


a. Command line

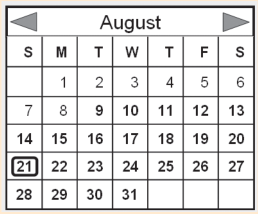
b. Form fill-in to reduce typing



c. Improved form fill-in to clarify and reduce errors



d. Pull-down menus offer meaningful names and eliminate invalid values



e. 2-D menus to provide context, show valid dates, and enable rapid single selection

Form fill-in When data entry is required, menu selection alone usually becomes cumbersome, and form fill-in (also called *fill in the blanks*) is appropriate. Users see a display of related fields, move a cursor among the fields, and enter data where desired. With the form fill-in interaction style, users must understand the field labels, know the permissible values and the data-entry method, and be capable of responding to error messages. Since knowledge of the keyboard, labels, and permissible fields is required, some training may be necessary. This interaction style is most appropriate for knowledgeable intermittent users or frequent users. Chapter 8 provides a thorough treatment of form fill-in.

Command language For frequent users, command languages (discussed in Section 9.4) provide a strong feeling of being in control. Users learn the syntax and can often express complex possibilities rapidly without having to read distracting prompts. However, error rates are typically high, training is necessary, and retention may be poor. Error messages and online assistance are hard to provide because of the diversity of possibilities. Command languages and query or programming languages are the domain of expert frequent users, who often derive great satisfaction from mastering a complex language. Powerful advantages include easy scripting and history keeping.

Natural language Increasingly, user interfaces respond properly to arbitrary spoken (for example, Siri on the Apple iPhone) or typed natural-language statements (for example, web search phrases). Speech recognition can be helpful with familiar phrases such as "Tell Catherine that I'll be there in ten minutes," but with novel situations users may be frustrated with the results (discussed in Chapter 9).

Blending several interaction styles may be appropriate when the required tasks and users are diverse. For example, a form fill-in interface for shopping checkout can include menus for items such as accepted credit cards, and a direct-manipulation environment can allow a right-click that produces a pop-up menu with color choices. Also, keyboard commands can provide shortcuts for experts who seek more rapid performance than mouse selection.

Increasingly, these five interaction styles are complemented by using context, sensors, gestures, spoken commands, and going beyond the screen to include enriched environments that enable users to activate doors, change sound volume, or turn on faucets. These enriched environments, such as those found in automobiles, game arcades, projected displays, wearable interfaces, musical instruments, and sound spaces, go beyond the desktop and mobile devices to produce playful and useful effects. The expansion of user interfaces into clothing, furniture, buildings, implanted medical devices, mobile platforms such as drones, and the Internet of Things enriches traditional strategies and expands the design possibilities.

Chapters 7–9 expand on the constructive guidance for using the different interaction styles outlined here, and Chapter 10 describes how input and output devices influence these interaction styles. Chapter 11 deals with interaction when using collaborative interfaces and participating in social media.

3.3.4 The Eight Golden Rules of Interface Design

This section focuses attention on eight principles, called "golden rules," that are applicable in most interactive systems and enriched environments. These principles, derived from experience and refined over three decades, require validation and tuning for specific design domains. No list such as this can be complete, but it has been well received as a useful guide to students and designers. The Eight Golden Rules are:

1. *Strive for consistency.* Consistent sequences of actions should be required in similar situations; identical terminology should be used in prompts, menus, and help screens; and consistent color, layout, capitalization, fonts, and so on, should be employed throughout. Exceptions, such as required confirmation of the delete command or no echoing of passwords, should be comprehensible and limited in number.
2. *Seek universal usability.* Recognize the needs of diverse users and design for *plasticity*, facilitating transformation of content. Novice to expert differences,

age ranges, disabilities, international variations, and technological diversity each enrich the spectrum of requirements that guides design. Adding features for novices, such as explanations, and features for experts, such as shortcuts and faster pacing, enriches the interface design and improves perceived quality.

3. *Offer informative feedback.* For every user action, there should be an interface feedback. For frequent and minor actions, the response can be modest, whereas for infrequent and major actions, the response should be more substantial. Visual presentation of the objects of interest provides a convenient environment for showing changes explicitly (see the discussion of direct manipulation in Chapter 7).
4. *Design dialogs to yield closure.* Sequences of actions should be organized into groups with a beginning, middle, and end. Informative feedback at the completion of a group of actions gives users the satisfaction of accomplishment, a sense of relief, a signal to drop contingency plans from their minds, and an indicator to prepare for the next group of actions. For example, e-commerce websites move users from selecting products to the checkout, ending with a clear confirmation page that completes the transaction.
5. *Prevent errors.* As much as possible, design the interface so that users cannot make serious errors; for example, gray out menu items that are not appropriate and do not allow alphabetic characters in numeric entry fields (Section 3.3.5). If users make an error, the interface should offer simple, constructive, and specific instructions for recovery. For example, users should not have to retype an entire name-address form if they enter an invalid zip code but rather should be guided to repair only the faulty part. Erroneous actions should leave the interface state unchanged, or the interface should give instructions about restoring the state.
6. *Permit easy reversal of actions.* As much as possible, actions should be reversible. This feature relieves anxiety, since users know that errors can be undone, and encourages exploration of unfamiliar options. The units of reversibility may be a single action, a data-entry task, or a complete group of actions, such as entry of a name-address block.
7. *Keep users in control.* Experienced users strongly desire the sense that they are in charge of the interface and that the interface responds to their actions. They don't want surprises or changes in familiar behavior, and they are annoyed by tedious data-entry sequences, difficulty in obtaining necessary information, and inability to produce their desired result.
8. *Reduce short-term memory load.* Humans' limited capacity for information processing in short-term memory (the rule of thumb is that people can remember "seven plus or minus two chunks" of information) requires that designers avoid interfaces in which users must remember information from

one display and then use that information on another display. It means that cellphones should not require reentry of phone numbers, website locations should remain visible, and lengthy forms should be compacted to fit a single display.

These underlying principles must be interpreted, refined, and extended for each environment. They have their limitations, but they provide a good starting point for mobile, desktop, and web designers. The principles presented in the ensuing sections focus on increasing users' productivity by providing simplified data-entry procedures, comprehensible displays, and rapid informative feedback to increase feelings of competence, mastery, and control over the system.

3.3.5 Prevent errors

"There is no medicine against death, and against error no rule has been found."

Sigmund Freud
(inscription he wrote on his portrait)

The importance of error prevention (the fifth golden rule) is so strong that it deserves its own section. Users of cellphones, e-mail, digital cameras, e-commerce websites, and other interactive systems make mistakes far more frequently than might be expected.

One way to reduce the loss in productivity due to errors is to improve the error messages provided by the interface. Better error messages can raise success rates in repairing the errors, lower future error rates, and increase subjective satisfaction. Superior error messages are more specific, positive in tone, and constructive (telling users what to do rather than merely reporting the problem). Rather than using vague (? or WHAT?) or hostile (ILLEGAL OPERATION or SYNTAX ERROR) messages, designers are encouraged to use informative messages, such as PRINTER IS OFF, PLEASE TURN IT ON or MONTHS RANGE FROM 1 TO 12.

Improved error messages, however, are only helpful medicine. A more effective approach is to prevent the errors from occurring. This goal is more attainable than it may seem in many interfaces.

The first step is to understand the nature of errors. One perspective is that people make mistakes or *slips* (Norman, 1983) that designers can help them to avoid by organizing screens and menus functionally, designing commands and menu choices to be distinctive, and making it difficult for users to take irreversible actions. Norman also offers other guidelines, such as providing feedback about the state of the interface (e.g., changing the cursor to show whether a map interface is in zoom-in or select mode) and designing for consistency of actions (e.g., ensuring that yes/no buttons are always displayed in the same order).

Norman's analysis provides practical examples and a useful theory. Additional design techniques to reduce errors include the following:

Correct actions. Industrial designers recognize that successful products must be safe and must prevent users from dangerously incorrect usage of the products. Airplane engines cannot be put into reverse until the landing gear has touched down, and cars cannot be put into reverse while traveling forward at faster than five miles per hour. Similar principles can be applied to interactive systems—for example, inappropriate menu items can be grayed out so they can't be inadvertently selected, and web users can be allowed to simply click on the date on a calendar instead of having to type in a month and day for a desired airline flight departure. Likewise, instead of having to enter a 10-digit phone number, cellphone users can scroll through a list of frequently or recently dialed numbers and select one with a single button press. A variant idea is to provide users with auto-completion for typing words, selecting from menus, or entering web addresses.

Complete sequences. Sometimes an action requires several steps to reach completion. Since users may forget to complete every step of an action, designers may attempt to offer a sequence of steps as a single action. In automobiles, drivers do not have to set two switches to signal a left turn; a single switch causes both (front and rear) turn-signal lights on the left side of the car to flash. Likewise, when a pilot throws a switch to lower the landing gear, hundreds of mechanical steps and checks are invoked automatically.

As another example, users of a text editor can indicate that all section titles are to be centered, set in uppercase letters, and underlined without having to make a series of selections each time they enter a section title. Then if users want to change the title style—for example, to eliminate underlining—a single change will guarantee that all section titles are revised consistently. As a final example, an air-traffic controller may formulate plans to change the altitude of a plane from 14,000 feet to 18,000 feet in two increments; after raising the plane to 16,000 feet, however, the controller may get distracted and fail to complete the action. The controller should be able to record the plan and then have the computer prompt for completion. The notion of complete sequences of actions may be difficult to implement because users may need to issue atomic actions as well as complete sequences. In this case, users should be allowed to define sequences of their own. Designers can gather information about potential complete sequences by studying sequences of actions that people actually take and the patterns of errors that people actually make.

Thinking about universal usability also contributes to reducing errors—for example, a design with too many small buttons may cause unacceptably high error rates among older users or others with limited motor control, but enlarging the buttons will benefit all users. Section 4.6 addresses the idea of logging user errors so designers can continuously improve designs.

3.3.6 Ensuring human control while increasing automation

The guidelines and principles described in the previous sections are often devoted to simplifying the users' tasks. Users can then avoid routine, tedious, and error-prone actions and can concentrate on making critical decisions, selecting alternatives if the original approach fails, and acting in unanticipated situations. Users can also make subjective value-based judgments, request help from other humans, and develop new solutions (Sanders and McCormick, 1993). (Box 3.3 provides a detailed comparison of human and machine capabilities.)

Computer system designers have generally been increasing the degree of automation over time as procedures become more standardized and the pressure for productivity grows. With routine tasks, automation is desirable, since it reduces the potential for errors and the users' workload (Cummings, 2014). However, even with increased automation, informed designers can still offer the predictable and controllable interfaces that users usually prefer. The human supervisory role needs to be maintained because the real world is an *open system* (that is, a nondenumerable number of unpredictable events and system failures are possible). By contrast, computers constitute a *closed system* (only a denumerable number of normal and failure situations can be accommodated in hardware and software).

BOX 3.3

Relative capabilities of humans and machines.

Humans Generally Better

- Sense-making from hearing, sight, touch, etc.
- Detect familiar signals in noisy background
- Draw on experience and adapt to situations
- Select alternatives if original approach fails
- Act in unanticipated situations
- Apply principles to solve varied problems
- Make subjective value-based judgments
- Develop new solutions
- Use information from external environment
- Request help from other humans

Machines Generally Better

- Sense stimuli outside human's range
- Rapid consistent response for expected events
- Retrieve detailed information accurately
- Process data with anticipated patterns
- Perform repetitive actions reliably
- Perform several activities simultaneously
- Maintain performance over time

For example, in air-traffic control, common actions include changes to a plane's altitude, heading, or speed. These actions are well understood and potentially can be automated by scheduling and route-allocation algorithms, but the human controllers must be present to deal with the highly variable and unpredictable emergency situations. An automated system might deal successfully with high volumes of traffic, but what would happen if the airport manager closed a runway because of turbulent weather? The controllers would have to reroute planes quickly. Now suppose that one pilot requests clearance for an emergency landing because of a failed engine, while another pilot reports a passenger with chest pains who needs prompt medical attention. Value-based judgment, possibly with participation from other controllers, is necessary to decide which plane should land first and how much costly and risky diversion of normal traffic is appropriate. Air-traffic controllers cannot just jump into an emergency; they must be intensely involved in the situation as it develops if they are to make informed and rapid decisions. In short, many real-world situations are so complex that it is impossible to anticipate and program for every contingency; human judgment and values are necessary in the decision-making process.

Another example of the complexity of life-critical situations in air-traffic control was illustrated by an incident on a plane that had a fire on board. The controller cleared other traffic from the flight path and began to guide the plane in for a landing, but the smoke was so thick that the pilot had trouble reading his instruments. Then the onboard transponder burned out, so the air-traffic controller could no longer read the plane's altitude from the situation display. In spite of these multiple failures, the controller and the pilot managed to bring down the plane quickly enough to save the lives of many—but not all—of the passengers. A computer could not have been programmed to deal with this particular unexpected series of events.

A tragic outcome of excess of automation occurred during a flight to Cali, Colombia. The pilots relied on the automatic pilot and failed to realize that the plane was making a wide turn to return to a location that it had already passed. When the ground-collision alarm sounded, the pilots were too disoriented to pull up in time; they crashed 200 feet below a mountain peak, killing all but four people on board.

The goal of design in many applications is to give users sufficient information about current status and activities to ensure that, when intervention is necessary, they have the knowledge and the capacity to perform correctly, even under partial failures (Endsley and Jones, 2004). The U.S. Federal Aviation Agency stresses that designs should place the users in control and automate only to "improve system performance, without reducing human involvement" (U.S. FAA, 2012). These standards also encourage managers to "train users when to question automation."

The entire user interface must be designed and tested not only for normal situations but also for as wide a range of anomalous situations as can be anticipated. An extensive set of test conditions might be included as part of the requirements document. Users need to have enough information that they can take responsibility for their actions. Beyond decision making and handling of failures, the users' role is to improve the interface design.

Advocates of increased autonomy, such as in driverless cars or unmanned aircraft, believe that rapid autonomous responses improve performance and produce fewer errors. However, autonomy has risks for unanticipated situations, such as changing weather or unusual trading activity. In 2015, Toyota shifted its driverless car research from autonomous designs to ones that leave drivers in control. The dangers of unanticipated situations for Unmanned Aerial Vehicles (UAVs) resulted in shifting to Remotely Piloted Vehicles (RPVs) with human control to improve reliability. While autonomy has its benefits, designs that allow human supervisory control, activity logging, and the capacity to review logs after failures appear to improve performance.

In costly business situations, such as high-speed stock market trading, clarifying responsibility for failures could lead to improved designs. Ensuring accountability and liability in advance can encourage designers to think more carefully about potential failures. Advocates of "algorithmic accountability" want developers who implement systems such as Google's search rankings or employee hiring systems to enable open access so as to limit bias and expose errors.

Questions about integrating automation with human control also emerge in consumer product user interfaces. Many designers are eager to create an autonomous agent that knows people's likes and dislikes, makes proper inferences, responds to novel situations, and performs competently with little guidance. They believe that human-human interaction is a good model for human-computer interaction, and they seek to create computer-based partners, assistants, or agents.

By contrast, many designers believe that tool-like interfaces are often more attractive than autonomous, adaptive, or anthropomorphic agents that carry out the users' intentions and anticipate needs. The agent scenarios may show a bow-tied butler-like human, like the helpful young man in Apple Computer's famous 1987 video on the *Knowledge Navigator*. Microsoft's ill-fated 1995 BOB program used cartoon characters, while its much-criticized Clippit, nicknamed Clippy, character was also withdrawn. Human-like bank machines or postal-service stations have largely disappeared, but avatars representing users, not computers, in game-playing and 3-D social environments have remained popular; users appear to enjoy the theatrical experience of creating a new identity, sometimes with colorful hair and clothes (Section 7.6).

The success of Apple's Siri's speech recognition and personality-rich voice response system shows that with careful development, useful tools can be developed, but there is little evidence of the benefit of a talking face (Moreno and Mayer, 2007). Robot designers have perennially used human and animal forms as an inspiration, encouraging some researchers to pursue human-like robots for care of older adults or as team members in work situations. These designs attract journalists and have entertainment value but have yet to gain widespread acceptance.

A variant of the agent scenario, which does not include an anthropomorphic realization, is that the computer program has a built-in *user model* to guide an adaptive interface. The program keeps track of user performance and adapts the interface to suit the users' needs. For example, when users begin to make menu selections rapidly, indicating proficiency, advanced menu items may appear. Automatic adaptations have been proposed for interface features such as the content of menus, order of menu items, and type of feedback (graphic or tabular). Advocates point to video games that increase the speed or number of dangers as users progress through game levels. However, games are notably different from most work situations, where users bring their goals and motivations to accomplish tasks.

There are opportunities for adaptive user models to tailor designs (such as for e-mail spam filters or search results ranking), but unexpected interface behavior can have negative effects that discourage use. If adaptive systems make surprising changes, such as altering the search results ranking, users may be puzzled about what has happened. Users may become anxious because they cannot predict the next change, interpret what has happened, or return to the previous state. Users may also be annoyed if a one-time purchase of a children's book as a gift leads to repeated promotions of more children's books.

An application of user modeling is *recommender systems* in web applications. In this case, there is no agent or adaptation in the interface, but the program aggregates information from multiple sources in some (often proprietary) way. Such approaches have great practical value such as suggesting movies, books, or music; users are often intrigued to see what suggestions emerge from their purchasing patterns. Amazon.com and other e-commerce companies successfully suggest that "customers who bought X also bought Y."

The philosophical alternative to agents and user modeling is to design comprehensible systems that provide consistent interfaces, user control, and predictable behavior. Designers who emphasize a direct-manipulation style believe that users have a strong desire to be in control and to gain mastery over the system, which allows them to accept responsibility for their actions and derive feelings of accomplishment (Shneiderman, 2007). Historical evidence suggests that users seek comprehensible and predictable systems and shy away from those that are complex or unpredictable; for example, pilots may disengage automatic piloting devices if they perceive that these systems are not performing as they expect.

Agent advocates promote autonomy, but this means they must take on the issue of responsibility for failures. Who is responsible when an agent violates

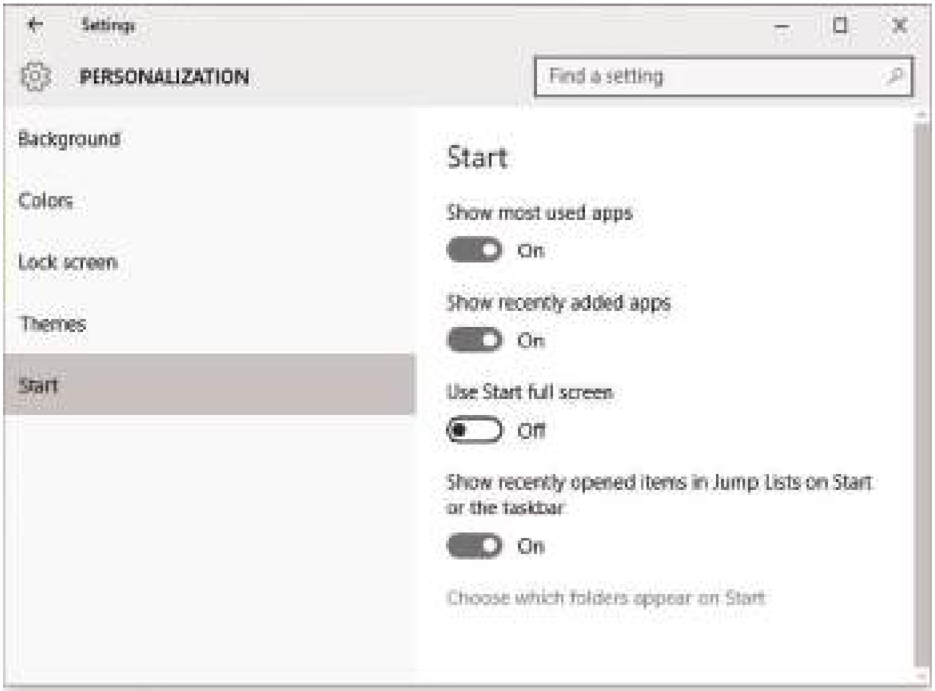


FIGURE 3.4

Windows 10 system preferences include control panels for personalization. Here we see the Start options, which allow users to control what items will display in the Start menu and taskbar.

copyright, invades privacy, or destroys data? Agent designs might be better received if they supported performance monitoring while allowing users to examine and revise the current user model.

An alternative to agents with user models may be to expand the control-panel model. Computer control panels (sometimes called settings, options, or preferences), like automobile cruise-control mechanisms and television remote controls, are designed to convey the sense of control that users seem to expect. Users employ control panels to set physical parameters, such as the cursor blinking speed or speaker volume, and to establish personal preferences such as time/date formats, color schemes, or the content of start menus (Fig. 3.4). Some software packages allow users to set parameters such as the speed of play in games. Users start at layer 1 and can then choose when to progress to higher levels; often they are content to remain experts at layer 1 of a complex interface rather than dealing with the uncertainties of higher layers. More elaborate control panels exist in style sheets of word processors, specification boxes of query facilities, and sliders of information-visualization tools.

3.4 Theories

One goal for the discipline of human-computer interaction is to go beyond the specifics of guidelines and build on the breadth of principles to develop tested, reliable, and broadly useful *theories*. Of course, for a topic as large as user-interface design, many theories are needed (Carroll, 2003; Rogers, 2012; Bødker, 2015).

Some theories are *descriptive*; these are helpful in developing consistent terminology and useful taxonomies, for objects and actions, thereby supporting collaboration and training. Other theories are *explanatory*, describing sequences of events and, where possible, cause and effect, making interventions possible. Still other theories are *prescriptive*, giving designers clear guidance for their choices. Finally, the most precise theories are *predictive*, enabling designers to compare proposed designs for execution time, error rates, conversion rates, or trust levels.

Another way to group theories is according to the types of skills involved, such as *motor* (pointing with a mouse), *perceptual* (finding an item on a display), or *cognitive* (planning the sequence of steps needed to pay a bill) (Norman, 2015). Motor skill performance predictions are well established and accurate for predicting keystroking or pointing times (see Fitts's Law, Section 10.3). Perceptual theories have been successful in predicting reading times for free text, lists, formatted displays, and other visual or auditory tasks. Cognitive theories, involving short-term, working, and long-term memory, are central to problem solving and play a key role in understanding productivity as a function of system response time (Chapter 12). However, predicting performance on complex cognitive tasks (combinations of subtasks) is especially difficult because of the many strategies that might be employed and the many opportunities for going astray. The ratio of times needed to perform complex tasks between novices and experts or between first-time and frequent users can be as high as 100 to 1. Actually, the contrast is even more dramatic because novices and first-time users often make errors and are unable to complete the tasks.

Web designers have emphasized information-architecture theories with navigation as the keys to user success. Web users can be considered as *foraging* for information, and therefore the effectiveness of the *information scent* of links is the issue (Pirolli, 2007). A high-quality link, relative to a specific task, gives users a good scent (or indication) of what is at the destination. For example, if users are trying to find a demonstration of a software package, a link with the text "download demo" has a good scent. The challenge to designers is to understand user tasks well enough to design a large website such that users will be able to find their way successfully from a home page to the right destination, even if it is three or four clicks away. Information-foraging theory attempts to

BOX 3.4

Multiple theory types that researchers and designers consider when evaluating user interfaces.

By Theory Type

• **Descriptive**	Describes user interfaces and their uses with consistent terminology and taxonomies
• **Explanatory**	Describes sequences of events with causal relationships
• **Prescriptive**	Offers guidelines for designers to make decisions
• **Predictive**	Enables comparison of design alternatives based on numeric predictions of speed or errors.

By Human Capacity

• **Motor**	Skill in pointing, clicking, dragging, or other movements
• **Perceptual**	Visual, auditory, tactile, and other human sensory inputs
• **Cognitive**	Problem solving with short- and long-term memory

predict user success rates given a set of tasks and a website so as to guide refinements.

Taxonomies can be an important part of descriptive and explanatory theories. A taxonomy imposes order by classifying a complex set of phenomena into understandable categories. For example, a taxonomy might be created for different kinds of input devices: direct versus indirect, linear versus rotary, 1-, 2-, 3- or higher-dimensional, and so on (Card et al., 1990). Other taxonomies might cover tasks (structured versus unstructured, novel versus regular) or user-interface styles (direct manipulation, menus, form fill-in). An important class of taxonomies has to do with the individual differences among users, such as personality styles (convergent versus divergent, field-dependent versus independent), technical aptitudes (spatial visualization, reasoning), and user experience levels (novice, knowledgeable, expert). Taxonomies facilitate useful comparisons, organize topics for newcomers, guide designers, and often indicate opportunities for novel products—for example, a task by type taxonomy organizes the information visualizations in Chapter 16.

Any theory that might help designers to predict performance for even a limited range of users, tasks, or designs is a contribution. At the moment, the field is filled with hundreds of theories competing for attention while being refined by their promoters, extended by critics, and applied by eager and hopeful—but skeptical—designers (Carroll, 2003, 2014; Rogers, 2012). This development is healthy for the growing discipline of human-computer interaction, but it means that practitioners must keep up with the rapid developments not only in software tools and design guidelines but also in theories. Critics raise two challenges:

1. *Theories should be more central to research and practice*. A good theory should guide researchers in understanding relationships between concepts and generalizing results. It should also guide practitioners when making design tradeoffs for products. The power of theories to shape design is most apparent in focused theories such as Fitts's Law; it is more difficult to demonstrate for explanatory theories, whose main impact may be in educating the next generation of designers.
2. *Theories should lead rather than lag behind practice*. Critics remark that too often a theory is used to explain what has been produced by commercial product designers. A robust theory should predict or at least guide practitioners in designing new products. Effective theories should suggest novel products and services while helping to refine existing ones.

Another direction for theoreticians is to predict the subjective satisfaction or emotional reactions of users. Researchers in media and advertising have recognized the difficulty of predicting emotional reactions, so they complement theoretical predictions with their intuitive judgments and extensive market testing (Nahl and Bilal, 2007).

Broader theories of small-group behavior, organizational dynamics, and sociology are proving to be useful in understanding social media and collaborative interfaces (Chapter 11). Similarly, the methods of anthropology or social psychology may be helpful in understanding technology adoption and overcoming barriers to new technology that cause resistance to change.

There may be "nothing so practical as a good theory," but coming up with an effective theory is often difficult. By definition, a theory, taxonomy, or model is an abstraction of reality and therefore must be incomplete. However, a good theory should be understandable, produce similar conclusions for all who use it, and help to solve design problems. This section reviews a range of descriptive and explanatory theories.

3.4.1 Design-by-levels theories

One approach to developing descriptive theories is to separate concepts according to levels. Such theories have been helpful in software engineering and network design. An appealing and easily comprehensible design-by-levels theory for interfaces is the four-level conceptual, semantic, syntactic, and lexical theory (Foley et al., 1995):

1. The *conceptual level* is the user's "mental model" of the interactive system. Two mental models for image creation are paint programs that manipulate pixels and drawing programs that operate on objects. Users of paint programs think in terms of sequences of actions on pixels and groups of

pixels, while users of drawing programs think in terms of sequences of actions on objects and groups of objects. Decisions about mental models affect each of the lower levels.

2. The *semantic level* describes the meanings conveyed by the user's input and by the computer's output display. For example, deleting an object in a drawing program could be accomplished by undoing a recent action or by invoking a delete-object action. Either action should eliminate only a single object and leave the rest untouched.
3. The *syntactic level* defines how the user actions that convey semantics are assembled into complete sentences to perform certain tasks. For example, the delete-files action could be invoked by dragging an object to a trash can followed by a click in a confirmation dialog box.
4. The *lexical level* deals with device dependencies and with the precise mechanisms by which users specify the syntax (for example, a function key or a mouse double-click within 200 milliseconds).

This four-level theory is convenient for designers because its top-down nature is easy to explain, matches the software architecture, and allows for useful modularity during design. Over the years, the success of graphical direct-manipulation interfaces has shifted attention up to the conceptual level, which is closest to the task domain (Parush, 2015). For example, designers of personal financial interfaces often use direct-manipulation interfaces. These interfaces build on the users' mental model of writing checks by showing the image of a check for users to fill in. The same image of a check serves as the query template so users can specify dates, payees, or amounts.

Increasingly, actions are shown by novel visual representations (for example, a trash can for deletion or a play button to start playing a video). Users have to learn the semantics (e.g., that they can recover a file by opening up the trash can or stop a video by clicking on the pause button), but if the designers choose familiar objects to associate with the actions, users can quickly acquire the correct mental model for operating the user interface. Of course, users also have to learn the syntax of dragging objects or clicking to initiate the actions, but these mechanisms are commonly used and have become well known.

The idea of design-by-levels is successful even in more complex systems with many objects and actions. For example, the human body can be discussed in terms of neural, muscular, skeletal, reproductive, digestive, circulatory, and other subsystems, which in turn might be described in terms of organs, tissues, and cells. Most real-world objects have similar decompositions: buildings into floors, floors into rooms, rooms into doors/walls/windows, and so on. Similarly, movies can be decomposed into scenes, scenes into shots, and shots into dialogue, images, and sounds. Since most objects can be decomposed in many ways, the designer's job is to create comprehensible and memorable levels of objects.

In parallel with the decomposition of objects, designers need to decompose complex actions into several smaller actions. For example, a baseball game has innings, pitches, runs, and outs, and a building-construction plan can be reduced to a series of steps such as surveying the property, laying the foundation, building the frame, raising the roof, and completing the interior. Most actions can also be decomposed in many ways, so again the designer's job here is to create comprehensible and memorable levels of actions. The goal of simplifying interface concepts while presenting visual representations of the objects and actions involved is at the heart of the direct-manipulation approach to design (Chapter 7).

When a complete user-interface design has been made, the user tasks can be described by a series of actions. These precise descriptions can serve as a basis for predicting the time required to perform tasks by simply counting up the number of milliseconds needed to complete all the steps. For example, resizing a photo may require several mouse drags, selections of menu items, clicks on dialog box buttons, and typing of dimensions, but each of these actions takes a predictable amount of time. Several researchers have successfully predicted the time required for complex tasks by adding up the times required for each component action. This predictive approach, based on *goals*, *operators*, *methods*, and *selection rules* (GOMS), decomposes goals into many operators (actions) and then into methods. Users apply selection rules to choose among alternate methods for achieving goals (Card et al., 1983; Baumeister et al., 2000).

The GOMS approach works best when the users are expert and frequent users who are working on their own, fully focused on the task, and make no mistakes. Advocates of GOMS have developed software tools to simplify and speed up the modeling process in the hope of increasing usage (John, 2011). Critics complain that broader theories are needed to predict novice user behavior, the transition to proficiency, the rate of errors, and retention over time.

Designers have discovered that using design-by-levels theories forces clear definitions of the high-level objects and actions, which are gathered from listening to the language used in the task domain. Music can be thought of as songs, organized by artists, albums, and genres. Users can find a song and then play it or add it to a playlist. The clarity of this conceptual structure earned it a patent and has stimulated multiple commercial successes.

3.4.2 Stages-of-action theories

Another approach to forming explanatory theories is to portray the stages of action that users go through in using interactive products such as information appliances, web interfaces, or mobile devices (e.g., music players). Norman (2013) offers seven stages of action, arranged in a cyclic pattern, as an explanatory theory of human-computer interaction:

1. Forming the goal
2. Forming the intention

3. Specifying the action
4. Executing the action
5. Perceiving the system state
6. Interpreting the system state
7. Evaluating the outcome

Some of Norman's stages correspond roughly to Foley et al.'s (1995) separation of concerns; that is, users form a conceptual intention, reformulate it into the semantics of several commands, construct the required syntax, and eventually produce the lexical token by the action of moving the mouse to select a point on the screen. Norman makes a contribution by placing his stages in the context of *cycles of action* and *evaluation*, which take place over seconds and minutes. This dynamic process of action distinguishes Norman's approach from the other theories, which deal mainly with knowledge that must be in the user's mind. Furthermore, the seven stages of action lead naturally to identification of the *gulf of execution* (the mismatch between the user's intentions and the allowable actions) and the *gulf of evaluation* (the mismatch between the system's representation and the user's expectations).

This theory leads Norman to suggest four principles of good design:

1. The state and the action alternatives should be visible.
2. There should be a good conceptual model with a consistent system image.
3. The interface should include good mappings that reveal the relationships between stages.
4. Users should receive continuous feedback.

Norman places a heavy emphasis on studying errors, describing how errors often occur in moving from goals to intentions to actions and to executions.

The stages-of-action theory helps designers to describe user exploration of an interface (Polson and Lewis, 1990). As users try to accomplish their goals, there are four critical points where user failures can occur: (1) users may form inadequate goals, (2) users might not find the correct interface object because of an incomprehensible label or icon, (3) users may not know how to specify or execute a desired action, and (4) users may receive inappropriate or misleading feedback.

Refinements of the stages-of-action theory have been developed for other domains. For example, information seeking has been characterized by these stages: (1) recognize, (2) accept the information problem, (3) formulate and (4) express the query, then (5) examine the results, (6) reformulate the problem, and (7) use the results (Marchionini and White, 2007). Of course, there are variations with users skipping stages or going back to earlier stages, but the model helps guide designers and users.

Commercial website designers know the benefit of a clear stages-of-action theory in guiding anxious users through a complex process. For example, the Amazon.com website converts the potentially confusing checkout process into a comprehensible four-stage process: (1) Sign-in, (2) Shipping & Payment, (3) Gift-Wrap, and (4) Place Order. Users can simply move through these four stages or back up to previous stages to make changes. Amazon.com also recognizes the need for a frequent user shortcut, the 1-click purchase, for products such as a Kindle book.

Designers can apply the stages-of-action theory by thinking deeply about the beginning, middle, and end stages to ensure that they cover a wide enough scope of usage. Many new products emerge as a result of adding novel features to what was considered a well-defined process; for example, expanding the music-playing process to include the earlier stages of music purchase or composition and the later stages of music sharing or reviewing/rating.

3.4.3 Consistency theories

An important goal for designers is a *consistent* user interface. The argument for consistency is that if terminology for objects and actions is orderly and describable by a few rules, users will be able to learn and retain them easily. This example illustrates consistency and two kinds of inconsistency (**A** illustrates lack of consistency, and **B** shows consistency except for a single violation):

Consistent	**Inconsistent A**	**Inconsistent B**
delete/insert table	delete/insert table	delete/insert table
delete/insert column	remove/add column	remove/insert column
delete/insert row	destroy/create row	delete/insert row
delete/insert border	erase/draw border	delete/insert border

Each of the actions in the consistent version is the same, whereas the actions vary for inconsistent version A. The inconsistent action verbs are all acceptable, but their variety suggests that they will take longer to learn, will cause more errors, will slow down users, and will be harder for users to remember. Inconsistent version B is somehow more startling, because there is a single unpredictable inconsistency; it stands out so dramatically that this language is likely to be remembered for its peculiar inconsistency.

Consistency for objects and actions (nouns and verbs) is a good starting point, but there are many other forms of consistency that require careful thought by designers. Consistent use of color, layout, icons, fonts, font sizes, button sizes, and much more is vital in giving users a clear understanding of the interface. Inconsistency in elements such as the positioning of buttons or colors will slow users down by 5–10%, while changes to terminology slow users by 20–25%.

Consistency is an important goal, but there may be conflicting forms of consistency, and sometimes inconsistency is a virtue (for example, to draw attention to a dangerous action). Competing forms of consistency require designers to make difficult choices or invent new strategies. For example, while automobile interface designers have agreed to always place the accelerator pedal to the right of the brake pedal, there's no agreement about whether turn signal controls should be to the right or left of the steering wheel.

Consistency issues are critical in the design of mobile devices. In successful products, users get accustomed to consistent patterns, such as initiating actions with a left-side button while terminating actions with a right-side button. Similarly, up and down scrolling actions should be done consistently using buttons that are vertically aligned. A frequent problem is the inconsistent placement of the Q and Z characters on phone buttons.

Designers can enforce consistency by developing detailed guidelines documents for their designs (Section 4.3) that spell out all of the consistency requirements. Expert reviewers of user interfaces can then verify the consistency of the design. This requires a careful eye and thoughtful attention to how each screen is laid out, each action sequence is carried out, and each sound is played.

3.4.4 Contextual theories

The design-by-levels, stages-of-action, and consistency theories address the specifics of how objects and actions appear on displays and what actions users take to carry out their tasks. These theories and design aspects might be called *micro-HCI*, since they cover measurable performance in terms of speed and errors. Micro-HCI is best studied with the scientific methods of experimental and cognitive psychology using 30- to 120-minute controlled experiments and statistical tests for significant differences between groups working on well-defined tasks

Micro-HCI has been and continues to be a great success story, but there is a growing awareness that tightly controlled laboratory studies of isolated phenomena are only one part of the story. The rise of *macro-HCI*, which emphasizes the user experience, the usage context, and social engagement, has opened up new possibilities for researchers and practitioners. While micro-HCI research is more about laboratory studies to collect clear performance measures for identifiable tasks (e.g., how many seconds to find the last flight on July 4 from Washington, DC, to London), macro-HCI research is more about ethnographic observation of users doing work or play in their familiar context over days or even months. The outcomes of micro-HCI research are statistically significant differences that support or refute a hypothesis, while the outcomes of macro-HCI research are insights about what leads to increased user satisfaction, how the context of use matters, and how new applications could improve education, health, safety, or the environment.

Macro-HCI thinking leads to different kinds of theories that might best be called contextual, since they consider the emotional, physical, and social contexts of use. Happy users will persevere in the face of frustrations, cope with interruptions from neighbors, and ask for help when they need it. In short, the

BOX 3.5

Theory types that organize evaluation of user interfaces and guide design.

Micro-HCI Theories	Focus on measurable performance (such as speed and errors) on multiple standard tasks taking seconds or minutes in laboratory environments
• **Design-by-levels**	Start with high-level design and move to smaller objects and actions
• **Stages-of-action**	Consider user behavior as they form intentions and seek to realize their goals.
• **Consistency**	Strive for consistency in objects and actions, shown by words, icons, colors, shapes, gestures, menu choices
Macro-HCI Theories	Focus on case studies of user experience over weeks and months in realistic usage contexts with rich social engagement
• **Contextual**	Support users who are embedded in emotional, physical, and social environments
• **Dynamic**	Design for evolution of user behavior as users move through levels of mastery, performance, and leadership

physical and social environments are inextricably intertwined with use of information and communications technologies. Design cannot be separated from patterns of use.

Suchman's (1987) analysis in her book *Plans and Situated Action* is often credited with launching this reconsideration of human-computer interaction. She argued that the cognitive model of orderly human plans that were executed when needed was insufficient to describe the richer and livelier world of work or personal usage. She proposed that users' actions were situated in time and place, making user behavior highly responsive to other people and to environmental contingencies. If users got stuck in using an interface, they might ask for help, depending on who was around, or consult a manual, if one were available. If they were pressed for time, they might risk some shortcuts, but if the work was life-critical, they would be extra cautious. Rather than having fixed plans, users constantly changed their plans in response to the circumstances. The argument of distributed cognition is that knowledge is not only in the users' minds but distributed in their environments—knowledge is stored on paper documents, accessible from electronic files, or available from colleagues.

Contextual theories also address the shift from use of a computer to interaction with a device-rich environment filled with sensors, responsive appliances, display walls, and audio generators. Rather than picking up a device, users activate automatic doors, hand dryers, or light switches. Sometimes users are inside a device such as an automobile, forcing designers to consider the surrounding space and the other people in the car as well as the sounds, vibrations, and forces of acceleration. Contextual theories often emphasize the social environment in which users are engaged with other people who can provide assistance or can be distractions.

Advocates of contextual theories believe that the turbulence of actual usage (as opposed to idealized task specifications) means that users have to be more than test subjects—they have to be participants in design processes. Proponents of contextual theories encourage more ethnographic observation, longitudinal case studies, and action research by participant observers (Boellstorff et al., 2012; Crabtree et al., 2012; Horst and Miller, 2013).

Breakdowns are often seen as sources of insight about design, and users are encouraged to become reflective practitioners who are continuously engaged in the process of design refinement. Understanding the transition from novice to expert and the differences in skill levels has become a focus of attention, further calling into question the utility of hour-long laboratory or half-day usability-testing studies as a guide to the behavior of users after a month or more of experience.

Contextual theories are especially relevant to mobile devices and ubiquitous computing innovations. Such devices are portable or installed in a physical space, and they are often designed specifically to provide place-specific information (for example, a city guide on a portable computer or a museum guide that gives information on a nearby painting). Location information by way of GPS systems enables new services but raises concerns about misuse of tracking information.

Designers can apply contextual theories by observing users in their own environments as they carry out their work, engage socially, or participate in sports or play. A detailed record of how tasks are chosen and carried out, including collaborations with others, internal or external interruptions, and errors that occur, would lay the basis for interface design. Contextual theories are about how people form intentions, how aspirations crystalize, how empathy is encouraged, and how trust shapes behavior; they are also about emotional states of excitement or frustration, the joy of attaining goals, and the disappointment of failure. These strong reactions are hard to capture in predictive mathematical equations, but it is important to study and understand them. To that end, many researchers are shifting their methods from controlled experiments to ethnographic observation, focus group discussions, and long-term case studies. Surveys and interviews can provide quantitative data for much-needed theories of how design variables affect users' levels of satisfaction, fear, trust, and cooperativeness.

While contextual theories emphasize the changes to observation and research, contextual theories can also guide design. If interruptions are an impediment, then users might be given the option of blocking them. If usage outdoors is a requirement, then contrast setting or font sizes should be easily adjustable. If collaboration with others is a high priority, then easy sharing of screens or texting should be possible.

A taxonomy of mobile device applications could guide innovators:

- *Monitor* blood pressure, stock prices, or air quality and give *alerts* when normal ranges are exceeded.
- *Gather* information from meeting attendees or rescue team members and *spread* the action list or current status to all.
- *Participate* in a large group activity by voting and *relate* to specific individuals by sending private messages.
- *Locate* the nearest restaurant or waterfall and *identify* the details of the current location.
- *Capture* information or photos left by others and *share* yours with future visitors.

These five pairs of actions could be tied to a variety of objects (such as photos, annotations, or documents), suggesting new mobile devices and services. They also suggest that one way of thinking about user interfaces is by way of the objects that users encounter and the actions that they take (Robinson et al., 2015). A more ambitious use of mobile devices is to aggregate information from thousands of cellphones to determine where there is highway congestion or which rides at an amusement park have the longest waiting lines.

3.4.4 Dynamic theories

A key aspect of macro-HCI is how users evolve over weeks and months, especially as they move from novices to experts, from new customers to frequent buyers, or from readers of Wikipedia to active collaborators or administrators. These theories address design for evolutionary development of skills mastery, behavior change, reputation growth, and leadership capacities.

Dynamic theories owe much to the theories of adoption or innovation diffusion (Rogers, 2003), which include five attributes:

1. relative advantage: faster, safer, more error free usage, or cheaper
2. compatibility: fitting for users' need, consistent with existing values
3. trial-ability: availability to experiment with innovation
4. observability: visibility of innovation to others
5. less complexity: ease of learning and use

These attributes lead to macro-HCI design guidelines, such as suggesting specific user-interface features, combining features to make some more visible than others, and providing informative feedback to users about their usage history. Other macro-HCI design guidelines will suggest ways of training users about features (informing them about new features), rewarding them for successes (showing their progress in reading a book or their score in a game), and sharing their progress with others (notifying friends about an exercise achievement or business associates about a price change).

Dynamic theories deal with long-term (weeks or months) changes in behavior for health (smoking cessation, diet, exercise, or performance in memory games) or education (completing an online course or demonstrating increased familiarity with a body of knowledge). A large category of dynamic theories cover customer loyalty plans that encourage increased commitment, such as awards from restaurants, airlines, or hotels. These carefully designed programs have multiple award levels, such as bronze, silver, gold, and platinum, with carefully chosen benefits to encourage increased activity.

Behavior change by badge awards and loyalty programs will become increasingly important because of the growing data sources about what works and what doesn't. The remarkably focused and personalized ways of persuading users and raising motivation will dramatically increase the possibilities for designers who understand when personal recognition, social rewards, community awareness, and financial compensation are most effective.

Dynamic theories are strong among designers of online communities and user-generated content sites. They know that users often move through stages as they gain confidence and a greater sense of responsibility for quality. There are many paths, but a study of Wikipedia contributors (Bryant et al., 2005) suggests at least these stages: (1) reader of articles related to personal interests, (2) fixer of mistakes and omissions in familiar topics, (3) registered user and caretaker for a collection of articles, (4) author for new articles, (5) participant in community of authors, and (6) administrator who is active in governance and future directions.

Following these results, the Reader-to-Leader Model described how to design user-interface and social engagement features to promote movement through these stages over a period of weeks or months (Preece and Shneiderman, 2009). At early stages, there are user-interface design guidelines, such as highlighting key features and valuable content, and social engagement design guidelines, such as encouragement from friends, family, and respected authorities. At later stages, there are user-interface design guidelines, such as visible recognition for contributions, and social engagement design guidelines, such as promoting empathy, supporting mentoring, raising trust, and facilitating conflict resolution.

Macro-HCI theories also promote the idea that user interfaces have profound societal effects with positive outcomes such as increased social communication, safety, or health awareness and negative outcomes such as undermining concentration, invading privacy, or exposing users to hackers. Visionaries see user interfaces as shaping personal processes of mindfulness, reflection, or empathy and community processes of civic participation, democratic sharing, or conflict resolution (Bell and Dourish, 2011; Nelson and Stolterman, 2012; Calvo and Peters, 2014). At a grander scale, macro-HCI dreamers believe that better user interfaces and user experiences can support international development, improved healthcare, environmental preservation, and peaceful dispute reconciliation.

Practitioner's Summary

Design principles and guidelines are emerging from practical experience and empirical studies. Managers can benefit by reviewing available guidelines documents and then constructing local versions. These documents record organizational policies, support consistency, and record the results of practical and experimental testing. Guidelines documents also stimulate discussion of user-interface issues and help train new designers. More established principles—such as recognizing user diversity, striving for consistency, and preventing errors—have become widely accepted, but they require fresh interpretation as technology and applications evolve. Automation is increasing for many tasks, but preserving human control is still a beneficial goal.

Micro-HCI and macro-HCI theories are being validated and refined to clarify the design implications. For expert users with established sequences of actions, predictive models that guide designers to reduce the time required for each step are valuable. For novel applications and novice users, clarifying task objects and actions (for example, songs and albums that can be played or added to playlists) and promoting consistency can lead to easily learned designs that promote user confidence. For every design, extensive testing and iterative refinement are necessary parts of the development process.

Researcher's Agenda

The central problem for human-computer–interaction researchers is developing adequate micro-HCI and macro-HCI theories. Traditional psychological theories must be extended and refined to accommodate the complex human learning, memory, and problem solving required in user interfaces and user experiences. Useful goals include descriptive taxonomies, explanatory theories, and predictive models. When predictions can be made for learning times, performance speeds, error rates, subjective satisfaction, or human retention over time, designers can more easily choose among competing designs.

Theories in human-computer interaction can be grouped into five families: those that focus on design by levels, stages of action, consistency, contextual awareness, and evolutionary dynamics. Theories can be useful even if they are narrowly focused on a specific task, such as choosing a video from a database of millions of videos. Even more powerful are theories that apply to diverse tasks such as web searching, online reviewing, or encouraging community participation. Applied research problems are suggested by each of the hundreds of design principles or guidelines that have been proposed. Each validation of these principles and clarification of the breadth of applicability is a small but useful contribution to the emerging mosaic of human performance with interactive systems.

WORLD WIDE WEB RESOURCES

www.pearsonglobaleditions.com/shneiderman

Many websites include guidelines documents for desktop, web, and mobile device interfaces and recommendations for universal usability strategies to accommodate users with disabilities or other special needs. Theories are proliferating, and the web is a good place to keep up with the latest ones from major developers and sources promoting universal usability:

- Apple Human Interface Guidelines: http://developer.apple.com
- Microsoft Windows User Experience Interaction Guidelines: https://msdn.microsoft.com
- World Wide Web Consortium (W3C) guidelines: http://www.w3.org/TR/WCAG20/
- Interaction Design Foundation Encyclopedia covers theories: https://www.interaction-design.org/

Debates over hot topics can be found in relevant blogs and newsgroups, which are searchable from many standard services such as Google or Bing.

Discussion Questions

1. Give a brief explanation of the Eight Golden Rules of Interface Design. State an example you have seen on a device, computer interface, or web site that violates those rules.
2. Don Norman suggests organizing screens and menus functionally, designing commands and menu choices to be distinctive, and making it difficult for users to take irreversible actions. Norman also says to provide feedback about the state of the interface (e.g., changing the cursor to show whether a map interface is in zoom-in or select mode) and designing for consistency of actions (e.g., ensuring that Yes/No buttons are always displayed in the same order). State one example you have seen where you know these rules have been violated. Although this is crucial to a user interface's success, suggest why there may be challenges to implement some of Norman's guidelines.
3. Clarify the difference among guidelines, principles, and theories.

4. What are some of the techniques that can be used to get the user's attention? Why is it important to exercise caution when using these techniques?
5. What are the stages of forming explanatory theories as suggested by Don Norman?

References

Baumeister, L., John, B. E., and Byrne, M., A comparison of tools for building GOMS models, *Proc. CHI 2000 Conference: Human Factors in Computing Systems*, ACM Press, New York (2000), 502–509.

Bell, Genevieve, and Dourish, Paul, *Divining a Digital Future: Mess and Mythology in Ubiquitous Computing*, MIT Press (2011).

Boellstorff, Tom, Nardi, Bonnie, Pearce, Celia, and Taylor, T. L., *Ethnography and Virtual Worlds: A Handbook of Method*, Princeton University Press (2012).

Bødker, Susanne, Third Wave HCI, 10 years later: Participation and sharing, *ACM* Interactions *22,* 5 (Sept.–Oct. 2015), 24.

Bryant, Susan, Forte, Andrea, and Bruckman, Amy, Becoming Wikipedian: Transformation of participation in a collaborative online encyclopedia, *Proc. ACM SIGGROUP International Conference on Supporting Group Work,* ACM Press, New York (2005), 1–10.

Calvo, Rafael A., and Peters, Dorian, *Positive Computing: Technology for Wellbeing and Human Potential*, MIT Press (2014).

Card, Stuart K., Mackinlay, Jock D., and Robertson, George G., The design space of input devices*, Proc. CHI '90 Conference: Human Factors in Computing Systems,* ACM Press, New York (1990), 117–124.

Card, Stuart, Moran, Thomas P., and Newell, Allen, *The Psychology of Human-Computer Interaction,* Lawrence Erlbaum Associates, Hillsdale, NJ (1983).

Carroll, John M. (Editor), *HCI Models, Theories, and Frameworks: Toward a Multidisciplinary Science,* Morgan Kaufmann, San Francisco, CA (2003).

Carroll, John M., Human computer interaction—brief intro. In Soegaard, Mads, and Dam, Rikke Friis (Editors), *The Encyclopedia of Human-Computer Interaction,* 2nd Edition, The Interaction Design Foundation (2014). Available at https://www.interaction-design.org/encyclopedia/human_computer_interaction_hci.html.

Crabtree, Andrew, Rouncefield, Mark, and Tolmie, Peter, *Doing Design Ethnography,* Springer, London (2012).

Cummings, M. L., Man versus machine or man + machine? *IEEE Intelligent Systems 29,* 5 (2014), 62–69.

Endsley, Mica R., and Jones, Debra G., *Situation Awareness: An Approach to User-Centered Design,* 2nd Edition, CRC Press (2004).

Foley, James D., van Dam, Andries, Feiner, Steven K., and Hughes, John F., *Computer Graphics: Principles and Practice in C,* 2nd Edition, Addison-Wesley, Reading, MA (1995).

Grudin, J., A moving target: The evolution of human-computer interaction. In J. Jacko (Editor), *Human-Computer Interaction Handbook: Fundamentals, Evolving Technologies, and Emerging Applications,* 2nd Edition, Taylor & Francis (2012).

Hackos, JoAnn T., and Redish, Janice C., *User and Task Analysis for Interface Design*, John Wiley & Sons, New York (1998).

Hartson, R., and Pyla, P., *The UX Book: Process and Guidelines for Ensuring a Quality User Experience*, Morgan Kaufmann (2012).

Horst, Heather A., and Miller, Daniel (Editors), *Digital Anthropology*, Bloomsbury (2013).

John, B. E., Using predictive human performance models to inspire and support UI design recommendations, in *Proceedings of the Conference on Human Factors in Computing Systems* (CHI '11), ACM, New York, NY (2011), 983–986.

Johnson, Jeff, *Designing with the Mind in Mind: Simple Guide to Understanding User Interface Design Rules,* 2nd Edition, Morgan Kaufmann, San Francisco, CA (2014).

Lynch, Patrick J., and Horton, Sarah, *Web Style Guide: Basic Design Principles for Creating Web Sites,* 3rd Edition, Yale University Press, New Haven, CT (2008).

Marchionini, G., and White, R. W., Find what you need, understand what you find, *International Journal of Human-Computer Interaction 23,* 3 (2007), 205–237.

Moreno, R., and Mayer, R. E., Interactive multimodal learning environments*, Educational Psychology Review* 19 (2007), 309–326.

Nahl, Diane, and Bilal, Dania (Editors), *Information and Emotion: The Emergent Affective Paradigm in Information Behavior Research and Theory*, Information Today, Medford, NJ (2007).

National Cancer Institute, *Research-based Web Design and Usability Guidelines*, Dept. of Health & Human Services, National Institutes of Health (2006, updated on the web at http://www.usability.gov).

Nelson, H. G., and Stolterman, E., *The Design Way: Intentional Change in an Unpredictable World,* 2nd Edition, MIT Press, Cambridge, MA (2012).

Norman, Donald A., Design rules based on analyses of human error, *Communications of the ACM* 26, 4 (1983), 254–258.

Norman, Donald A., *Design of Everyday Things,* Revised Edition, Basic Books, New York (2013).

Norman, Kent L., *Cyberpsychology*: *An Introduction to the Psychology of Human-Computer Interaction,* 2nd Edition, Cambridge University Press, New York (2015).

Parush, Avi, *Conceptual Design for Interactive Systems: Designing for Performance and User Experience,* Elsevier/Morgan Kaufmann (2015).

Pirolli, Peter, *Information Foraging: Adaptive Interaction with Information*, Oxford University Press (2007).

Polson, Peter, and Lewis, Clayton, Theory-based design for easily learned interfaces, *Human-Computer Interaction* 5 (1990), 191–220.

Preece, J., and Shneiderman, B., The Reader-to-Leader Framework: Motivating technology-mediated social participation, *AIS Transactions on Human-Computer Interaction 1*, 1 (March 2009), 13–32. Available at http://aisel.aisnet.org/thci/vol1/iss1/5/.

Robinson, Simon, Jones, Matt, and Marsden, Gary, *There's Not an App for That: Mobile User Experience Design for Life*, Morgan Kaufmann (2015).

Rogers, Everett M., *Diffusion of Innovations,* 5th Edition, Free Press, New York (2003).

Rogers, Yvonne, *HCI Theory: Classical, Modern, and Contemporary*, Synthesis Lectures in Human-Centered Informatics (Series Editor John M. Carroll), Morgan & Claypool Publishers (2012).

Sanders, M. S., and McCormick, E. J., *Human Factors in Engineering and Design*, 7th Edition, McGraw-Hill, New York (1993).

Shneiderman, Ben, Promoting universal usability with multi-layer interface design, *ACM Conference on Universal Usability*, ACM Press, New York (2003), 1–8.

Shneiderman, Ben, Human responsibility for autonomous agents, *IEEE Intelligent Systems 22,* 2 (March/April 2007), 60–61.

Smith, Sid L., and Mosier, Jane N., *Guidelines for Designing User Interface Software*, Report ESD-TR–86–278, Electronic Systems Division, MITRE Corporation, Bedford, MA (1986). Available from National Technical Information Service, Springfield, VA.

Suchman, Lucy A., *Plans and Situated Actions: The Problem of Human-Machine Communication*, Cambridge University Press, Cambridge, U.K. (1987).

U.S. Federal Aviation Administration, *The Human Factors Design Standard*, Atlantic City, NJ (updated May 2012). Available at http://hf.tc.faa.gov/hfds/.

Wickens, Christopher D., Hollands, Justin G., Banbury, Simon, and Parasuraman, Raja, *Engineering Psychology and Human Performance,* 4th Edition, Psychology Press (2012).

PART 2

DESIGN PROCESSES

PART OUTLINE

Design is a deliberate process of increasing fixation, where a designer creates progressively detailed plans for a specific manufactured object or service. Interaction design applies these ideas to the creation of digital artifacts, such as applications, websites, and devices. Why include an entire part focused on design in a book on human-computer interaction? One reason is that, after all, the word *design* is part of the title of this book. However, a more accurate answer is that human-computer interaction at its core is a design discipline. There are no intrinsic "laws" for interaction design, only guidelines, rules of thumb, and best practices. For this reason, applications of design thinking, processes, and methods are already scattered throughout this book in the respective chapters, such as on menu design (Chapter 8), devices (Chapter 10), and visualization (Chapter 16). To facilitate understanding these specific applications of design, it makes sense to collect all fundamental design concepts into a single part of the book to make referring to the material fast and convenient for the reader.

Because of the complexity of the interaction design process, we have split this part into three chapters. The first chapter (Chapter 4) deals with the creative phases of the design process: requirements analysis, preliminary and detailed design, and implementation. Chapter 5 discusses the evaluation phase in great detail, ranging from usability studies to expert reviews. Finally, Chapter 6 showcases how to apply our design process in practice through three detailed case studies involving ATMs, Apple, and Volvo.


Experience
Web research
Gaining Skills
Protests
Deadlines
Semester


space

CHAPTER 4

Design

"Just as we can assert that no product has ever been created in a single moment of inspiration . . . nobody has ever produced a set of requirements for any product in a similarly miraculous manner. These requirements may well begin with an inspirational moment but, almost certainly, the emergent bright idea will be developed by iterative processes of evaluation until it is thought to be worth starting to put pencil to paper. Especially when the product is entirely new, the development of a set of requirements may well depend upon testing initial ideas in some depth."

W. H. Mayall
Principles in Design, 1979

"The Plan is the generator. Without a plan, you have lack of order and willfulness. The Plan holds in itself the essence of sensation."

Le Corbusier
Towards a New Architecture, 1931

CHAPTER OUTLINE

4.1 Introduction

Design can be loosely defined as the outcome or the process of creating specifications for synthetic artifacts, such as products, services, and processes. All manufactured objects in the world—objects that were made by people and are not found in nature—are the result of some form of design process, whether a deliberate one or otherwise. User interfaces, which are very much synthetic and decidedly do not occur in nature, are no exception. However, while early computer manufacturers were quick to enlist industrial designers to shape the physical form factors of the first computers, they were much less agile in recognizing the need for *interaction design* (Moggridge, 2007): the design of the digital interface itself. Now an established design discipline in its own right, interaction design is defined as making plans and specifications for digital objects, which include devices, interfaces, services, and information.

Every time designers create a new digital artifact, they make decisions—unconscious or not—on how the artifact will look, feel, and function. If they carefully consider how digital products and services are created, they can make appealing products and services that respond to human needs with user interfaces that are easy to learn, comprehensible, and efficient to use. Early computer applications were designed by programmers to be highly functional for the programmers themselves and their peers, but this approach quickly failed when the audience for computers grew to non-technical fields. Bill Moggridge (2007) calls this phenomenon being "kind to chips but cruel to people," and it was an early failing of interaction design.

The current generation of users for smartphones, social media, and e-commerce have vastly different backgrounds from programmers and engineers. They have no interest in obscure interfaces but are more oriented toward their professional or recreational needs and are less dedicated to the technology itself. Therefore, effective interaction design takes the intended user as its starting point and focuses on facilitating the function of the artifact. As a result, professional interaction designers carefully observe their users, iteratively refine their prototypes based on thoughtful analysis, and systematically validate their interfaces through early usability and acceptance tests. However, as for any design discipline, function is not the only important aspect of a digital object. Form is another aspect, and it sometimes comes in conflict with function. While on the one hand it can be argued that good form will facilitate function (since an aesthetically appealing artifact can invite use), it is also true that a highly convoluted form may inhibit it. Consider a kitchen cabinet door with no handles: slick and appealing according to contemporary design sense but lacking visual indications of how and where to open the door. In fact, it may not even be immediately obvious that a door with no handles is in fact a door, let alone that it can be opened! The same is true for interfaces: It often makes sense to let form follow function (Sullivan, 1896). Tradeoffs between form and function are discussed in Chapter 12.

If there are several similarities between interaction design and other design disciplines, what is particularly unique about interaction design? One of the key characteristics of digital media is that they are freely reproducible without consuming the original copy or costing additional resources. They also have few of the physical requirements that real materials must obey, such as cost, ease of manufacturing, or physical robustness. In essence, information technology is thus a "material without qualities" (Löwgren and Stolterman, 2004). For software engineering, this fact has led to the global open source movement, where programmers, even professional ones, are willing to give away the results of their hard work for free. In the context of interfaces and interactions, digital media mean that designers work under few physical constraints compared to tangible artifacts. A digital button can be arbitrarily large or small, or it can be entirely gold- or diamond-plated, with no added or reduced cost to the overall project. In fact, the designer can freely experiment with any number of alternate designs during the process without incurring any other cost than time. However, this added freedom is a double-edged sword in that constraints can often be helpful in reducing the space of potential designs (also known as the *design space*) and even boosting the creativity of the designer; after all, necessity is said to be the mother of invention. With no such helpful constraints to reduce the design space for digital interfaces and objects, interaction designers are often left with a much more daunting problem than industrial designers working in the real world.

The key to good design starts in the organization itself. The primary reason for this is that design is unpredictable, which requires an agile organizational structure as well as a comprehensive business strategy oriented around diverse design processes. In fact, some companies, such as Apple, Pepsi, Philips, and Kia Motors, have hired chief design officers (CDOs) in recognition of this unpredictability. Section 4.2 offers examples of such structures and strategies that managers can adapt to suit their organizations, projects, schedules, and budgets.

This unpredictable and dynamic nature requires a robust and flexible design process. Section 4.3 describes a four-phase iterative design process consisting of requirements analysis (Phase 1), preliminary and detailed design (Phase 2), build and implementation (Phase 3), and finally evaluation (Phase 4, described in Chapter 5). This cycle is repeated until the outcome from the evaluation phase is acceptable given the requirements. The design cycle itself is part of a larger cycle that incorporates the entire life cycle of a product, including deployment, maintenance, and new updates to the system.

Design frameworks are discussed in Section 4.4 and permeate the entire design philosophy and design methods used in the process. Three specific frameworks are of particular interest to interaction designers: agile and rapid prototyping, user-centered design, and participatory design. The exact choice of which design framework to use depends on the organization, the project team, and the product being designed.

If frameworks provide the high-level structure, then the design methods are the building blocks that are used to populate the structure. Section 4.5 reviews popular interaction design methods for each phase of the design process, including ethnographic observation and sketching (Phase 1), storyboarding and scenario development (Phase 2), and paper mockups and prototyping (Phase 3). Evaluation methods for Phase 4 are described separately in Chapter 5.

Design is a challenging activity and is difficult to learn in a purely theoretical setting, particularly for newcomers but also for seasoned designers entering a new domain. Section 4.6 offers several practical, hands-on best practices to facilitate the design process, including UX prototyping tools, UX guidelines documents, and the notion of *design patterns* for interaction design and UX. Originally derived for as disparate areas as urban planning (Alexander, 1977) and software engineering (Freeman et al., 2004), design patterns are concrete and reusable solutions to commonly occurring problems. The section shows how such design patterns can be applied to interaction design.

This chapter concludes with Section 4.7, which describes legal concerns that should be addressed in the design process, including topics such as privacy, safety, intellectual property, standardization, and certification.

See also:

Chapter 5, Evaluation and the User Experience

Chapter 12, Advancing the User Experience

4.2 Organizational Support for Design

Most companies may not yet have chief usability officers (CUOs) or vice presidents for usability, but some companies are beginning to employ chief design officers (CDOs), which may help to promote usability and design thinking at every level. A case in point is Apple Inc., which was one of the first companies with a CDO and which accordingly has been praised for its innovative, well-designed, and usable products. Even if a company has no CDO, organizational awareness can be stimulated by presentations, internal seminars, newsletters, and awards. However, resistance to new techniques and changing roles for software engineers can become a problem in traditional organizations.

Organizational change is difficult, but creative leaders blend inspiration and provocation. The high road is to appeal to the desire for quality that most professionals share. When they are shown data on shortened learning times, faster performance, or lower error rates on well-designed interfaces, managers are likely to be more sympathetic to applying usability-engineering methods. Even more compelling for e-commerce managers is evidence of higher rates of conversion, enlarged market share, and increased customer retention. For managers of consumer products, the goals include fewer returns/complaints, increased brand loyalty, and more referrals. The low road is to point out the frustration, confusion, and high error rates caused by current complex designs while citing the successes of competitors who apply usability-engineering methods.

Collecting momentum for organizational change can come from different sources. Major corporations almost always question the *return on investment* (ROI) for usability engineering and interaction design. However, the business case for focusing on usability has been made powerfully and repeatedly (Karat, 1994; Marcus, 2002; Bias and Mayhew, 2005; Nielsen, 2008). Claire-Marie Karat's (1994) business-like reports within IBM became influential documents when they were published externally. She reported up to $100 payoffs for each dollar spent on usability, with identifiable benefits in reduced program-development costs, reduced maintenance costs, increased revenue due to higher customer satisfaction, and improved user efficiency and productivity. Other economic analyses showed fundamental changes in organizational productivity (with improvements of as much as 720%) when designers kept usability in mind from the beginning of development projects (Landauer, 1995).

The necessary pressure for change may also come from the customers themselves. Corporate marketing and customer-assistance departments are becoming more aware of the importance of usability and are a source of constructive encouragement. When competing products provide similar functionality, usability engineering is vital for product acceptance. Today's customers are discerning and expect high quality, and their overall brand loyalty is steadily diminishing. Retaining as well as increasing its customer base can provide a powerful incentive for an organization to improve its focus on interaction design and usability engineering.

Finally, usability engineering is required for certification and standardization in some industries. For example, the aerospace industry has Human Systems Integration (HSI) requirements that deal with a combination of human factors, usability, display design, navigation, and so forth (National Research Council, 2007).

As a result, most large and many small organizations now maintain a centralized human factors group or usability/UX laboratory as a source of expertise in design and testing techniques. In fact, many organizations have created dedicated usability laboratories to provide expert reviews and to conduct usability tests of products during development in carefully supervised conditions (Rubin and Chisnell, 2008). Beyond internal usability teams, outside experts can sometimes provide fresh and unbiased insights on difficult design and usability decisions. These and other evaluation strategies are covered in Chapter 5.

Organizational support for usability testing is not sufficient, however, but should also include the creative parts of the design process. Each project should have its own user-interface architect who develops the necessary skills, manages the work of other people, prepares budgets and schedules, and coordinates with internal and external human-factors professionals when further expertise, references to the literature, or usability tests are required. Organizations with a strong design ethos understand this, and their example can be used in enacting change in more traditional corporations.

There are interaction design activities where the ROI for usability analysis during the development cycle is not immediately apparent but true usability of the delivered system is crucial for success. One familiar example is voting machines. An end result of confused, misinterpreted voting results would be catastrophic and counter to the best interests of the voting population, but the usability analysis and associated development costs should be manageable by the government contractor building the electronic voting system.

As the field of interaction design has matured, projects have grown in complexity, size, and importance. Role specialization is emerging, as it has in fields such as architecture, aerospace, and book design. Interaction design takes on new perspectives when writing web, mobile, or desktop applications, with an emerging discipline in translating the same information across each of these media. Eventually, individuals will become highly skilled in specific problem areas, such as user-interface–building tools, graphic display strategies, voice and audio design, shortcuts, navigation, and online tutorial writing. Consultation with graphic artists, book designers, advertising copywriters, textbook authors, game designers, or animators is expected. Perceptive system developers recognize the need to employ psychologists for conducting experimental tests, sociologists for evaluating organizational impact, educational psychologists for refining training procedures, and social workers for guiding customer-service personnel.

Usability engineers and *user-interface architects*, sometimes called the user experience (UX) team, are gaining experience in managing organizational change. As attention shifts away from software engineering or management-information systems, battles for control and power manifest themselves in budget and personnel allocations. Well-prepared managers who have a concrete organizational plan, defensible cost/benefit analyses, and practical development methodologies are most likely to be winners.

4.3 The Design Process

Design is inherently creative and unpredictable, regardless of discipline. In the context of interactive systems, successful designers blend a thorough knowledge of technical feasibility with an uncanny aesthetic sense of what attracts and satisfies users. One way to define design is by its operational characteristics (Rosson and Carroll, 2002):

- Design is a *process*; it is not a state, and it cannot be adequately represented statically.
- The design process is *nonhierarchical*; it is neither strictly bottom-up nor strictly top-down.
- The process is *radically transformational*; it involves the development of partial and interim solutions that may ultimately play no role in the final design.
- Design intrinsically involves the *discovery of new goals*.

These characterizations of design convey the dynamic nature of the process. An iterative design process based on this operational definition would consist of four distinct phases (Fig. 3.1): requirements analysis (Phase 1), preliminary and detailed design (Phase 2), build and implementation (Phase 3), and evaluation (Phase 4). This is a bare-bones process that describes its overall structure; individual applications of this process in specific design teams and for specific design artifacts will differ in terms of the frameworks, methods, and tools used. The primary feature of this process is that it is *iterative* and *cyclical*; unlike linear

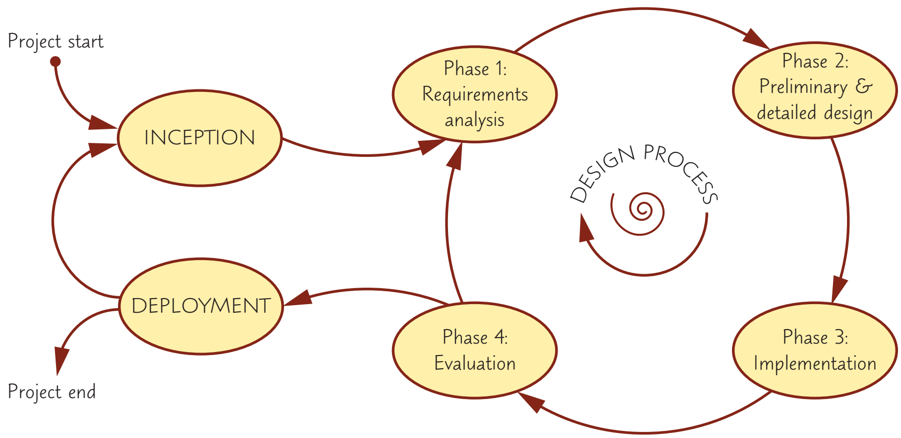


FIGURE 4.1

An iterative design process for interaction design.

waterfall models where one phase of a pipeline feeds the next, our design process repeats each phase over and over until the final product is of acceptable quality. Second, there are several cross-cutting factors that contribute to each phase of the cycle, including academic and user research, guidelines and standards, and tools and patterns. Each of these is described below.

Our focus here is purely on the human and social aspects of an interactive system or product, but the overall design process encompasses also technical aspects. Many technical design processes, such as in software engineering, follow a similar four-phase cycle, allowing interaction design and engineering to be integrated with them easily.

4.3.1 Phase 1: Requirements analysis

This phase collects all of the necessary requirements for an interactive system or device and yields a requirements specification or document as its outcome. In general, soliciting, capturing, and specifying user requirements are major keys to success in any development activity (Selby, 2007). Methods to elicit and reach agreement upon interaction requirements differ across organizations and industries, but the end result is the same: a clear specification of the user community and the tasks the users perform.

Collecting interaction design requirements is part of the overall requirements analysis and management phase and often has a direct impact on the engineering aspects of the design; for example, a finger painting app requires a multi-touch display with low touch latency. Thus, even requirements documents written specifically for user experience and interaction design aspects are often specified in terms of three components (see Box 4.1 for specific examples):

- *Functional requirements* define specific behavior that the system should support (often captured in so-called *use cases*, see below);
- *Non-functional requirements* specify overall criteria governing the operation of the interactive system without being tied to a specific action or behavior (hardware, software, system performance, reliability, etc.); and
- *User experience requirements* explicitly specify non-functional requirements for the user interaction and user interface of the interactive system (navigation, input, colors, etc.).

Requirements documents provide a shared understanding between the members of the product team. The success or failure of software projects often depends on the precision and completeness of this understanding between all the designers, developers, and users. What happens without adequate requirements definition? You are not sure what problem you are solving, and you do not know when you are done.

Box 4.1 gives an example of interaction design requirements for an e-commerce website, an ATM, and a mobile messaging app. Be careful not to

BOX 4.1

Examples of requirements regarding system behavior for three distinct types of interactive systems: an e-commerce website, an ATM, and a mobile messaging app.

Functional requirements:

- **Website:** The website shall allow users to purchase items and shall provide other, related merchandise based on past visits and purchases.
- **ATM:** The system shall let users enter a PIN code as identification and shall ensure that the code matches the one on file.
- **Mobile app:** The app shall be able to send messages at all times, even when out of the service area (in which case they are saved for later sending).

Non-functional requirements:

- **Website:** The website shall give users the ability to access their user account at all times, allowing them to view and modify name, mail address, e-mail address, phone, etc.
- **ATM:** The system shall permit the ATM customer 15 seconds to make a selection. The customer shall be warned that the session will be ended if no selection is made.
- **Mobile app:** Messages should send within 2 seconds, returning the user to the new message window (continuing in the background if necessary).

User experience requirements:

- **Website:** The website shall always have a visible navigation menu in the same position on the screen.
- **ATM:** On-screen prompts and instructions shall be clear and accessible. The ATM should return the user's commands within half a second.
- **Mobile app:** The mobile app shall support customization such as color schemes, skins, and sounds.

impose human operator actions (requirements) onto the interaction design requirements. For example, it is best not to specify a requirement like this: "The user shall decide how much to withdraw from the ATM within 15 seconds." Rather, allocate that same requirement to the computer system: "The ATM shall permit a user 15 seconds to select a withdrawal amount . . . before prompting for a response."

While it is possible to write functional requirements as simply an informal list of actions (as in Box 4.1), the concept of a *use case* from software engineering can come in handy here because of its direct connection to users and interaction. Put simply, a use case is a formalized scenario that captures an operation between an actor and the system (in general software engineering, the actor could be another system, but the focus is on human users here) in a step-by-step manner. The rule is that a system should simply be a sum of its use cases: No functionality should be implemented that does not explicitly support at least one use case. This also gives a straightforward recipe for evaluating the system (in Phase 4); if all use cases can be completed successfully, the system is correct and valid.

Several methods exist for actually collecting and analyzing interaction design requirements, including ethnographic observation, focus groups, and user interviews. Common among all of these is that they are intended to monitor the context and environment of real users, either in action or in their own words. Section 4.5 describes these methods in detail. Tradeoffs between what functions are done best by computers versus humans in human-computer interaction (Section 3.3.6) should also be discussed at this point in the development process.

4.3.2 Phase 2: Preliminary and detailed design

The core of the design process is realizing the requirements from the previous phase. The design phase in turn consists of two stages: a preliminary stage, where the high-level design or architecture of the interactive system is derived, and a detailed stage, where the specifics of each interaction are planned out. The outcome from the design phase is a detailed design document.

The preliminary design is also known as *architectural design*, and in engineering settings this stage often entails deriving the architecture of the system. For user experience and interaction design, preliminary design consists of mapping out the high-level concepts such as the user, controls, interface displays, navigation mechanisms, and overall workflow. Preliminary design can also be called *conceptual design*, particularly in software engineering, because it is sometimes useful to organize the high-level concepts into a conceptual map with their relations. Overall, this activity is about developing the mental model that users should have about the interactive system when using it. Is your system focused on a central view, such as a map or a table, or is it a sequence of forms or a set of linked displays? Is it an app that integrates with other apps to pop up on demand, or is it intended for focused, sustained use? These are questions to answer and refine during this stage.

The high-level concepts and their relations provide a starting point for the detailed design. This stage entails planning out all of the operations that take place between user and interactive system to a level where only implementation and technical details remain. Regardless whether you are using the use case concept discussed in the previous section, this can be done by creating and refining a step-by-step list for the exchanges between the user and the system.

One difficulty in designing interactive systems is that customers and users may *not* have a clear idea of what the system will look like when it is done. Since interactive systems are novel in many situations, users may not realize the implications of design decisions. Unfortunately, it is difficult, costly, and time-consuming to make major changes to systems once those systems have been implemented. Although this problem has no complete solution, some of the more serious difficulties can be avoided if, at an early stage, the customers and users can be given a realistic impression of what the final system will look like. Suitable methods for the design phase should thus go beyond eliciting the needs of the users and instead find ways to fulfill these needs.

Examples of suitable design methods include sketching, paper mockups, and high-fidelity prototypes. Furthermore, all methods can be informed through the use of tools, patterns, and best practices. For example, guidelines documents give direction on specific design choices, such as menu design, display layout, and navigation techniques. Patterns suggest effective ways to design an interface, such as single-page applications for websites or multi-document interfaces for desktop tools. Dedicated wireframing tools allow for rapidly creating mockups of a design. Section 4.6 discusses these tools and patterns in depth.

4.3.3 Phase 3: Build and implementation

The implementation phase is where all of the careful (or not very careful at all, depending on your design approach; see the agile development framework in Section 4.4.1) planning gets turned into actual, running code. The outcome from this phase is a working system, albeit not necessarily the final one. The actual software and hardware engineering needed to achieve this are outside the scope of this book. It is worth, however, briefly mentioning some suitable software development platforms for interactive applications based on your computing platform:

- *Mobile:* Building mobile apps typically requires using the SDK (software development kit) and development environment provided by the manufacturer of the operating system: the Android SDK in Java, the Apple iOS SDK in Objective-C, and the Windows Phone/Mobile SDKs. Most of these SDKs require registering as a developer to have access to the app exchange for making your app available to general users. Since mobile app development typically is *cross-platform*—the development is actually conducted on a personal computer—all of these SDKs include emulators for testing the app on a virtual phone residing on the personal computer itself.
- *Web:* The browser has become a ubiquitous information access platform, and modern web technologies are both pervasive and full-featured to the point that they can emulate or replace traditional computer software. Web applications and services typically consist of both client and server software: Client-side software runs in the user's browser and is accordingly built in JavaScript—the programming language of the browser—whereas server-side software runs on the web server or connected hosts and is often implemented in languages such as PHP, Ruby, Java, or even JavaScript (using Node.js).

A recent change in web development has been to build mobile apps using web technologies; the resulting app runs in a dedicated browser instance and is almost indistinguishable from a normal app built using the native SDK yet has the benefit of being cross-platform across different mobile operating systems.

- *Personal Computers:* Developing dedicated applications for a personal computer typically requires using the native SDKs for the specific operating system. Development environments such as Microsoft's Visual Basic/C++ are easy to get started with yet have an excellent set of features. C# and the .NET Framework are other good candidates for your project. For cross-platform software development that works regardless of operating system, Oracle's Java™ is a popular choice. People who want to write their own Java programs can use the Java Development Kit™ (JDK).

Regardless of platform, make sure to evaluate tool capabilities, ease of use, ease to learn, cost, and performance. Tailor your tool choices for the size of the job. Building a software architecture that supports your user-interface project is just as important as it is for any other (particularly large-scale) software development activity.

4.3.4 Phase 4: Evaluation

In the final phase of the design cycle, developers test and validate the system implementation to ensure that it conforms to the requirements and design set out earlier in the process. The outcome of the validation process is a validation report specifying test performance. As discussed above, a straightforward approach to validate a system specified using use cases is simply to check that each use case can be completed successfully. Since an interactive system is the sum of all of its conceivable user operations, such a test covers all of the system functionality. Depending on this outcome, the design team can decide to proceed with production and deployment of the system or to continue another cycle through the design process.

Validation is a vital part of the design process. Theatrical producers know that extensive rehearsals and previews for critics are necessary to ensure a successful opening night. Early rehearsals may involve only the key performers wearing street clothes, but as opening night approaches, dress rehearsals with the full cast, props, and lighting are required. Aircraft designers carry out wind-tunnel tests, build plywood mockups of the cabin layout, construct complete simulations of the cockpit, and thoroughly flight-test the first prototype. Similarly, website designers know that they must carry out many small and some large pilot tests of components before release to customers (Rubin and Chisnell, 2008). In addition to a variety of expert review methods, tests with the intended users, surveys, and automated analysis tools are proving to be valuable. Procedures vary greatly depending on the goals of the usability study, the number of expected users, the danger of errors, and the level of investment. Chapter 5 covers a range of suitable evaluation methods for this phase in depth.

4.4 Design Frameworks

While the design process discussed above generally should remain the same for all your projects, the approach to performing it may vary. The concept of *design frameworks* captures this idea: the specific flavor and approach the design takes to conducting the design process. More specifically, interaction design practice over the past few decades has unearthed several unique approaches to conducting the design process. This section reviews the concepts of user-centered design (UCD), participatory design (PD), and the nascent idea of agile interaction design.

4.4.1 User-centered design

Many software development projects fail to achieve their goals; some estimates of the failure rate put it as high as 50% (Jones, 2005). Much of this problem can be traced to poor communication between developers and their business clients or between developers and their users. The result is often systems and interfaces that force the users to adapt and change their behavior to fit the interface rather than an interface that is customized to the needs of the users.

User-centered design (UCD) is a counterpoint to this fallacy and prescribes a design process that primarily takes the needs, wants, and limitations of the actual end users into account during each phase of the design process (Lowdermilk, 2013). Directly involving the intended users in the process constantly challenges the assumptions of the design team about user behavior in the real world and gives designers a much-needed understanding of what their users actually need. In particular, careful attention to user-centered design issues during the early stages of software development dramatically reduces both development time and cost. UCD leads to systems that generate fewer problems during development and have lower maintenance costs over their lifetimes. They are easier to learn, result in faster performance, reduce user errors substantially, and encourage users to explore features that go beyond the minimum required to get by. Most importantly, UCD reduces the risk of designers building the "wrong system": a system that the end users neither need nor asked for. In addition, user-centered design practices help organizations align system functionality with their business needs and priorities.

While the main premise of UCD—user involvement—is straightforward, it is also its most significant challenge. For example, finding users may be difficult because of the need to select a manageable number of representative users, because the users may be unable or unwilling to participate, and because the users often lack the technical expertise needed to communicate effectively with the designers. Even when these challenges have been overcome, many users may not have a clear understanding of what they need in the new system or product. Successful developers work carefully to understand the business's needs and refine their skills in eliciting accurate requirements from non-technical business managers. In addition, since business managers may lack the technical knowledge to understand proposals made by the developers, dialogue is necessary to reduce confusion about the organizational implications of design decisions.

4.4.2 Participatory design

Going beyond user-centered design, *participatory design* (PD) (also known as *cooperative design* in Scandinavia) is the direct involvement of people in the collaborative design of the things and technologies they use. The arguments in favor suggest that more user involvement brings more accurate information about tasks and an opportunity for users to influence design decisions. The sense of participation that builds users' ego investment in successful implementation may be the biggest influence on increased user acceptance of the final system (Kujala, 2003; Muller and Druin, 2012). On the other hand, extensive user involvement may be costly and may lengthen the implementation period. It may also generate antagonism from people who are not involved or whose suggestions are rejected and potentially force designers to compromise their designs to satisfy incompetent participants.

Participatory design experiences are usually positive, however, and advocates can point to many important contributions that would have been missed without user participation. Many variations of participatory design have been proposed that engage participants to create dramatic performances, photography exhibits, games, or merely sketches and written scenarios. For example, users can be asked to sketch interfaces and use slips of paper, pieces of plastic, and tape to create low-fidelity early prototypes. A scenario walkthrough can be recorded on video for presentation to managers, users, or other designers. High-fidelity prototypes and simulations can also be key in eliciting user requirements.

Careful selection of users helps to build a successful participatory design experience. A competitive selection increases participants' sense of importance and emphasizes the seriousness of the project. Participants may be asked to commit to repeated meetings and should be told what to expect about their roles and their influence. They may have to learn about the technology and business plans of the organization and be asked to act as a communication channel to the larger group of users that they represent.

The social and political environment surrounding the implementation of complex interfaces is not amenable to study by rigidly defined methods or controlled experimentation. Social and industrial psychologists are interested in these issues, but dependable research and implementation strategies may never emerge. The sensitive project leader must judge each case on its merits and must decide on the correct level of user involvement. The personalities of the participatory design team members are such critical determinants that experts in group dynamics and social psychology may be useful as consultants. Many questions remain to be studied, such as whether homogeneous or diverse groups are more successful, how to tailor processes for small and large groups, and how to balance decision-making control between typical users and professional designers.

The experienced interaction designer knows that organizational politics and the preferences of individuals may be more important than technical issues in governing the success of an interactive system. For example, warehouse managers who see their positions threatened by an interactive system that provides senior managers with up-to-date information through digital displays may try to ensure that the system fails by delaying data entry or by being less than diligent in guaranteeing data accuracy. The interaction designer should take into account the system's effect on users and should solicit their participation to ensure that all concerns are made explicit early enough to avoid counterproductive efforts and resistance to change. Novelty is threatening to many people, so clear statements about what to expect can be helpful in reducing anxiety.

Ideas about participatory design are being refined with diverse users, ranging from children to older adults. Arranging for participation is difficult for some users, such as those with cognitive disabilities or those whose time is limited (for example, surgeons). The levels of participation are becoming clearer; one taxonomy describes the roles of children in developing interfaces for children, older adults in developing interfaces whose typical users will be other older adults, and so on, with roles varying from testers to informants to partners (Druin, 2002; Fig. 4.2). Testers are merely observed as they try out novel designs, while informants comment to designers through interviews and focus groups. The key characteristic of participatory design is that the design partners are active, first-class members of the product design team.

4.4.3 Agile interaction design

Traditional design processes can be described as heavyweight in that they require significant investments in time, manpower, and resources to be successful. In particular, such processes are often not sufficiently reactive to today's fast-moving markets and dynamic user audiences. Originally hailing from software engineering, *agile development* is a family of development methods for self-organizing, dynamic teams that facilitate flexible, adaptive, and rapid

FIGURE 4.2
Intergenerational and interdisciplinary design team from the University of Maryland's KidsTeam working on new human-computer interaction technologies using paper prototypes (http://hcil.umd.edu/children-as-design-partners/).

development that is robust to changing requirements and needs. These methods are based on *evolutionary development*, where software is built incrementally and in rapid release cycles. Similarly, *rapid prototyping* comes from manufacturing disciplines and describes a family of techniques for quickly fabricating physical parts or assemblies using computer-aided design (CAD). Both methods counter traditional heavyweight processes that have plagued design and facilitate a more flexible and, indeed, agile approach to design. Taken together, the methods can also be applied to interaction design to enable the rapid creation of interactive systems to meet user needs. In fact, taking users and usability into account during agile development may help to address a common weakness of agile development methods: Constant interface changes due to continuous iterative design may lead to inconsistent and confusing user experience poorly matched to the user.

Thus, agile interaction design uses lightweight design processes that facilitate the incremental and iterative nature of agile software developments. Instead of

FIGURE 4.3

Professor Jon Froehlich and his students working in the HCIL Hackerspace at University of Maryland, College Park.

costly and time-consuming documentation, high-fidelity prototypes, and usability evaluation and workshops that are common to heavyweight design processes, agile interaction design will use sketches, low-fidelity mockups, and fast usability inspections (Gundelsweiler et al., 2004). This enables practical and pragmatic design, short development cycles, and dynamic designs that are responsive to changing needs. Good resources for more information on agile interaction design and extreme usability (XU) can be found in Ambler (2002, 2008).

The contemporary "maker culture" movement on technology-based tinkering and manufacturing is a prime example of agile methods in action, where the focus is heavily on rapid and informal experimentation and prototyping by like-minded individuals gathering in so-called makerspaces, hackerspaces, or fablabs. Fig. 4.3 shows the Hackerspace at University of Maryland, College Park. For more information on maker culture, see Anderson (2014).

4.5 Design Methods

Design methods are the practical building blocks that form the actual day-to-day activities in the design process. There are dozens of design methods in the literature, but designers may want to focus on the most common ones (discussed below). See Holtzblatt and Beyer (2014) and Jacko (2012) for more details on specific methods or additional methods beyond these.

What is the relation between design frameworks and design methods? It is certainly true that specific design frameworks have an affinity to specific design methods; for example, participatory and user-centered design tends to incorporate a lot of ethnographic observation, whereas rapid and agile development employs sketching to a high degree. However, the design frameworks also provide a *flavor* for the overall process and each of the design methods: An agile approach to sketching will focus on collecting quick ideas from the design team, whereas a user-centered or participatory approach will let the intended users themselves be part of the sketching process. The description below discusses such variations and affinities.

4.5.1 Ideation and creativity

One way to think about design is as an incremental fixation of the solution space, where the range of possible solutions is gradually whittled down until only a single solution exists. This is the final product or service that then goes on to ship and be deployed. Gradually reducing the solution space in this manner is called *convergence* or *convergent thinking*, particularly for teams of designers

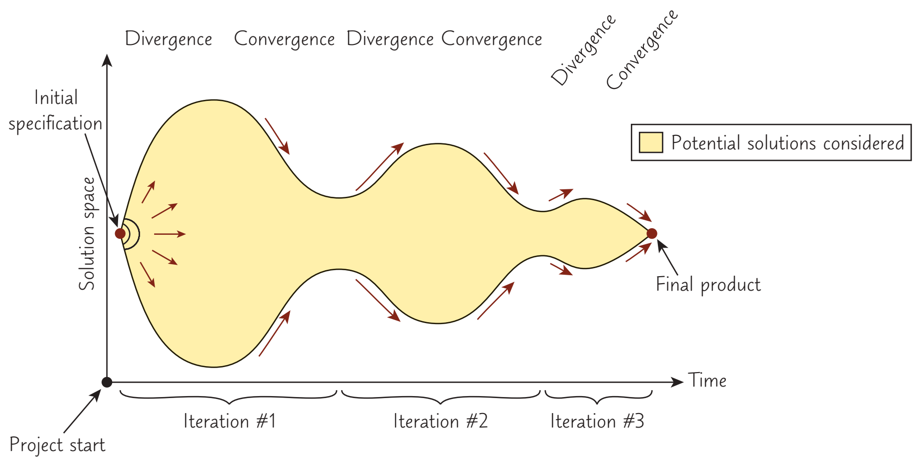


FIGURE 4.4

Illustration of how the solutions considered during a design process will grow (diverge) and shrink (converge) iteratively until they eventually fixate on a single point, the finished product. This particular design process involves three iterations, but real processes may have more or fewer iterations.

who each bring their own expertise and visions to the table. However, employing convergent thinking alone runs the risk of imposing conformance and uniformity too early in the process, yielding an end result that is a local rather than a global optimum. To avoid such local optima, there is a need to introduce *divergence* or *divergent thinking* into the design process as well (Löwgren, 2004). Fig. 4.4 demonstrates how interleaving divergent and convergent thinking iteratively can lead to a well-rounded and balanced design process that considers a large portion of the potential solution space.

Ideation (or idea generation) and creativity techniques are methods for such divergent thinking in that they require designers to test their limits, abandon their assumptions, and reframe their problems. Many creativity techniques exist in the literature, including lateral thinking, brainstorming and brainwriting, improvisation and role playing, and aleatoricism (incorporation of chance) and bootlegging (Holmquist, 2008). Common among many of these techniques is that they incorporate visual aids, sketching (Buxton, 2007), and physical artifacts. For example, brainstorming often results in *mind maps* that show the main concepts as bubbles with links describing relations between the concepts. The mere process of creating hastily drawn visual depictions of ideas and concepts—*sketching* (Buxton, 2007)—has been shown to facilitate both divergence and convergence by inviting both common ground and consensus as well as deviation and diversity. Fig. 4.5 shows two concept sketches of personal hovercrafts drawn by two different designers.

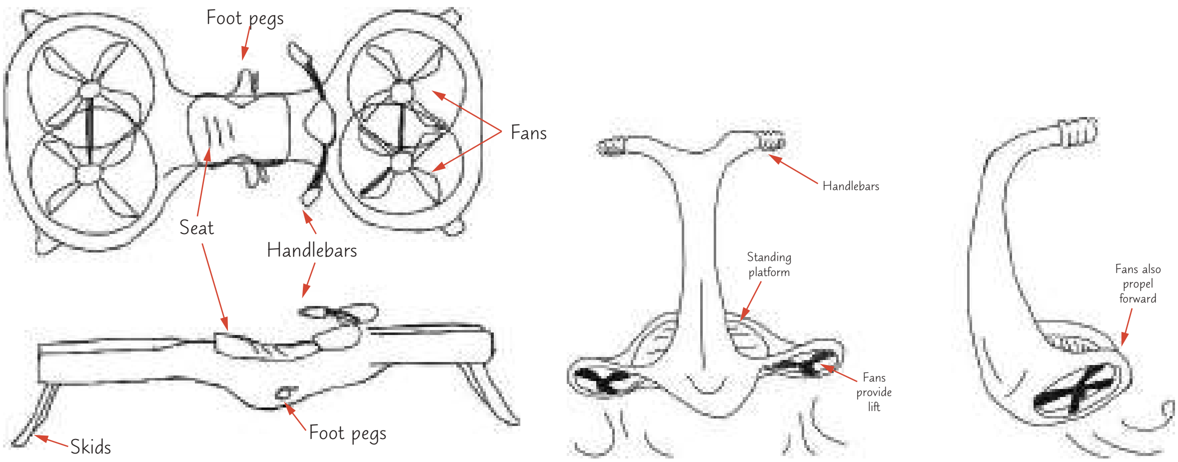


FIGURE 4.5

Concept sketches of personal hovercraft drawn by two different designers (divergence). In a follow-up step, the designers may work together to combine ideas from each separate sketch (convergence).

An in-depth discussion of ideation and creativity techniques is beyond the scope of this book. Many excellent resources on creativity exist; see, for example, the book by Buxton (2007).

4.5.2 Surveys, interviews, and focus groups

The most straightforward way to elicit requirements and desires from users is simply to ask them. Surveys—online or paper-based—are the simplest and cheapest approach and simply entail distributing a questionnaire to representative users. Online surveys can have significant reach but often yield a low response rate. Furthermore, the feedback received is often superficial in nature.

In-person interviews are more labor-intensive than surveys but will yield more accurate and high-quality responses. Interviews can take place either in one-on-one settings between designer and user or in *focus group* discussions with multiple users and designers. Again, the choice between individual or group interviews depends largely on cost; a group session requires less time investment but may not be able to collect in-depth feedback from all participants. On the other hand, sometimes group dynamics yield synergistic effects where one participant's response may trigger additional feedback from other participants. Thus, group interviews are often used in UCD and PD design frameworks.

Depending on the goal of the interview, the designer will take a structured or unstructured approach. *Structured interviews* are essentially verbal surveys, but the method often lets the designer follow up with additional questions based on the answer. *Unstructured interviews* have no specific questions to ask the user, only a general discussion topic. Successful designers often combine both strategies in the same session: Some questions are fixed, whereas some are more open-ended. This allows for collecting unsolicited feedback as well as getting answers to specific questions that the designer did not know to ask but that are important to the user.

A common problem with soliciting feedback directly from users is that they often don't really know what they want, either because they are too accustomed to the current way of doing things or because they don't know what is technologically possible (as well as impossible) to do. For this reason, the interaction designer often has to take the role of a therapist, trying to tease out deeper meaning and underlying motivation from the things that participants say. Furthermore, successful designers are often creative in interpreting participant responses and use them as a springboard for future ideas. For example, if several users mention that they tend to think of their job duties as a list of tasks to be crossed out one by one once completed, the designer may use the idea of a dynamic timeline of tasks as a central metaphor in future sketches and prototypes.

4.5.3 Ethnographic observation

The early stages of most methodologies include observation of users. Since interface users form a unique culture, ethnographic methods for observing them in the workplace are becoming increasingly important (Fig. 4.2). Ethnographers join work or home environments to listen and observe carefully, sometimes stepping forward to ask questions and participate in activities (Fetterman, 2009; Dourish and Bell, 2011; Bazeley, 2013). As ethnographers, interaction designers gain insight into individual behavior and the organizational context. However, they differ from traditional ethnographers in that, in addition to seeking understanding of their subjects, interaction designers focus on interfaces for the purpose of changing and improving those interfaces. Also, whereas traditional ethnographers immerse themselves in cultures for weeks or months, interaction designers usually need to limit this process to a period of days or even hours to obtain the relevant data needed to influence a redesign (Crabtree et al., 2012).

The goal of ethnographic observation for interaction design is to obtain the necessary data to influence interface redesign. Unfortunately, it is easy to misinterpret observations, to disrupt normal practice, and to overlook important information. Following a validated ethnographic process reduces the likelihood of these problems. Examples of ethnographic observation research include (1) how cultural probes have been adopted and adapted by the HCI community (Boehner et al., 2007), (2) development of an interactive location-based service for supporting distributed mobile collaboration for home healthcare (Christensen et al., 2007), and (3) social dynamics influencing technological solutions in developing regions (Ramachandran et al., 2007). Box 4.2 provides some guidelines for how to prepare for the evaluation, perform the field study, analyze the data, and report the findings.

These notions seem obvious when stated, but they require interpretation and attention in each situation. For example, understanding the differing perceptions that managers and users have about the efficacy of the current interface will alert you to the varying frustrations of each group. Managers may complain about the unwillingness of staff to update information promptly, but staff may be resistant to using the interface because the login process is tedious and time-consuming. Respecting the rules of the workplace is important for building rapport: In preparing for one observation, we appreciated that the manager called to warn us that graduate students should not wear jeans because the users were prohibited from doing so. Learning the technical language of the users is also vital for establishing rapport. It is useful to prepare a long list of questions that you can then filter down by focusing on the proposed goals. Awareness of the differences between user communities, such as those mentioned in Chapter 2, will help to make the observation and interview process more effective.

BOX 4.2

Guidelines for conducting ethnographic studies for interaction design.

Preparation

- Understand policies in the target environment (work, home, public space, etc.).
- Familiarize yourself with the existing interface and its history.
- Set initial goals and prepare questions.
- Gain access and permission to observe or interview.

Field study

- Establish a rapport with all users.
- Observe or interview users in their setting, and collect subjective and objective quantitative and qualitative data.
- Follow any leads that emerge from the visits.
- Record your visits.

Analysis

- Compile the collected data in numerical, textual, and multimedia databases.
- Quantify data and compile statistics.
- Reduce and interpret the data.
- Refine the goals and the process used.

Reporting

- Consider multiple audiences and goals.
- Prepare a report and present the findings.

Data collection can include a wide range of subjective impressions that are qualitative or of subjective reactions that are quantitative, such as rating scales or rankings. Objective data can consist of qualitative anecdotes or critical incidents that capture user experiences or can be quantitative reports about, for example, the number of errors that occur during a one-hour observation of six users. Deciding in advance what to capture is highly beneficial, but remaining alert to unexpected happenings is also valuable. Written report summaries have proved to be valuable far beyond expectations; in most cases, raw transcripts of every conversation are too voluminous to be useful.

Making the process explicit and planning carefully may seem awkward to many people whose training stems from computing and information technology. However, a thoughtfully applied ethnographic process has proved to have many benefits. It can increase trustworthiness and credibility, since designers learn about the complexities of the intended environment by visits to the workplace, school, home, or other environment where the eventual system will be deployed. Personal presence allows designers to develop working relationships with several end users to discuss ideas, and, most importantly, the users may consent to be active participants in the design of their new interface.

4.5.4 Scenario development and storyboarding

Scenario development builds on the use case concept and allows for developing specific scenarios when a user engages the interactive system to solve a particular task. Storyboarding is the use of graphical sketches and illustrations to convey important steps in a scenario (Fig. 4.6). Several additional methods for scenario development are useful. Often, a flowchart or transition diagram helps designers to record and convey the sequences of possible actions; the thickness of the connecting lines indicates the frequency of the transitions. An easy way to describe a novel system is to write scenarios of usage and then, if possible, to act them out as a form of theater. This technique can be especially effective when multiple users must cooperate (for example, in control rooms, cockpits, or financial trading rooms) or multiple physical devices are used (for example, at customer-service desks, medical laboratories, or hotel check-in areas). Scenarios can represent common or emergency situations with both novice and expert users. Personas can also be included in scenario generation.

Some scenario writers take a further step and produce videos to convey their intentions. In 2011, Corning Incorporated released a futuristic video named “A Day Made of Glass” that showed a vision for how interactive surfaces can become increasingly incorporated into both our professional and

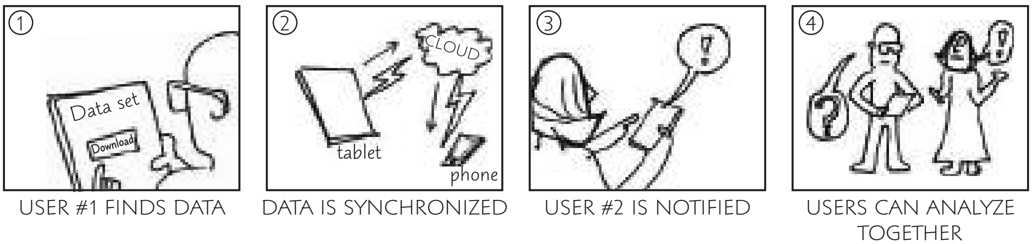


FIGURE 4.6

Hand-drawn storyboard for collaborative software that allows multiple people to view a common dataset using their personal smartphones and tablets.

everyday lives in the near future. Similarly, in 2013, "Future Vision 2020" showed a similar vision of the future with integrated displays everywhere. Finally, in 2015, Microsoft released the video "Microsoft: Productivity Future Vision," which followed several individuals across the globe as they went through their day five to ten years in the future collecting, analyzing, and summarizing their data in mostly professional settings. All three videos feature large, transparent, and touch-sensitive displays as well as many personal, small, thin, and even flexible displays that are all seamlessly connected and integrated with each other. Taken together, these videos all point to an increasingly augmented digital future where computing continues to become part of the fabric of everyday life.

4.5.5 Prototyping

Prototypes, or *physical sketches* as Buxton (2007) calls them, are particularly powerful design tools because they allow users and designers alike to see and hold (for physical prototypes) representations of the intended interface. They also allow the design team to play out specific scenarios and tests using the prototype. For example, a printed version of the proposed displays can be used for pilot tests, whereas an interactive display with an active keyboard and mouse can be used for more realistic tests. Fig. 4.7 shows a hand-drawn sketch for a mouse equipped with a touch display, and Fig. 4.8 represents the corresponding physical prototype that has been constructed simply by attaching a smartphone to an off-the-shelf mouse.

Increasing realism, or *fidelity*, for a prototype governs the time investment in creating it. Obviously, low-fidelity prototypes are more suitable for early ideation and creativity because they are easily generated and as easily

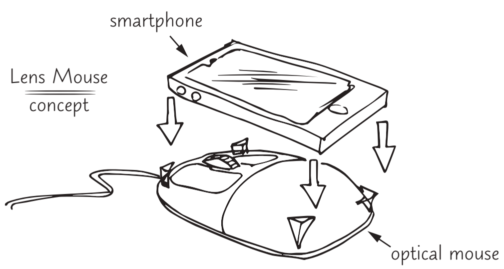


FIGURE 4.7

Hand-drawn concept sketch for a so-called LensMouse: a mouse that incorporates a touch display. The sketch suggests constructing the mouse by simply attaching a smartphone to a traditional optical mouse.

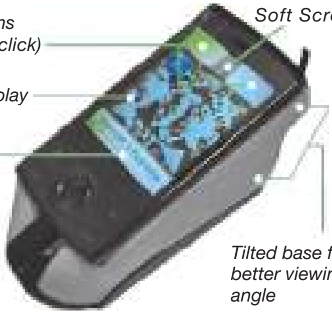


FIGURE 4.8

Prototype of the LensMouse. This high-fidelity prototype was created by simply attaching a smartphone on top of a standard wireless optical mouse (Yang et al., 2010).

discarded; in fact, the very vagueness of a "quick and dirty" sketch communicates the uncertainty of the ideation process and invites improvements or rejection. Here are some examples of prototypes at different levels of fidelity:

- **Low-fidelity prototypes** are generally created by sketching, using sticky notes, or cutting and gluing pieces of paper together (paper mockups);
- **Medium-fidelity prototypes** are often called *wireframes*, provide some standardized elements (such as buttons, menus, and text fields), even if potentially drawn in a sketchy fashion, and have some basic navigation functionality; and
- **High-fidelity prototypes** look almost like the final product and may have some rudimentary computational capabilities; however, the prototype is typically not complete and may not be fully functional.

4.6 Design Tools, Practices, and Patterns

Beyond the theoretical frameworks and methods discussed above, design activities today are supported by current design practice: tools that have arisen to support many design methods, guidelines and standards to inform working designers, and patterns that provide reusable solutions to commonly occurring problems encountered by interaction designers.

4.6.1 Design tools

Creating prototypes beyond paper mockups requires using computer programs to prototype a specific interface or app. The simplest approach is to use general-purpose drawing and drafting applications for this purpose. For example, prototypes have been developed with simple drawing or word-processing tools or even Microsoft PowerPoint® presentations of screen drawings manipulated with PowerPoint slideshows and other animation. Other design tools that can be used are Adobe InDesign®, Photoshop®, or Illustrator®.

Dedicated prototyping design tools are specifically designed for the purpose of creating interface mockups rapidly and effortlessly. Since the visual design language varies across different platforms, different tools exist for desktop, mobile, and web. Many design tools use the actual buttons, dropdown menus, and scrollbars used in the interfaces on the specific platform. However, this has the danger of looking "too polished" and suggesting to the user that the interface mockup is final and cannot be changed. To avoid this, design tools such as Balsamiq Mockups (Fig. 6.4 shows Balsamiq being used in a case study by Volvo) use a sketchy, hand-drawn look for the interface elements. Similar to a hastily sketched design on the back of a napkin, the purpose of the hand-drawn look is to convey to the viewer that a design mockup is still a sketch, that it does not represent the final version of the interface, and that it can still be changed with little cost or time investment.

Finally, dedicated design tools—so-called graphical user-interface builders—also exist for the final implementation phase when the development team is realizing the planned interface. Many of these builders use a drag-and-drop graphical editor where the interaction designer can construct the final interface by assembling existing interface elements from a library of elements. Builders often automatically generate the necessary source code from the graphical specification, requiring the developer only to write his or her own source code to manage the events resulting from the user interaction.

4.6.2 Design guidelines and standards

Early in the design process, the interaction design team should generate a set of working guidelines. Two people might work for one week to produce a 10-page document, or a dozen people might work for two years to produce a 300-page document. One component of Apple's success with the original Macintosh was the machine's early and readable guidelines document, which provided a clear set of principles for the many application developers to follow and thus ensured harmony in design across products. Microsoft's *Windows User Experience Guidelines*, which have been refined over the years, also provide a good starting point and an educational experience for many programmers.

These and other guidelines documents are referenced and described briefly in the general reference section at the end of Chapter 1. The Eight Golden Rules of interface design in Section 3.3.4 are also applicable to most interactive systems.

Guidelines documents are a powerful tool for interaction design because they

- Provide a social process for developers;
- Record decisions for all parties to see;
- Promote consistency and completeness;
- Facilitate automation of design;
- Allow multiple levels:
  - Rigid standards
  - Accepted practices
  - Flexible guidelines
  - Industry-specific guidelines
- Announce policies for:
  - Education: How to get it?
  - Enforcement: Who reviews?
  - Exemption: Who decides?
  - Enhancement: How often?

Guidelines creation (Box 4.3) is often a social process within an organization in order to help gain visibility and build support for the guidelines. Controversial guidelines (for example, on when to use voice alerts) can be reviewed by colleagues or tested empirically. Procedures should be established to distribute the guidelines, to ensure enforcement, to allow exemptions, and to permit enhancements. Effective guidelines documents are living texts that are adapted to changing needs and refined through experience. Acceptance may be increased by a three-level approach of rigid standards, accepted practices, and flexible guidelines. This approach clarifies which items are firmer and which items are susceptible to change.

The creation of a guidelines document at the beginning of an implementation project focuses attention on the interface design and provides an opportunity for discussion of controversial issues. When the development team adopts the guidelines, the implementation proceeds quickly and with few design changes. For large organizations, there may be two or more levels of guidelines to provide organizational identity while allowing projects to have distinctive styles and local control of terminology. Some organizations develop "style guides" to capture this (see, for example, Microsoft, 2014).

BOX 4.3

Suggested content in user experience guidelines documents.

Words, icons, and graphics

- Terminology (objects and actions), abbreviations, and capitalization
- Character set, fonts, font sizes, and styles (bold, italic, underline)
- Icons, buttons, graphics, and line thickness
- Use of color, backgrounds, highlighting, and blinking

Display layout issues

- Menu selection, form fill-in, and dialog-box formats
- Wording of prompts, feedback, and error messages
- Justification, white space, and margins
- Data entry and display formats for items and lists
- Use and contents of headers and footers
- Strategies for adapting to very small and very large displays

Input and output devices

- Keyboard, display, cursor control, and pointing devices
- Sound, voice feedback, speech I/O, touch input, etc.
- Response times for a variety of tasks
- Alternatives for users with disabilities

Action sequences

- Direct-manipulation clicking, dragging, dropping, and gestures
- Command syntax, semantics, and sequences
- Shortcuts and programmed function keys
- Touch input for devices such as smartphones, tablets, and large touch displays
- Error handling and recovery procedures

Training

- Online help, tutorials, and support groups
- Training and reference materials

The "four Es" provide a basis for creating a living document and a lively process:

- *Education*. Users need training and a chance to discuss the guidelines. Developers must be trained in the resultant guidelines.
- *Enforcement*. A timely and clear process is necessary to verify that an interface adheres to the guidelines.

- *Exemption*. When creative ideas or new technologies are used, a rapid process for gaining exemption is needed.
- *Enhancement*. A predictable process for review, possibly annually, will help keep the guidelines up-to-date.

While creating and using guidelines documents help ensure success—to the point that we propose our own Eight Golden Rules (Section 3.3.4)—we also want to reiterate our argument from Chapter 3 that the discipline of human-computer interaction needs to transcend specific guidelines and derive basic theories underlying these phenomena. Our discussion in Section 3.4 presents several both micro-level and macro-level HCI theories to which most practical guidelines can be traced back. While a list of guidelines can be highly useful precisely because they are practical and pragmatic, successful designers remain aware of the underlying theories from where they stem.

4.6.3 Interaction design patterns

Design patterns, originally proposed for urban planning (Alexander, 1977) and later software engineering (Freeman et al., 2004), are best-practice solutions to commonly occurring problems specified in such a way that they can be reused and applied to slightly different variations of a problem over and over again. Regardless of discipline, patterns help address a common problem for novice designers: They have very little experience of past work to draw upon when tackling a new problem. In this way, design patterns constitute valuable experience-in-a-can, ready to be used when needed.

While software engineering design patterns are quite technical in nature, they are particularly useful for the interaction designer with a software engineering bent; see, for example, Freeman et al. (2004) for details. In fact, user-interface toolkits were one of the original inspirations for software engineering design patterns, and, accordingly, many of the original 23 design patterns deal directly with user interfaces, such as Decorator, Composite, and Command. As a result, several of these patterns are manifested in modern user-interface toolkits.

The fact that patterns were originally used to solve problems in urban planning demonstrates that the pattern concept transcends specific disciplines, and the idea has further been applied to areas such as pedagogy, game design, communication policy, visualization, and even chess strategy. Analogously, a pattern approach to interaction design would suggest reusable solutions to commonly occurring problems in user-interface and interaction design. While an exhaustive discussion of this topic would likely require an entire book of its own (Tidwell, 2005), here is a list of a few useful interaction design patterns along these lines:

- *Model-View-Controller (MVC).* A so-called *architectural pattern* for implementing user interfaces, MVC governs how information should flow between three specific components in the interface: models that represent the state (e.g., a string for an input field or a number for a dial), views that render the state on the display (e.g., the text box or the spinner), and controllers that change the models (e.g., editing the string or increasing/decreasing the number) as well as the views (e.g., scrolling through a long document).

- *Document interface.* Many applications, particularly those designed for personal computers, allow opening more than one document at the same time. Document interface patterns capture different ways of managing multiple documents for an application:
  - *Single document interface (SDI).* The simplest document interface pattern, each document opens a new instance of the application. Mobile apps and web applications tend to be built using this pattern.
  - *Multiple document interface (MDI).* Here each document opens an internal window in the main frame, allowing for a single application window even for multiple open documents. Common in personal computer applications.
  - *Tabbed document interface (TDI).* A compromise between SDI and MDI, the tabbed document interface pattern places multiple open documents in *tabs* in a single instance of the application. Most web browsers use TDI.
- *Web application page architecture.* Designing a web application is subtly different than designing an application for a personal computer or a mobile device. The page architecture is one of the most important interaction design aspects here:
  - *Multi-page application (MPA).* The traditional way of building a web application is to use multiple pages, one for each specific function in the application. This mimics dialog boxes in desktop applications and is easy to implement by the very nature of HTML and the web, which is organized into separate pages but requires reloading for each page and may thus cause disruption in the user experience.
  - *Single-page application (SPA).* These applications fit on a single webpage, thus mimicking a desktop application, and require no reloading or mode changes, thereby making the user experience fluid and uninterrupted. Instead of page loads, the application state changes dynamically through communication with the web server using modern web technologies such as JavaScript, HTML, and CSS.

Further discussion of interaction design patterns is beyond the scope of this book. The reader may want to refer to Tidwell (2005) for more on this topic. Also of interest is Schell and O'Brien's review (2015) of 13 so-called *anti-patterns*—straightforward or seemingly good ideas that ultimately do not tend to work out—in the context of user experience.

4.7 Social Impact Analysis

Interactive systems often have a dramatic impact on large numbers of users. To minimize risks, a thoughtful statement of anticipated impacts circulated among stakeholders can be a useful process for eliciting productive suggestions early in the development when changes are easiest.

Governments, utilities, and publicly regulated industries increasingly require information systems to provide services. However, some critics have strong negative attitudes toward modern technologies and see only a hopeless technological determinism: "Technopoly eliminates alternatives to itself. It consists in the deification of technology, which means that the culture seeks its authorization in technology, finds its satisfactions in technology, and takes its orders from technology" (Postman, 1993).

Postman's endless fears do not help us to shape more effective technology or to prevent damage from technology failures. However, constructive criticism and guidelines for design could be helpful in reversing the long history of incorrect credit histories, dislocation through de-skilling or layoffs, and deaths from flawed medical instruments. Current concerns focus on privacy invasion from surveillance systems, government attempts to restrict access to information, and voting fraud because of poor security. While guarantees of perfection are not possible, policies and processes can be developed that will more often than not lead to satisfying outcomes.

A *social impact statement*, similar to an environmental impact statement, might help to promote high-quality systems in government-related applications (reviews for private-sector corporate projects would be optional and self-administered). Early and widespread discussion can uncover concerns and enable stakeholders to state their positions openly. Of course, there is the danger that these discussions will elevate fears or force designers to make unreasonable compromises, but these risks seem reasonable in a well-managed project. An outline for a social impact statement might include these sections (Shneiderman and Rose, 1996):

- Describe the new system and its benefits.
  - Convey the high-level goals of the new system.
  - Identify the stakeholders.
  - Identify specific benefits.
- Address concerns and potential barriers.
  - Anticipate changes in job functions and potential layoffs.
  - Address security and privacy issues.
  - Discuss accountability and responsibility for system misuse and failure.

- Avoid potential biases.
- Weigh individual rights versus societal benefits.
- Assess tradeoffs between centralization and decentralization.
- Preserve democratic principles.
- Ensure diverse access.
- Promote simplicity and preserve what works.

- Outline the development process.
    - Present an estimated project schedule.
    - Propose a process for making decisions.
    - Discuss expectations of how stakeholders will be involved.
    - Recognize needs for more staff, training, and hardware.
    - Propose a plan for backups of data and equipment.
    - Outline a plan for migrating to the new system.
    - Describe a plan for measuring the success of the new system.

A social impact statement should be produced early enough in the development process to influence the project schedule, system requirements, and budget. It can be developed by the system design team, which might include end users, managers, internal or external software developers, and possibly clients. Even for large systems, the social impact statement should be of a size and complexity that make it accessible to users with relevant backgrounds.

After the social impact statement is written, it should be evaluated by the appropriate review panel as well as by managers, other designers, end users, and anyone else who will be affected by the proposed system. Potential review panels include federal government units (for example, the General Accounting Organization or Office of Personnel Management), state legislatures, regulatory agencies (for example, the Securities and Exchange Commission or the Federal Aviation Administration), professional societies, and labor unions. The review panel will receive the written report, hold public hearings, and request modifications. Citizen groups also should be given the opportunity to present their concerns and to suggest alternatives.

Once the social impact statement is adopted, it must be enforced. A social impact statement documents the intentions for the new system, and the stakeholders need to see that those intentions are backed up by actions. Typically, the review panel is the proper authority for enforcement.

The effort, cost, and time involved should be appropriate to the project, while facilitating a thoughtful review. The process can offer large improvements by preventing problems that could be expensive to repair, improving privacy protection, minimizing legal challenges, and creating more satisfying work environments. Information-system designers take no Hippocratic Oath, but pledging themselves to strive for the noble goal of excellence in design can win respect and inspire others.

4.8 Legal Issues

As user interfaces have become more prominent in society, serious legal issues have emerged. Every developer of software and information should review legal issues that may affect design, implementation, deployment, marketing, and use. This section merely touches upon the most important such concerns. For more information, Baase (2013) gives an in-depth overview of such social, legal, philosophical, ethical, political, constitutional, and economic implications of computing.

Privacy and security are always a concern whenever computers are used to store data or to monitor activity. Medical, legal, financial, and other data often have to be protected to prevent unapproved access, illegal tampering, inadvertent loss, or malicious mischief. Recently implemented privacy assurance laws such as those imposed on the medical and financial communities can lead to complicated, hard-to-understand policies and procedures. Physical security measures to prohibit access are fundamental; in addition, privacy protection can involve user-interface mechanisms for controlling password access, identity checking, and data verification. Effective protection provides a high degree of privacy with a minimum of confusion and intrusion into work. Website developers should provide easily accessible and understandable privacy and security policies.

A second concern encompasses safety and reliability. User interfaces for aircraft, automobiles, medical equipment, military systems, utility control rooms, and the like can affect life-or-death decisions. If air traffic controllers are confused by the situation display, they can make fatal errors. If the user interface for such a system is demonstrated to be difficult to understand, it could leave the designer, developer, and operator open to a lawsuit alleging improper design. Designers should strive to make high-quality and well-tested interfaces that adhere to state-of-the-art design guidelines and requirements. Accurate records documenting testing and usage will protect designers in case problems arise.

A third issue is copyright or patent protection for software (Lessig, 2006; Samuelson and Schultz, 2007; McJohn, 2015). Software developers who have spent time and money developing a package are understandably frustrated when potential users make illegal copies of the package rather than buying it. Technical schemes have been tried to prevent copying, but clever hackers can usually circumvent the barriers. It is unusual for a company to sue an individual for copying a program, but cases have been brought against corporations and universities. There is also a vocal community of developers, led by the League for Programming Freedom, that opposes software copyright and patents, believing that broad dissemination is the best policy. An innovative legal approach, Creative Commons™, enables authors to specify more liberal terms for others to use their works. The open source software movement has enlivened these controversies. The Open Source Initiative describes the movement as follows: "When programmers can read, redistribute, and modify the source code for a piece of software, the software evolves. People improve it, people adapt it, people fix bugs. And this can happen at a speed that, if one is used to the slow pace of conventional software development, seems astonishing." Some open source products, such as the Linux® operating system and the Apache™ web server, have become successful enough to capture a substantial portion of the market share.

A fourth concern is with copyright protection for online information, images, or music. If customers access an online resource, do they have the right to store the information electronically for later use? Can the customer send an electronic copy to a colleague or friend? Who owns the "friends" list and other shared data in social networking sites? Do individuals, their employers, or network operators own the information contained in e-mail messages? The expansion of the web, with its vast digital libraries, has raised the temperature and pace of copyright discussions. Publishers seek to protect their intellectual assets, while librarians are torn between their desire to serve patrons and their obligations to publishers. If copyrighted works are disseminated freely, what incentives will there be for publishers and authors? If it is illegal to transmit any copyrighted work without permission or payment, science, education, and other fields will suffer. The fair use doctrine of limited copying for personal and educational purposes helped cope with the questions raised by photocopying technologies. However, the perfect rapid copying and broad dissemination permitted by the Internet demand a thoughtful update (Samuelson, 2003; Lessig, 2006).

A fifth issue is freedom of speech in electronic environments. Do users have a right to make controversial or potentially offensive statements through e-mail or social media? Are such statements protected by freedom of speech laws, such as the U.S. First Amendment? Are networks similar to street corners, where freedom of speech is guaranteed, or are networks similar to television broadcasting, where community standards must be protected? Should network operators be responsible for or prohibited from eliminating offensive or obscene jokes, stories, or images? Controversy has raged over whether Internet service providers have a right to prohibit e-mail messages that are used to organize consumer rebellions against themselves. Another controversy emerged over whether a network operator has a duty to suppress racist e-mail remarks or postings to a social media platform. For example, Twitter has been commonly used by racists, bullies, and terrorist organizations. If libelous statements are transmitted, can a person sue the network operator as well as the source? Should designers build systems where the default is to "opt out" of lists and users have to explicitly "opt in" by making a selection from a dialog box?

Other legal concerns include adherence to laws requiring equal access for users with disabilities and attention to changing laws in countries around the world. Do Yahoo! and eBay have to enforce the laws of every country in which they have customers? These and other issues mean that developers of online services must be sure to consider all the legal implications of their design decisions.

The Internet Association (http://internetassociation.org/), the spiritual successor to the venerable NetCoalition, is a collective political lobbying organization in Washington, DC, that monitors many of the legal issues raised here. Founded by Amazon, eBay, Facebook, and Google, its website is an excellent source for information about privacy legislation and related issues. For the international level, the Electronic Frontier Foundation (https://www.eff.org/), founded in 1990, is a non-profit digital rights group providing support to individuals fighting corporations and governments against baseless or misdirected legal threats. There are also many other legal issues to be aware of today, including anti-terrorism, counterfeiting, spam, spyware, liability, Internet taxation, and others. These issues certainly require your attention, and legislation may eventually be needed.

Practitioner's Summary

Interaction design is maturing rapidly, with once-novel ideas becoming standard practices. Design has increasingly taken center stage in organizational and product planning. Development frameworks such as user-centered, participatory, and agile design help by offering validated processes with predictable schedules and meaningful deliverables. Specific design methods such as surveys, focus groups, and ethnographic observation can provide information to guide requirements analysis. Usage logs provide valuable data about task frequencies and sequences. Scenario writing helps to bring common understanding of design goals, is useful for managerial and customer presentations, and helps to plan usability tests. For interfaces developed by governments, public utilities, and regulated industries, an early social impact statement can elicit public discussion that is likely to identify problems and produce interfaces that have high overall societal benefits. Designers and managers should obtain legal advice to ensure adherence to laws and protection of intellectual property.

Researcher's Agenda

Human-computer interaction guidelines are often based on best-guess judgments rather than on empirical data. More research could lead to refined standards that are more complete and dependable and to more precise knowledge of how much improvement can be expected from a design change. Deriving the underlying micro-HCI and macro-HCI theories from which practical guidelines are drawn would have far-reaching consequences for interaction design. In particular, because technology is continually changing, we will never have a stable and complete set of guidelines, but such scientific theories will allow us to retain reliability and quality of interface design. It would also enable evolving design processes, ethnographic methods, participatory design activities, scenario writing, and social impact statements to address emergent issues such as international diversity, special populations such as children or older adults, and long-term studies of actual usage. Thoughtful case studies of design processes would lead to their refinement and promote more widespread application. Creative processes are notoriously difficult to study, but well-documented examples of success stories will inform and inspire.

Discussion Questions

1. You are the new Chief Design Officer (CDO) of a start-up, DTUI Inc. The project is to design a system for selling pottery. The aim is to develop an interface that meets the needs of both the potters and the customers. Describe in detail a design methodology of four stages to facilitate the proper design of such a system.

   Write your answer in the form of a management plan for this project. For each stage, indicate the number of weeks that should be allocated. Hint: Note the four phases of the design process:

   - Requirements analysis
   - Preliminary (conceptual) and detailed design
   - Build and implementation
   - Evaluation

For questions 2–4, refer to the following scenario:
The State of Maryland is developing a web-voting interface. For selecting the candidates, one design (RB) is a set of radio buttons and another is (CB) a

combo-box (drops down when selecting the scroll arrow icon), both using standard fonts at 10-point size.

2. Compare these two designs when there are four candidates and predict the relative speed of performance and error rates. Support your choice by a thoughtful argument.
3. An expert reviewer complains that both designs may work with young users who are familiar and expert in using a mouse, but that there will be problems for elderly and motor-impaired users who have difficulty controlling a mouse. The reviewer recommends a new design that includes a larger font (20-point size) and a numbered list to allow selection by keyboard easily. Describe a participatory design or social impact statement process that might clarify this issue with elderly users.
4. Design an experiment to help resolve the issue brought up in question 3. Assume you have substantial resources and access to subjects.
5. What is user-centered design? What are its benefits?
6. Consider a system that does not yet exist; for example, a totally automated fast-food restaurant, where customers order via touch screen interactions, pay by swiping their debit or credit cards, and then pick up their food—analogous to the self-check-out at some supermarkets, but even more extreme. Discuss how you would conduct a contextual inquiry for a system that does not yet exist.

CASE STUDY: Evaluate an existing system where you would propose improvements to the user interface. For example, visit the local mass transit station (e.g., the subway) and observe users purchasing tickets. If possible, interview users regarding their preferences for improvements in usability. One method to facilitate user cooperation may be to solicit fellow students or colleagues who are new to your community and have not yet taken a ride on the subway system. Plan for contextual interviews by developing questionnaires. Conduct the contextual interviews, then tabulate the results and analyze them with an eye to understanding the demographics and skills of the user community. Then follow the remaining steps outlined in this list, eventually developing storyboards and prototypes to evaluate with users.

References

Alexander, Christopher, *A Pattern Language: Towns, Buildings, Construction*, Oxford University Press, USA (1977).

Ambler, Scott W., *Agile Modeling*, John Wiley and Sons (2002).

Ambler, Scott W., Tailoring usability into Agile Software Development projects, in *Maturing Usability, Human-Computer Interaction Series*, Springer Verlag (2008), 75–95.

Anderson, Chris, *Makers: The New Industrial Revolution*, Crown Business (2014).

Baase, Sara, *A Gift of Fire: Social, Legal, and Ethical Issues for Computing Technology*, 4th Edition, Pearson Education (2013).

Bazeley, Patricia, *Qualitative Data Analysis: Practical Strategies*, SAGE Publications (2013).

Bias, Randolph, and Mayhew, Deborah, *Cost-Justifying Usability: An Update for the Internet Age*, 2nd Edition, Morgan Kaufmann, San Francisco, CA (2005).

Boehner, Kirsten, Vertesi, Janet, Sengers, Phoebe, and Dourish, Paul, How HCI interprets the probes. In *Proceedings of the ACM Conference on Human Factors in Computing Systems,* ACM Press, New York (2007), 1077–1086.

Buxton, Bill, *Sketching User Experiences: Getting the Design Right and the Right Design*, Morgan Kaufmann, San Francisco, CA (2007).

Christensen, Claus M., Kjeldskov, Jesper, and Rasmussen, Klaus K., GeoHealth: A location-based service for nomadic home healthcare workers, In *Proceedings of the ACM Conference of the Computer-Human Interaction Special Interest Group (OZCHI) of Australia on Computer-Human Interaction*, ACM Press, New York (2007), 273–281.

Crabtree, Andrew, Rouncefield, Mark, and Tolmie, Peter, *Doing Design Ethnography, Human-Computer Interaction Series*, Springer Verlag (2012).

Dourish, Paul, and Bell, Genevieve, *Divining a Digital Future: Mess and Mythology in Ubiquitous Computing*, MIT Press, Cambridge, MA (2011).

Druin, Allison, The role of children in the design of new technology. *Behaviour & Information Technology 21,* 1 (2002), 1–25.

Fetterman, David M., *Ethnography: Step by Step*, 3rd Edition, SAGE Publications, Thousand Oaks, CA (2009).

Freeman, Eric, Robson, Elisabeth, Sierra, Kathy, and Bates, Bert, *Head First Design Patterns*, O'Reilly Media (2004).

Gundelsweiler, Fredrik, Memmel, Thomas, and Reiterer, Harald, *Agile Usability Engineering*. In Keil-Slawik, Reinhard, Selke, Harald, and Szwillus, Gerd (Editors), *Mensch & Computer: Allgegenwärtige Interaktion* (2004), 33–42.

Holmquist, Lars Erik, Bootlegging: Multidisciplinary brainstorming with cut-ups. In *Proceedings of the Conference on Participatory Design* (2008) 158–161.

Holtzblatt, Karen, and Beyer, Hugh, *Contextual Design: Evolved, Synthesis Lectures on Human-Centered Informatics,* Morgan & Claypool, San Rafael, CA (2014).

Jacko, Julie (Editor), *The Human-Computer Interaction Handbook*, CRC Press, Boca Raton, FL (2012).

Jones, Capers, A CAI State of the Practice Interview, Computer Aid, Inc. (July 2005). Available at http://web.ecs.baylor.edu/faculty/grabow/Fall2007/COMMON/Secure/Refs/capersjonesinterview1.pdf.

Karat, Claire-Marie, A business case approach to usability, In Bias, Randolph, and Mayhew, Deborah (Editors), *Cost-Justifying Usability*, Academic Press, New York (1994), 45–70.

Kujala, Sari, User involvement: A review of the benefits and challenges. *Behaviour & Information Technology 22*, 1 (2003), 1–16.

Landauer, Thomas K., *The Trouble with Computers: Usefulness, Usability, and Productivity*, MIT Press, Cambridge, MA (1995).

Lessig, Lawrence, *Code and Other Laws of Cyberspace, Version 2.0*, Basic Books, New York (2006).

Lowdermilk, Travis, *User-Centered Design: A Developer's Guide to Building User-Friendly Applications*, O'Reilly Media (2013).

Löwgren, Jonas, and Stolterman, Erik, *Thoughtful Interaction Design: A Design Perspective on Information Technology*, MIT Press (2004).

Marcus, Aaron. Return on investment for usable user-interface design: Examples and statistics (2002). Available at http://www.amanda.com/resources/ROI/AMA_ROIWhitePaper_28Feb02.pdf

McJohn, Stephen M., *Examples & Explanations: Intellectual Property*, 5th Edition, Wolters Kluwer, New York, NY (2015).

Microsoft, Inc., *Microsoft Windows 8 Design and Coding Guidelines* (2014). Available at http://go.microsoft.com/fwlink/p/?linkid=258743.

Moggridge, Bill, *Designing Interactions*. MIT Press, Cambridge, MA (2007).

Muller, Michael J., and Druin, Allison, Participatory design: The third space in human-computer interaction. In Jacko, Julie (Editor), *The Human-Computer Interaction Handbook*, CRC Press, Boca Raton, FL (2012), 1125–1154.

National Research Council, Committee on Human Factors, Committee on Human-System Design Support for Changing Technology, Pew, Richard W., and Mavor, Anne S. (Editors), *Human-System Integration in the System Development Process: A New Look*, National Academies Press, Washington, DC (2007).

Nielsen, Jakob, Usability ROI declining, but still strong (2008). Available at http://www.useit.com/alertbox/roi.html.

Postman, Neil, *Technopoly: The Surrender of Culture to Technology*, Vintage Books, New York (1993).

Ramachandran, Divya, Kam, Matthew, Chiu, Jane, Canny, John, and Frankel, James F., Social dynamics of early stage co-design in developing regions. In *Proceedings of the ACM Conference on Human Factors in Computing Systems*, ACM Press, New York (2007), 1087–1096.

Rosson, Mary Beth, and Carroll, John M., *Usability Engineering: Scenario-Based Development of Human Computer Interaction*, Morgan Kaufmann, San Francisco, CA (2002).

Rubin, Jeffrey, and Chisnell, Dana, *Handbook of Usability Testing: How to Plan, Design, and Conduct Effective Tests*, 2nd Edition, John Wiley & Sons, New York (2008).

Samuelson, Pamela, Digital rights management {and, or, vs.} the law, *Communications of the ACM 46*, 4 (April 2003), 41–45.

Samuelson, Pamela, and Schultz, Jason, Should copyright owners have to give notice about their use of technical protection measures? *Journal of Telecommunications & High Technology Law 6* (2007), 41–76.

Selby, Richard W. (Editor), *Software Engineering: Barry W. Boehm's Lifetime Contributions to Software Development, Management, and Research*, John Wiley & Sons, New York, NY (2007), 663–685.

Schell, Martina, and O'Brien, James, *Communicating the UX Vision: 13 Anti-Patterns That Block Good Ideas*, Morgan Kaufmann (2015).

Shneiderman, Ben, and Rose, Anne, Social impact statements: Engaging public participation in information technology design, In *Proceedings of the ACM SIGCAS Symposium on Computers and the Quality of Life*, ACM Press, New York (1996), 90–96.

Sullivan, Louis H., The tall office building artistically considered, *Lippincott's Magazine*, 403–409 (March 1896).

Tidwell, Jenifer, *Designing Interfaces: Patterns for Effective Interaction Design*, O'Reilly Media, Sebastopol, CA (2005).

Yang, Xing-Dong, Mak, Edward, McCallum, David, Irani, Pourang, and Izadi, Shahram, LensMouse: Augmenting the mouse with an interactive touch display. In *Proceedings of the ACM Conference on Human Factors in Computing Systems*, ACM Press, New York (2010), 2431–2440.

This page intentionally left blank


Noldus

CHAPTER 5

Evaluation and the User Experience

"The test of what is real is that it is hard and rough . . . .
What is pleasant belongs in dreams."

Simone Weil
Gravity and Grace, 1947

CHAPTER OUTLINE

5.1 Introduction

Designers can become so entranced with their creations that they may fail to evaluate them adequately. Experienced designers have attained the wisdom and humility to know that extensive testing and evaluation are necessities. If feedback is the "breakfast of champions," then testing and evaluation is the "dinner of the gods." However, careful choices must be made from the large menu of evaluation possibilities to create a balanced meal.

There are many factors that influence when, where, and how evaluation is performed within the development cycle. Some sample factors include the following:

- Stage of design (early, middle, late)
- Novelty of the project (well-defined versus exploratory)
- Number of expected users
- Criticality of the interface (for example, life-critical medical system versus museum-exhibit system)
- Costs of the product and finances allocated for testing
- Time available
- Experience of the design and evaluation team
- Environment where interface is used

The range of evaluation plans might be anywhere from an ambitious two-year test with multiple phases for a new national air-traffic-control system to a one-day test with six users for a small internal website. The range of costs can vary from a small amount to a larger and substantial cost. Testing should occur at different times in the evaluation cycle, ranging from early to just before release.

A few years ago, considering evaluating usability was seen as a good idea that might help you get ahead of the competition. However, the rapid growth of interest in the user experience means that failing to test is now risky indeed. Not only has the competition strengthened, but customary design practice now requires adequate testing and follow-through with recommended changes as appropriate and as time and budgeting permit. Failure to perform and document testing as well as not heeding the changes recommended from the evaluation process could lead to failed contract proposals or malpractice lawsuits from users where errors arise that may have been avoided if the problems had been detected and changes made.

One troubling aspect of testing is the uncertainty that remains even after exhaustive testing by multiple methods. Perfection is not possible in complex human endeavors, so planning must include continuing methods to assess and repair problems during the life cycle of an interface. Second, even though problems may continue to be found, at some point a decision has to be made about completing prototype testing, moving forward with the final design, and delivering the product. Third, most testing methods will account appropriately for normal usage, but performance is extremely difficult to test in unpredictable situations or times with high levels of input, such as nuclear reactor control, air-traffic-control emergencies, or heavily subscribed voting times (e.g., presidential elections). Development of testing methods to deal with stressful situations and even with partial equipment failures will have to be undertaken as user interfaces are developed for an increasing number of life-critical applications. Traditional lab testing (Section 5.3) may not accurately and with sufficient fidelity represent the high-stress and often hostile environments in which systems developed for healthcare providers, first responders, or the military are employed. Likewise, testing a global-positioning driving system will not work in a laboratory or other stationary location; it can only be tested out in the field. Some special medical devices may also need to be tested in their natural environments, such as a hospital, an assisted living facility, or even a private home. Mobile devices are better evaluated in their natural contexts as well. Evaluations might need to be done "in-the-wild" as field studies creating situations where the evaluator may not be nearby recording and observing (Rogers et al., 2013).

Discussions about the best ways to do usability testing and how to report the results generate lively debate among researchers. The choice of evaluation methodology (Vermeeren et al., 2010) must be appropriate for the problem or research question under consideration. Usability evaluators must broaden their methods and be open to non-empirical methods such as user sketches (Greenberg et al., 2012) and ethnographic studies. Producing sketches of possible user-interface designs, similar to the design sketches used by architects, is one interesting approach. This allows more alternatives to be explored in the early stage before the design becomes permanent.

Usability is about more than just ease of use; the entire user experience needs to be considered. A portion of that is defining whether the system is useful (MacDonald and Atwood, 2014). Today, complex systems exist that are hard to test with simple controlled experiments. Active discussions continue concerning the number of users (Schmettow, 2012; Kohavi et al., 2013) that participate in a usability study. Although the number of participants adds power and strength to the recommendations that come forth from usability studies, it is equally important to focus on common tasks and potentially troublesome tasks. Usability and user experience must be viewed as a multi-dimensional concept from varying perspectives. Testing novel devices, such as direct touch tabletops, may require special considerations. Usability inspection techniques may have to be modified to take into account the concept of shared and personal spaces when using large displays. Devices today range from the very small up to wall-size and even mall-size, and today's users are sophisticated with high levels of expectations based upon a multitude of previous experiences. Being aware that some systems are used by thousands, even millions, of users can affect the usability testing process and the user experience.

Usability testing has become an established and accepted part of the design process (see Chapter 4), but it needs to be broadened and understood in the context of today's highly sophisticated systems, a diversity of users with high expectations, mobile and other innovative devices (such as gaming systems and controllers), and competition and speed in the marketplace. A series of usability evaluations and related analyses have been conducted over the years by Rolf Molich, referred to as the Comparative Usability Evaluation (CUE) studies (http://www.dialogdesign.dk/CUE.html). These findings have shown that the number of usability problems in a website is so large, only a fraction of the problems will be found, and even professional usability evaluators can make mistakes in the evaluation process. Spool (2007) suggests three radical changes to the usability evaluation process: (1) stop making recommendations and instead present observational findings, (2) stop conducting evaluations and push the research onto the design team, and (3) seek out new techniques because new tools are needed. Others studies look at the relevance of empirical studies (Bargas-Avila and Hornbæk, 2011) and the move from quantitative data to qualitative data (Dimond et al., 2012). As HCI is maturing, perspectives are changing (Roto and Lund, 2013). Experiential computing requires an expanded perspective to include situated, cultural, emotional, and phenomenological aspects. More studies are being done in real-life environments. Researchers are still measuring, but now adding social dimensions and affective states including fun, emotion, enjoyment, and creating a fulfilling user experience. An interesting 30-year history of usability and lessons learned for future usability testing is presented by Lewis (2014). This is an exciting and provocative time in usability and user experience evaluation; practitioners should heed this advice, look closely at current procedures, and continue to grow in the area of user experience.

This chapter is organized as follows. Section 5.2 discusses expert reviews and heuristics, including heuristics for specialized devices like mobile and gaming. Section 5.3 covers conventional usability labs and the spectrum of usability testing. Section 5.4 provides some advice on survey instruments. Section 5.5 covers acceptance testing with Section 5.6 continuing with evaluation during

See also:

Chapter 1, Usability of Interactive Systems

Chapter 2, Universal Usability

Chapter 4, Design

Chapter 13, The Timely User Experience

active use and beyond. Finally, the chapter concludes with Section 5.7, which covers controlled psychologically oriented experiments.

5.2 Expert Reviews and Heuristics

A natural starting point for evaluating new or revised interfaces is to present them to colleagues or customers and ask for their opinions. Such informal demos with test subjects can provide some useful feedback, but more formal so-called *expert reviews* have proven to be far more effective. These methods depend on having experts (whose expertise may be in the application or user-interface domain) available on staff or as consultants. The reviews can then be conducted rapidly and on short notice by having the expert walk through the key functionality of the interface using a disciplined approach.

Expert reviews can occur early or late in the design phase. The outcome may be a formal report with problems identified or recommendations for changes. Alternatively, the expert review may culminate in a discussion with or presentation to designers or managers. Expert reviewers should be sensitive to the design team's ego, involvement, and professional skill; suggestions should be made cautiously in recognition of the fact that it is difficult for someone freshly inspecting an interface to fully understand the design rationale and development history. When reviewing complex interfaces, such as gaming applications, domain expertise can be a critical component (Barcelos et al., 2012). The reviewers can note possible problems to discuss with the designers, but development of solutions generally should be left to the designers.

Expert reviews usually take from half a day to one week, although a lengthy training period may be required to explain the task domain or operational procedures. It may be useful to employ the same expert reviewers as well as fresh ones as the project progresses. There are a variety of expert-review methods from which to choose.

Heuristic evaluation. The expert reviewers critique an interface to determine conformance with a short list of design heuristics, such as the Eight Golden Rules (Section 3.3.4). It makes an enormous difference if the experts are familiar with the rules and are able to interpret and apply them. Although interfaces have changed vastly over the years, the creation of most sets of heuristics is based on those proposed by Nielsen (1994). Today, there are many different types of devices that may be subject to a heuristic evaluation, and it is important that the heuristics match the application. Box 5.1 lists some heuristics developed specifically for video games. A similar set of 29 *playability heuristics* also exists. This set splits the heuristics into three categories: game usability, mobility heuristics, and gameplay heuristics (Korhonen and Koivisto, 2006). Gameplay heuristics are

BOX 5.1

Heuristics for the gaming environment (Pinelle et al., 2008).

- Provide consistent responses to user's actions.
- Allow users to customize video and audio setting, difficulty, and game speed.
- Provide predictable and reasonable behavior for computer controlled units.
- Provide unobstructed views that are appropriate for the user's current actions.
- Allow users to skip non-playable and frequently repeated content.
- Provide intuitive and customizable input mappings.
- Provide controls that are easy to manage and that have an appropriate level of sensitivity and responsiveness.
- Provide users with information on game status.
- Provide instructions, training, and help.
- Provide visual representations that are easy to interpret and that minimize the need for micromanagement.

the most difficult to evaluate because familiarity with all aspects of the game is required. Using the heuristics to follow good interaction design principles while maintaining the challenge and suspense of the game is a difficult balance. Other specialized heuristics exist, such as for mobile app design (Joyce et al., 2014) and interactive systems (Masip et al., 2011)

Guidelines review. The interface is checked for conformance with the organizational or other guidelines document (see Chapter 1 for a list of organizational guidelines documents and Section 3.2 and Chapter 4 for more on guidelines). Because guidelines documents may contain a thousand items or more, it may take the expert reviewers some time to absorb them and days or weeks to review a large interface.

Consistency inspection. The experts verify consistency across a family of interfaces, checking the terminology, fonts, color schemes, layout, input and output formats, and so on, within the interfaces as well as any supporting materials. Software tools (Section 5.6.5) can help automate the process as well as produce concordances of words and abbreviations. Often large-scale interfaces may be developed by several groups of designers; this can help smooth over the interface and provide a common and consistent look and feel.

Cognitive walkthrough. The experts simulate users walking through the interface to carry out typical tasks. High-frequency tasks are a starting point, but rare critical tasks, such as error recovery, also should be walked through. Some form of simulating a day in the life of a user should be part of the expert review process. Cognitive walkthroughs were initially developed for interfaces that can be learned by exploratory browsing (Wharton et al., 1994), but they are useful even for interfaces that require substantial training. An expert might try the walkthrough privately and explore the system, but there also should be a group meeting with designers, users, or managers to conduct a walkthrough and provoke discussion. Extensions can cover website navigation and incorporate richer descriptions of users and their goals. Newer walkthrough models include the collaborative critique method assessing the user's cognitive and physical effort with the interaction (Babaian et al., 2012).

Formal usability inspection. The experts hold a courtroom-style meeting, with a moderator or judge, to present the interface and to discuss its merits and weaknesses. Design-team members may rebut the evidence about problems in an adversarial format. Formal usability inspections can be educational experiences for novice designers and managers, but they may take longer to prepare and need more personnel to carry out than do other types of review.

Expert reviews can be scheduled at several points in the development process, when experts are available and when the design team is ready for feedback. The number of expert reviews will depend on the magnitude of the project and on the amount of resources allocated. Often a domain expert might review the tool, but be aware the expert may not be skilled in the tool itself.

An expert review report should aspire to comprehensiveness rather than making opportunistic comments about specific features or presenting a random collection of suggested improvements. The evaluators might use a guidelines document to structure the report, then comment on novice, intermittent, and expert features and review consistency across all displays, paying attention to ensure that the usability recommendations are both useful and usable. Some suggestions for writing effective usability recommendations can be found in Box 5.2.

If the report ranks recommendations by importance and expected effort level, managers are more likely to implement them (or at least the high-payoff, low-cost ones). For example, in one expert review, the highest priority was to shorten a three- to five-minute login procedure that required eight dialog boxes and passwords on two networks. The obvious benefit to already over-busy users was apparent, and they were delighted with the improvement. Common middle-level recommendations include reordering the sequence of pages, providing improved instructions or feedback, and removing non-essential actions. Expert reviews should also include required small fixes such as spelling mistakes, poorly aligned data-entry fields, or inconsistent button placement. A final category includes less vital fixes and novel features that can be addressed in the next version of the interface.

Expert reviewers should be placed in a situation as similar as possible to the one that intended users will experience. They should take training courses, read the documentation (if it exists), take tutorials, and try the interface in as close as possible to a realistic work environment, complete with noise and distractions.

BOX 5.2

Making usability recommendations useful and usable (Molich et al., 2007).

- Communicate each recommendation clearly at the conceptual level.
- Ensure that the recommendation improves the overall usability of the application.
- Be aware of the business or technical constraints.
- Show respect for the product team's constraints.
- Solve the whole problem, not just a special case.
- Make recommendations specific and clear.
- Avoid vagueness by including specific examples in your recommendations.

However, expert reviewers may also retreat to a quieter environment for a detailed and extensive review of the entire interface.

Another approach, getting a bird's-eye view of an interface by studying a full set of printed pages laid out on the floor or pinned to walls, has proved to be enormously fruitful in detecting inconsistencies and spotting unusual patterns. The bird's-eye view enables reviewers to quickly see if the fonts, colors, and terminology are consistent and whether the multiple developers have adhered to a common style.

Expert reviewers may also use software tools to speed their analyses, especially with large and complex interfaces. Sometimes string searches on design documents, help text, or program code can be valuable, but more specific interface-design analyses—such as web-accessibility validation, privacy-policy checks, and download-time reduction—are growing more effective. A further discussion of automated tools can be found in Section 5.6.5.

The danger with expert reviews is that the experts may not have an adequate understanding of the task domain or user communities. Different experts tend to find different problems in an interface, so involving three to five experts in the review can be highly productive. Usability testing can offer additional advice and should be used as a necessary complement. Experts come in many flavors, and conflicting advice can further confuse the situation (cynics say, "For every Ph.D., there is an equal and opposite Ph.D."). To strengthen the possibility of successful expert review and an enhanced user experience, it helps to choose knowledgeable experts who are familiar with the project and task domain and who have a long-term relationship with the organization. These people can be called back to see the results of their intervention, and they can be held accountable. However, even experienced expert reviewers have difficulty knowing how typical users—especially first-time users—will behave.

5.3 Usability Testing and Laboratories

The emergence of usability testing and laboratories since the early 1980s is an indicator of the profound shift in attention toward user experience and user needs. Traditional managers and developers resisted at first, saying that usability testing seemed like a nice idea but that time pressures or limited resources prevented them from trying it. As experience grew and successful projects gave credit to the testing process, demand swelled and design teams began to compete for the scarce resource of the usability laboratory staff. Managers came to realize that having a usability test on the schedule was a powerful incentive to complete a design phase. The usability test report provided supportive confirmation of progress and specific recommendations for changes. Designers sought the bright light of evaluative feedback to guide their work, and managers saw fewer disasters as projects approached delivery dates. The remarkable surprise was that usability testing not only sped up many projects but also produced dramatic cost savings (Rubin and Chisnell, 2008; Lund, 2011; Hartson and Pyla, 2012). As a matter of fact, the words *usability*, *usability testing*, and *user experience (UX)* have now made their way into our common vocabulary.

Usability laboratory advocates split from their academic roots as these practitioners developed innovative approaches that were influenced by advertising and market research. While academics were developing controlled experiments to test hypotheses and support theories, practitioners developed usability-testing methods to refine user interfaces rapidly. Controlled experiments (Section 5.7) have at least two treatments and seek to show statistically significant differences; usability tests are designed to find flaws in user interfaces. Both strategies use a carefully prepared set of tasks, but usability tests have fewer participants (maybe as few as three), and their outcome is a report with recommended changes as opposed to validation or rejection of a hypothesis. The move to gather qualitative data is taking a larger role in the user evaluation process. Sometimes because of the novelty or size of the device, conventional testing tasks may not be appropriate. Of course, there is a useful spectrum of possibilities between rigid controls and informal testing, and sometimes a combination of approaches is appropriate while always keeping the user experience in mind.

5.3.1 Usability labs

The movement toward usability testing stimulated the construction of usability laboratories (Nielsen, 1993; Rubin and Chisnell, 2008). Having a physical laboratory makes an organization's commitment to usability clear to employees, customers, and users. A typical modest usability laboratory would have two 10-by-10-foot areas, divided by a half-silvered mirror—one for the participants to do their work and the other for the testers and observers (designers, managers, and customers). IBM was an early leader in developing usability laboratories. Microsoft started later but has wholly embraced the idea with many usability test labs. Many other software development companies have followed suit, and a consulting community that will do usability testing for hire also has emerged. See Fig. 5.1 for a layout of a typical usability lab.

Usability laboratories are typically staffed by one or more people with expertise in testing and user-interface design who may serve 10 to 15 projects per year throughout an organization. The laboratory staff meet with the user experience architect or manager at the start of the project to make a test plan with scheduled dates and budget allocations. Usability laboratory staff members participate in early task analysis or design reviews, provide information on software tools or

FIGURE 5.1

Noldus Usability Lab

The usability lab consists of two areas, the testing room and the observation room. The testing room is typically smaller and accommodates a small number of people. Those in the observation room can see into the testing room typically via a one-way mirror. The observation room is larger and can hold the usability testing facilitators with ample room to bring in others, such as the developers of the product being tested. There may be recording equipment as well.

literature references, and help to develop the set of tasks for the usability test. Two to six weeks before the usability test, the detailed test plan is developed; it contains the list of tasks plus subjective satisfaction and debriefing questions. The number, types, and sources of participants are also identified—sources might be customer sites, temporary personnel agencies, or advertisements placed in newspapers. A pilot test of the procedures, tasks, and questionnaires with one to three participants is conducted approximately one week before the test, while there is still time for changes. This typical preparation process can be modified in many ways to suit each project's unique needs. Fig. 5.2 provides a detailed breakdown of steps to follow when conducting usability assessments.

After changes are approved, participants are chosen to represent the intended user communities, with attention to their backgrounds in computing, experience with the task, motivation, education, ability with the natural language used in the interface, and familiarity with the environment. Usability laboratory staff also must control for physical concerns (such as eyesight, left- versus right-handedness, age, gender, education, and computer experience) and for other experimental conditions (such as time of day, day of the week, physical

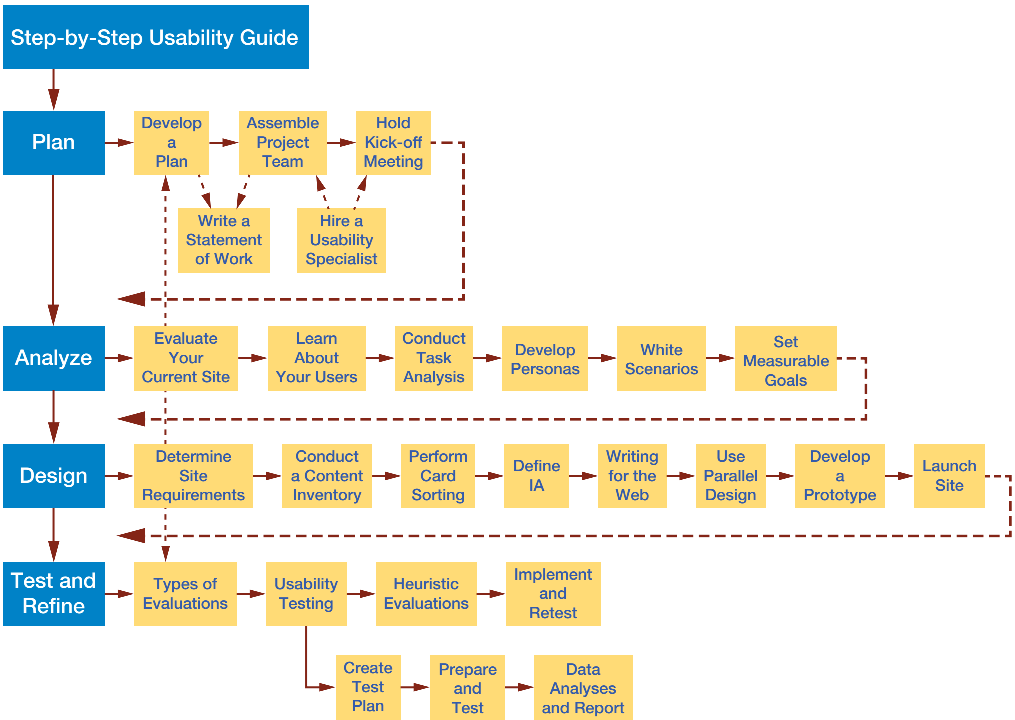


FIGURE 5.2

Step-by-Step Usability Guide

This guide from Usability.gov shows all the steps from planning a usability test to performing the actual test and reporting the results.

surroundings, noise, room temperature, and level of distractions). The main goal is to find participants who are representative of the intended user audience.

Recording participants performing tasks is often valuable for later review and for showing designers or managers the problems that users encounter. Reviewing the recordings is a tedious job, so careful logging and annotation (preferably automatic) during the test are vital to reduce the time spent finding critical incidents. Most usability laboratories have acquired or developed software to facilitate logging of user activities (e.g., typing, mousing, reading displays, reading manuals, etc.) by observers with automatic timestamping. Some of the more popular data-logging tools include Adobe Prelude Live Logger, Morae from TechSmith®, LogSquare from Mangold, Bit Debris, Observert XT from Moldus, and Ovo Logger. Participants may be anxious about the recording process at the start of the test, but within minutes they usually focus on the tasks and ignore the recording process. The reactions of designers seeing the actual recordings of users failing with their interfaces are sometimes powerful and may be highly motivating. When designers see participants repeatedly picking the wrong menu item, they often realize that the label or placement needs to be changed.

Another useful technique available to today's usability evaluation professional is *eye-tracking* hardware and software (Fig. 5.3). The eye-tracking data can

FIGURE 5.3

This shows a picture of the glasses worn for eye-tracking. This particular device is being used to track the participant's eye movements when using a mobile device. Tobii is one of several manufacturers. (Tobii AB)

FIGURE 5.4
The eye-tracking software is attached to the airline check-in kiosk. It allows the designer to collect data observing how the user "looks" at the screen. This helps determine if various interface elements (e.g., buttons) are difficult (or easy) to find. (Tobii AB)

show where participants gazed at the display and for how long. The results are displayed in color-coded heat maps (Fig. 14.3) that clearly demonstrate which areas of the display are viewed and which areas are ignored. This software has dropped in price substantially and become less cumbersome; it can now be supplied as a simple and affordable add-on to a computing device (Fig. 5.4). When testing with small mobile devices, special equipment may be needed to capture the user's device and associated activities (Fig. 5.5). Sometimes with mobile and other technology platforms, appropriate testing may need to be done "in the wild" with different constraints and procedures (Rogers et al., 2013).

At each design stage, the interface can be refined iteratively and the improved version can be tested. It is important to fix even small flaws (such as spelling errors or inconsistent layout) quickly because they influence user expectations.

5.3.2 Ethics in research practices with human participants

Participants should always be treated with respect and should be informed that it is not they who are being tested; rather, it is the software and user interface that are under study. They should be told about what they will be doing (for example, finding products on a website, creating a diagram using a mouse, or studying a restaurant guide on a touchscreen) and how long they will be expected to stay. Participation should always be voluntary, and informed consent in research is important (Box 5.3). Sometimes deception may need to be included in an experiment to fully test the hypothesis. Ethical practices would allow this as long as the benefits outweighed any potential or real harm.

In the United States, the Institutional Review Board (IRB) governs any research conducted on university campuses with human participants. There are different levels of review and precise procedures that must be followed. Special populations may also have unique considerations that need to be attended to. Most universities have a representative who can explain these procedures in detail. Other institutions and organizations have guidelines on ethical research practices with human participants.

FIGURE 5.5
Special mobile camera to track and record activities on a mobile device. Note the camera is up and out of the way, allowing the user to use his or her normal finger gestures to operate the device.
(© by Noldus Information Technology)

5.3.3 Think-aloud and related techniques

An effective technique during usability testing is to invite users to *think aloud* (sometimes referred to as *concurrent think-aloud*) about what they are doing as they are performing the task. The designer or tester should be supportive of the participants, not taking over or giving instructions but prompting and listening for clues about how they are dealing with the interface. Think-aloud protocols yield interesting clues for observant usability testers; for example, they may hear comments such as “This webpage text is too small . . . so I’m looking for something on the menus to make the text bigger . . . maybe it’s on the top in the icons . . . I can’t find it . . . so I’ll just carry on.”

After a suitable time period for accomplishing the task list (provided as part of the evaluation protocol)—usually one to three hours—the participants can be

BOX 5.3

Informed consent guidelines (Dumas and Loring, 2008).

Each informed consent statement should contain:

- The purpose of the study (an explanation of why the study is being done).
- The procedure being used for the study. This section should also include a time expectation for the participant and the protocol for requesting a break.
- If there will be any type of recording, who will see the recordings, and what happens to the recording material when the testing is completed (not all studies involve recordings).
- A statement of confidentiality and how the anonymity of the participant is preserved.
- Any risks to the participant (in most usability studies there is minimal risk).
- The fact that participation is voluntary and that the participant can withdraw at any time with no penalty.
- Whom to contact with questions and for any further information after the study and a statement that initial questions about the testing have been answered satisfactorily.

The informed consent statement should be signed prior to the start of any testing.

invited to make general comments or suggestions or to respond to specific questions. The informal atmosphere of a think-aloud session is pleasant and often leads to many spontaneous suggestions for improvements. In their efforts to encourage thinking aloud, some usability laboratories have found that having two participants working together produces more talking, as one participant explains procedures and decisions to the other (see Fig. 5.6). Researchers need to be aware that people may not always say exactly what they are thinking. Also, describing their thoughts can alter the process.

Another related technique is called *retrospective think-aloud*. With this technique, after completing a task, users are asked what they were thinking as they performed the task. The drawback is that the users may not be able to wholly and accurately recall their thoughts after completing the task; however, this approach allows users to focus all their attention on the tasks they are performing and generates more accurate timings. Two other variants include *concurrent probing* and *retrospective probing*. These techniques both interfere with any traditional user-interface measurements and take the participant away from the task at hand but provide insight into the user's thinking process.

FIGURE 5.6

Having people work in pairs gives the additional advantage of having some insight into the thought process as they discuss it and an unobstructed view into their information transfer channels (speech and body language, etc.). Coupling this with a pattern (Elmqvist and Yi, 2012) such as *pair analytics*, a system can be evaluated in the early formative stages.

It is important to consider timing when using think-aloud techniques. The standard think-aloud procedure may alter the true task time, as verbalizing the thought process creates additional cognitive load, and the users may pause the task activity as they vocalize their thoughts. Think-aloud can also be used when doing expert reviews. Retrospective think-aloud procedures will not alter the task timings themselves, but because the users need to perform the tasks and then reflect on and review them again, their overall time commitment may be doubled. Also, be aware that using the think-aloud technique along with eye-tracking may generate invalid results: Users' eyes may wander while they are speaking, causing spurious data to be generated.

5.3.4 The spectrum of usability testing

Usability testing comes in many different flavors and formats. Most of the current research demonstrates the importance of testing often and at varied times during the design cycle. The purpose of the test and the type of data that is needed are important considerations. Testing may be done at the exploratory stage, when the designers are trying to conceive the correct design, or as a validation effort to ensure that certain requirements were met. The following is a list of the various types of usability testing. Testing can be performed using combinations of these methods as well.

Paper mockups and prototyping. Early usability studies can be conducted using paper mockups of pages to assess user reactions to wording, layout, and sequencing. A test administrator plays the role of the computer by flipping the pages while asking a participant user to carry out typical tasks. This informal testing is inexpensive, rapid, and usually productive. Typically designers create *low-fidelity* paper prototypes of the design, but today there are computer programs (e.g., Microsoft Visio, SmartDraw, Gliffy, Balsamiq, MockingBird) that can allow designers to create more detailed *high-fidelity* prototypes with minimal effort. Interestingly enough, users have been shown to respond more openly to the lower-fidelity designs, potentially because the sketchy and less polished appearance of early prototypes clearly communicates to the user that the design can still be changed without major cost or time investment. Although prototypes are typically performed with the user and the administrator in the same physical place, with today's technologies these activities can also be done remotely. Additional information on prototyping can be found in Chapter 4.

Discount usability testing. This quick-and-dirty approach to task analysis, prototype development, and testing has been widely influential because it lowers the barriers to newcomers (Nielsen, 1993). A controversial aspect is the recommendation to use only three to six test participants. Advocates point out that most serious problems are found with only a few participants, enabling prompt revision and repeated testing, while critics hold that a broader subject pool is required to thoroughly test more complex systems. One resolution to the controversy is to use discount usability testing as a *formative evaluation* (while designs are changing substantially) and more extensive usability testing as a *summative evaluation* (near the end of the design process). The formative evaluation identifies problems that guide redesign, while the summative evaluation provides evidence for product announcements ("94% of our 120 testers completed their shopping tasks without assistance") and clarifies training needs ("with four minutes of instruction, every participant successfully programmed the device"). Small numbers may be valid for some projects, but when dealing with web companies with a large public web-facing presence, experiments may need to be run at large scale with thousands of users (see A/B testing below).

Competitive usability testing. Competitive testing compares a new interface to previous versions or to similar products from competitors. This approach is close to a controlled experimental study (Section 5.7), and staff must be careful to construct parallel sets of tasks and to counterbalance the order of presentation of the interfaces. Within-subjects designs seem the most powerful because participants can make comparisons between the competing interfaces—fewer participants are needed, although each is needed for a longer time period.

A/B testing. This method tests different designs of an interface. Typically, it is done with just two groups of users to observe and record differences between the designs. Sometimes referred to as *bucket testing*, it is similar to a *between-subjects design* (Section 5.7). This method of testing is often used with large-scale online controlled experiments (Kohavi and Longbotham, 2015). A/B testing involves randomly assigning two groups of users to either the control group (no change) or the treatment group (with the change) and then having some dependent measure that can be tested to see if there is a difference between the groups (Fig. 5.7). Before running A/B testing, it is often suggested (Crook et al., 2009) to run an *A/A test* or a null test. In A/A testing, there are still two groups, but they both receive the same treatment (the control); then the variability for power calculations and the experimentation system can be tested. In a true test, with a 95% confidence level, the null hypotheses should be rejected. This testing method has been used at Bing, where more than 200 experiments are run concurrently with 100 million customers spanning billions of changes. Some of the items tested may be new ideas and others are modifications of existing items (Kohavi et al., 2013).

Universal usability testing. This approach tests interfaces with highly diverse users, hardware, software platforms, and networks. When a wide range of international users is anticipated, such as for consumer electronics products, web-based information services, or e-government services, ambitious testing is necessary to clean up problems and thereby help ensure success. Trials with small and large displays, slow and fast networks, and a range of operating systems or Internet browsers will do much to improve the user experience. Being aware

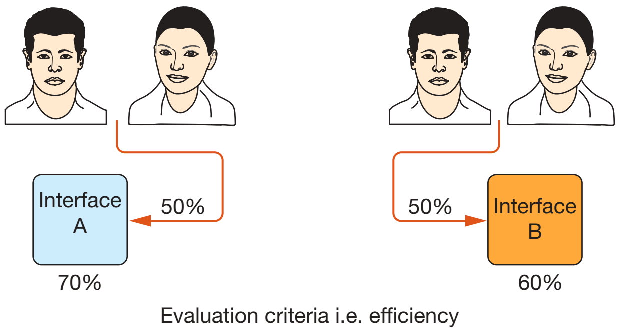


FIGURE 5.7

Example of A/B Testing

Participants are assigned randomly to one of two testing groups (A or B). The interface is similar, but a particular criterion is being evaluated and a different version of the interface is given to each group. The results are evaluated to see if there is a difference. This typically is done with relatively large numbers of participants in each group. It can be repeated many times with small variations between the interfaces each time.

of any perceptual or physical limitations of the users (e.g., vision impairments, hearing difficulties, motor or mobility impairments) and modifying the testing to accommodate these limitations will result in the creation of products that can be used by a wider variety of users (see Chapter 2).

Field tests and portable labs. This testing method puts new interfaces to work in realistic environments or in a more naturalistic environment in the field for a fixed trial period. These same tests can be repeated over longer time periods to do longitudinal testing. Field tests can be made more fruitful if logging software is used to capture error, command, and help frequencies as well as productivity measures. Portable usability laboratories with recording and logging facilities have been developed to support more thorough field testing. Today's computing devices are portable and easy to transport. If a large monitor is needed, it can often be rented at the testing location. A different kind of field testing involves supplying users with test versions of new software or consumer products; tens or even thousands of users might receive beta versions and be asked to comment. Some companies that provide this service include Noldus, UserWorks, Ovo Studios, and Experience Dynamics. Sometimes the interface requires a true immersion into the environment and an "in-the-wild" testing procedure is required (Rogers et al., 2013).

Remote usability testing. Since web-based applications are available across the world, it is tempting to conduct usability tests online, avoiding the complexity and cost of bringing participants to a lab. This makes it possible to have larger numbers of participants with more diverse backgrounds, and it may add to the realism, since participants do their tests in their own environments and use their own equipment. Participants can be recruited by e-mail from customer lists or through online communities including Amazon Mechanical Turk. This opens the pool of participants to sophisticated users who, perhaps because of their remote locations or other physical challenges, could not otherwise get to a lab location. The downside is that there is less control over user behaviors and diminished ability to observe users' reactions, although usage logs and phone interviews are useful supplements. These tests can be performed both synchronously (users do tasks at the same time while the evaluator observes) and asynchronously (users perform tasks independently and the evaluator looks at the results later). Some studies have shown remote usability testing to find more problems than traditional usability testing. Synchronous remote usability testing has been shown to be a valid evaluation technique. There are many platforms that support this type of testing. They include Citrix GoToMeeting, Cisco WebEx, IBM Sametime, join.me, and Google Hangouts. Interesting approaches include synchronous remote usability testing using virtual worlds (Madathil and Greenstein, 2011).

Can-you-break-this tests. Game designers pioneered the can-you-break-this approach to usability testing by providing energetic teenagers with the challenge of trying to beat new games. This destructive testing approach, in which the users try to find fatal flaws in the system or otherwise destroy it, has been used in other types of projects as well and should be considered seriously. Users today have little patience with flawed and poorly designed products and are often fickle with company loyalty if reasonable competitors exist.

For all its success, usability testing does have at least two serious limitations: it emphasizes first-time usage and provides limited coverage of the interface features. Since usability tests are usually only one to three hours long, it is difficult to ascertain how performance will be after a week or a month of regular usage. Within the short time of a usability test, the participants may get to use only a small fraction of the system's features, menus, dialog boxes, or help pages. These and other concerns have led design teams to supplement usability testing with varied forms of expert reviews.

Further criticisms of usability lab testing come from proponents of activity theory and those who believe that more realistic test environments are necessary to evaluate information appliances, ambient technologies, and other consumer-oriented mobile devices. Furthermore, tests of interfaces used in high-stress situations and mission-critical domains such as military combat, law enforcement, first response, and similar situations often cannot be conducted in traditional usability lab settings. Creating a realistic environment is critical to adequately test such interfaces, but it is not always possible. Designers must be aware of the total cognitive or mental load placed on the users and its implications.

Usability testing with mobile devices also needs special attention. Some issues to be aware of include availability of extra batteries and chargers, signal strength issues, network failures, ensuring that the user is focusing on the interface, and being sure users and their fingers are not blocking the observer from seeing what was tapped.

The continued interest in usability testing is apparent from the assortment of books devoted to the topic. These sources (Dumas and Loring, 2008; Rubin and Chisnell, 2008; Barnum, 2011; Nielsen and Budiu, 2012; Reiss, 2012; MacKenzie, 2013; Wilson, 2013; Preece et al., 2015) discuss setting up usability labs, the role of the usability monitor, the collection and reporting of the test data, and other information needed to run professional usability tests.

5.3.5 Usability test reports

The U.S. National Institute for Standards and Technology (NIST) took a major step toward standardizing usability test reports in 1997 when it convened a group of software manufacturers and large purchasers to work for several years to produce the Common Industry Format (CIF) for summative usability testing results. The format describes the testing environment, tasks, participants, and results in a standard way so as to enable consumers to make comparisons. The group's work (http://www.nist.gov/itl/iad/vug/) is ongoing; the participants are developing guidelines for formative usability test reports, and some best practice guidelines are emerging. Key points are that it is important to understand the audience (who will be reading the report) and to keep the report concrete and specific.

5.4 Survey Instruments

User surveys (written or online) are familiar, inexpensive, and generally acceptable companions for usability tests and expert reviews. Managers and users can easily grasp the notion of surveys, and the typically large numbers of respondents (hundreds to thousands of users) confer a sense of authority compared to the potentially biased and highly variable results from small numbers of usability-test participants or expert reviewers. The keys to successful surveys are clear goals in advance and development of focused items that help to attain those goals. Two critical aspects of survey design are validity and reliability. Experienced surveyors know that care is needed during design, administration, and data analysis (Lazar et al., 2009; Cairns and Cox, 2011; Kohavi et al., 2013; Tullis and Albert, 2013). Additional information on surveys can be found in Chapter 4.

5.4.1 Preparing and designing survey questions

A survey form should be prepared, reviewed by colleagues, and tested with a small sample of users before a large-scale survey is conducted. Methods of statistical analysis (beyond means and standard deviations) and presentation (histograms, scatterplots, and so on) should also be developed before the final survey is distributed. In short, directed activities are more successful than unplanned statistics-gathering expeditions. Our experience is that directed activities also provide the most fertile frameworks for unanticipated discoveries. Since biased samples of respondents can produce erroneous results, survey planners need to build in methods to verify that respondents represent the population in terms of age, gender, experience, and other relevant characteristics.

It is important to pre-test or pilot-test any survey instrument prior to actual use. Users can be asked for their subjective impressions about specific aspects of the interface, such as the representation of:

- Task domain objects and actions
- Interface domain metaphors and action handles
- Syntax of inputs and design of screen displays

It may also be useful to ascertain certain characteristics about the users, including:

- Background *demographics* (age, gender, origins, native language, education, income)
- Experience with computers (specific applications or software packages, length of time, depth of knowledge, whether knowledge was acquired through formal training or self-teaching)
- Job responsibilities (decision-making influence, managerial roles, motivation)
- Personality style (introvert versus extrovert, risk taking versus risk averse, early versus late adopter, systematic versus opportunistic)
- Reasons for not using an interface (inadequate services, too complex, too slow, afraid)
- Familiarity with features (printing, macros, shortcuts, tutorials)
- Feelings after using an interface (confused versus clear, frustrated versus in control, bored versus excited)

Online and web-based surveys avoid the cost and effort of printing, distributing, and collecting paper forms. Many people prefer to answer a brief survey on a computer or other electronic device instead of filling in and returning a printed form, although there is a potential bias in the self-selected sample. Some surveys can have very large numbers of respondents. Some companies that provide computerized surveys include Survey Monkey, Survey Gizmo, Qualtrics, and Question Pro. Academic or educational discounts may be available.

Typically, participants are asked to respond to a series of statements according to the following commonly used *Likert* scale:

Strongly agree Agree Neutral Disagree Strongly disagree

The items in the survey could be similar to the following:

- I can effectively perform the task using this interface
- Items are placed where I expected to find them in the interface

Such a list of statements can help designers to identify problems users are having and to demonstrate improvements to the interface as changes are made; progress is demonstrated by improved scores on subsequent surveys.

Another approach is to use a set of bipolar semantically anchored items (pleasing versus irritating, simple versus complicated, concise versus redundant) that ask users to describe their reactions to using the interface. Users have to rate the items on 1-to-7 scales:

Hostile	1	2	3	4	5	6	7	Friendly
Easy to use	1	2	3	4	5	6	7	Difficult to use
Clear	1	2	3	4	5	6	7	Confusing

Yet another approach is to ask users to evaluate various aspects of the interface design, such as the readability of characters, use of the terminology, organization of the structure, or the meaning of the icons/controls. If users rate as poor one aspect of the system, the designers have a clear indication of what needs to be redone. If precise—as opposed to general—questions are used in surveys, there is a greater chance that the results will provide useful guidance for taking action.

Additional attention may be needed when dealing with special populations (see Chapter 2). For example, questionnaires for children must be in age-appropriate language, questionnaires for international users may need to be translated, larger fonts may be needed for older adults, and special accommodations may need to be made for users with disabilities.

5.4.2 Sample questionnaires

Questionnaires and surveys are commonly used in usability evaluation. Several instruments and scales have been developed and refined over time. The early questionnaires concentrated on elements such as clarity of fonts, appearance on the display, and keyboard configurations. Later questionnaires dealt with multimedia components, conferencing, and other current interface designs including consumer electronics and mobile devices. Here is some information on a few (most use a Likert-like scale):

The *Questionnaire for User Interaction Satisfaction* (QUIS). The QUIS (http://lap.umd.edu/quis/) has been applied in many projects with thousands of users, and new versions have been created that include items relating to website design. The University of Maryland's Office of Technology Commercialization licenses the QUIS. Special licensing terms may be available for students. Table 5.1 contains a portion of the QUIS, including an example for collecting computer experience data.

The *System Usability Scale* (SUS). Developed by John Brooke, it is sometimes referred to as the "quick and dirty" scale. The SUS consists of 10 statements

Examples of the specific satisfaction scale questions:					
5.4	Messages which appear on display:	confusing	clear	1 2 3 4 5 6 7 8 9	NA
5.4.1	Instructions for commands or choice:	confusing	clear	1 2 3 4 5 6 7 8 9	NA

TABLE 5.1

Questionnaire for User Interaction Satisfaction (QUIS)
(© University of Maryland, 1997).

		Strongly disagree				**Strongly agree**
1	I think that I would like to use this system frequently	1	2	3	4	5
2	I found the system unnecessarily complex	1	2	3	4	5

TABLE 5.2
System Usability Scale (SUS) example (Brooke, 1996).

with which users rate their agreement (on a 5-point scale). Half of the questions are positively worded, and the other half are negatively worded. A score is computed that can be viewed as a percentage. Table 5.2 contains a sample from the SUS.

The *Computer System Usability Questionnaire* (CSUQ). A later development by IBM (based on the earlier PSSUQ) contains 19 statements to which participants respond using a 7-point scale. Table 5.3 contains a sample from the CSUQ.

The *Software Usability Measurement Inventory* (SUMI). Developed by the Human Factors Research Group (HFRG), it contains 50 items designed to measure users' perceptions of their affect (emotional response), efficiency, and control and of the learnability and helpfulness of the interface (Kirakowski and Corbett, 1993). Table 5.4 contains a sample from the SUMI.

The *Website Analysis and MeasureMent Inventory* (WAMMI) questionnaire. It was designed for web-based evaluations and is available in more than a dozen languages (http://www.wammi.com/).

Although many of these questionnaires were developed a while ago, they still serve as reliable and valid instruments. Some have been transformed by changing the focus of the items asked about. Specialized questionnaires have been developed and tested based on these proven instruments. One example is the Mobile Phone Usability Questionnaire (MPUQ), which consists of 72 items

			1 2 3 4 5 6 7		**NA**
1	Overall, I am satisfied with how easy it is to use this system.	Strongly disagree	• • • • • • •	Strongly agree	•
2	I can effectively complete my work using this system.	Strongly disagree	• • • • • • •	Strongly agree	•

TABLE 5.3
Computer System Usability Questionnaire (CSUQ) example.

		Agree	Undecided	Disagree
1	This software responds too slowly to inputs.	❏	❏	❏
2	I would recommend this software to my colleagues.	❏	❏	❏

TABLE 5.4

Software Usability Measurement Inventory (SUMI) example.

broken down into six factors: ease of learning and use, helpfulness and problem-solving capabilities, affective aspect and multimedia properties, commands and minimal memory load, control and efficiency, and typical tasks for mobile phones (Ryu, 2009). An example of sample questions from the MPUQ can be found in Table 5.5. The SUS has also been used with cellphones as well as interactive voice systems, web-based interfaces, and other interfaces and continues to be a robust and versatile tool. UMUX-LITE is another option as a shortened SUS (Lewis et al., 2013). As with any metric, any score should not be used in isolation. The best testing procedure, leading to the most confidence-inspiring results, would include triangulating the data from multiple methods, such as observations, interviews, logging of interface usage, and qualitative satisfaction data as well.

Writing and designing good questionnaires is an art as well as a science. Several books (Rubin and Chisnell, 2008; Sauro and Lewis, 2012; Tullis and Albert, 2013) and articles exist that provide further reading on use, validity, and development of good and valid questionnaires. In addition to the standard-type measures of satisfaction, specialized devices (e.g., mobile devices) and gaming interfaces may require unique measures such as pleasure, joy, affect, challenge, or realism. Some links to some older questionnaires include the following: Gary Perlman (http://garyperlman.com/quest/quest.cgi?form=USE) and Jurek Kirakowski (http://www.ucc.ie/hfrg/resources/qfaq1.html).

Examples of sample questions relating to mobile phones:
Is it easy to change the ringer signal?
Can you personalize the ringer signal with this product? If so, is that feature useful and enjoyable for you?
Do you feel excited when using this product?
Is it easy to use the phone book feature of this product?

TABLE 5.5

Mobile Phone Usability Questionnaire (MPUQ) example.

5.5 Acceptance Tests

For large implementation projects, the customer or manager usually sets objective and measurable goals for hardware and software performance. Many authors of requirements documents are even so bold as to specify the mean time between failures as well as the mean time to repair for hardware and, in some cases, software failures. More typically, a set of test cases is specified for the software, with possible response-time requirements for the hardware/software combination (see Chapter 12). If the completed product fails to meet these acceptance criteria, the system must be reworked until success is demonstrated.

These notions can be neatly extended to the human interface. Explicit acceptance criteria should be established when the requirements document is written or when a contract is offered. Rather than use the vague and misleading criterion of "user friendliness," measurable criteria for the user interface can be established for the following:

- Time for users to learn specific functions
- Speed of task performance
- Rate of errors by users
- User retention of commands over time
- Subjective user satisfaction

An acceptance test for a food-shopping website might specify the following:

> The participants will be 35 adults (25–45 years old), native speakers with no disabilities, hired from an employment agency. They will have moderate web-use experience: one to five hours/week for at least a year. They will be given a five-minute demonstration on the basic features. At least 30 of the 35 adults should be able to complete the benchmark tasks within 30 minutes.

Another testable requirement for the same interface might be this:

> Special participants in three categories will also be tested: (a) 10 older adults aged 55–65; (b) 10 adult users with varying motor, visual, and auditory disabilities; and (c) 10 adult users who are recent immigrants and use English as a second language.

Since the choice of the benchmark tasks is critical, pilot testing must be done to refine the materials and procedures used. A third item in the acceptance test plan might focus on retention:

> Ten participants will be recalled after one week and asked to carry out a new set of benchmark tasks. In 20 minutes, at least eight of the participants should be able to complete the tasks correctly.

In a large interface, there may be 8 or 10 such tests to carry out different components of the interface and with different user communities. Other criteria, such as subjective satisfaction, output comprehensibility, system response time, installation procedures, documentation, or graphics appeal, may also be considered in acceptance tests of complete commercial products.

If precise acceptance criteria are established, both the customer and the interaction designer can benefit. Arguments about user friendliness are avoided, and contractual fulfillment can be demonstrated objectively. Acceptance tests differ from usability tests in that the atmosphere may be adversarial, so outside testing organizations are often appropriate to ensure neutrality. The central goal of acceptance testing is not to detect flaws, but rather to verify adherence to requirements.

After successful acceptance testing, there may be a period of field testing or an extensive beta test with real users before national or international distribution. In addition to further refining the user interface, field tests can improve training methods, tutorial materials, telephone-help procedures, marketing methods, and publicity strategies.

The goal of early expert reviews, usability testing, surveys, acceptance testing, and field testing is to force as much as possible of the evolutionary development into the pre-release phase, when change is relatively easy and less expensive to accomplish.

5.6 Evaluation during Active Use and Beyond

A carefully designed and thoroughly tested interface is a wonderful asset, but successful active use requires constant attention from dedicated managers, user-service personnel, and maintenance staff. Everyone involved in supporting the user experience can contribute to interface refinements that provide ever-higher levels of service. You cannot please all of the users all of the time, but earnest effort will be rewarded by the appreciation of a grateful user community. Perfection is not attainable, but incremental improvements are possible and are worth pursuing.

Gradual interface dissemination is useful so that problems can be repaired with minimal disruption. As user numbers grow, major changes to the interface should be limited to an announced revision. If interface users can anticipate the changes, resistance will be reduced, especially if they have positive expectations of improvement. More frequent changes are expected in the rapidly developing web and interactive environments, but stable access to key resources even as novel services are added and sincere interest in the user experience may be the winning policy.

5.6.1 Interviews and focus-group discussions

Interviews with individual users can be productive because the interviewer can pursue specific issues of concern to help in better understanding the user's perspective. Interviewing can be costly and time-consuming, so usually only a small fraction of the user community is involved. On the other hand, direct contact with users often leads to specific, constructive suggestions. Professionally led focus groups can elicit surprising patterns of usage or hidden problems, which can be quickly explored and confirmed by participants. On the other hand, outspoken individuals can sway the group or dismiss comments from weaker participants. Interviews and focus groups can be arranged to target specific sets of users, such as experienced or long-term users of a product, generating different sets of issues than would be raised with novice users.

5.6.2 Continuous user-performance data logging

The software architecture should make it easy for system managers to collect data about the patterns of interface usage, speed of user performance, rate of errors, and/or frequency of requests for online assistance. Logging data provide guidance in the acquisition of new hardware, changes in operating procedures, improvements to training, plans for system expansion, and so on.

For example, if the frequency of each error message is recorded, the highest-frequency error is a candidate for attention. The message could be rewritten, supporting materials could be revised, the software could be changed to provide more specific information, or the syntax could be simplified. Without specific logging data, however, the system-maintenance staff has no way of knowing which of the many hundreds of error-message situations presents the biggest problem for users. Similarly, staff should examine messages that never appear to see whether there is an error in the code or whether users are avoiding use of some facility.

If logging data are readily available, changes to the human-computer interface can be made to simplify access to frequently used features. Managers also should examine unused or rarely used functionality to understand why users are avoiding those features. A major benefit of usage-frequency data is the guidance that they provide to system maintainers in optimizing performance and in reducing costs for all participants. This latter argument may yield the clearest advantage to cost-conscious managers, whereas the increased quality of the interface is an attraction to service-oriented managers. Zooming in on specific events (e.g., undo and erase) provides a cost-effective, automated approach to facilitate the detection of critical incidents that may not be discovered by self-reporting (Akers et al., 2009).

Logging may be well-intentioned, but users' rights to privacy deserve to be protected. Links to specific user names should not be collected unless necessary.

When logging aggregate performance crosses over to monitoring individual activity, managers must inform users of what is being monitored and how the gathered information will be used. Although organizations may have a right to measure workers' performance levels, workers should be able to view the results and to discuss the implications. If monitoring is surreptitious and is later discovered, resulting worker mistrust of management could be more damaging than the benefits of the collected data. Manager and worker cooperation to improve productivity and worker participation in the process and benefits are advised.

With the huge impact of the internet on e-commerce, many companies are interested in tracking hits on their sites, page views, and so on. There has been an explosion of companies (Google, Microsoft, Yahoo!, and others) that offer such services, referred to as *web analytics* (sometimes called *big data*). This data gathering can provide companies with detailed tracking information on their websites, including graphic displays, dashboards, and calculations and computations to demonstrate the impact on return on investment changes and other modifications. This information can be presented in graphical dashboards to provide visualizations of the data (see Chapter 16).

With the interest in big data, many services are making a success of providing clients with log data and analyses of web visits from their panels of users. These users have provided their demographic information and are paid to answer surveys or allow their web-visitation patterns to be logged. The purchasers of the data are interested in knowing what kinds of people buy books, visit news sites, or seek healthcare information to guide their marketing, product development, and website design efforts. Some of these services include Alexa, Quora, Pew Internet, Hitwise, Google Analytics, Forrester, comScore, and Nielsen Digital Ad Ratings.

5.6.3 Online or chat consultants, e-mail, and online suggestion boxes

Online or chat consultants can provide extremely effective and personal assistance to users who are experiencing difficulties. Many users feel reassured if they know that there is a human being to whom they can turn when problems arise. These consultants are an excellent source of information about problems users are having and can suggest improvements and potential extensions.

Some organizations offer toll-free numbers through which the users can reach a knowledgeable consultant; others charge for consultation by the minute or offer support only to elite or premium customers. On some systems, the consultants can monitor or even control the user's computer and see the same display that the user sees while maintaining telephone or other chat contact (Fig. 5.8). This service can be extremely reassuring because users know that someone can walk them through the correct sequence to complete their tasks. When users want service, they typically want it immediately, and users often work worldwide

FIGURE 5.8

Online chat consultant. Typically, the consultants are on a headset and may or may not be able to view the participants. They will communicate by a vocal or chat dialogue. If it is a chat dialogue, there is usually some indication for the participant to wait while the consultant is typing.

in a 24-hour, 7-day-a-week environment. Many organizations are using software agents with *recommender systems* to provide real-time chat facilities, thereby integrating the human touch with common automated responses. Such services help users, build customer loyalty, and provide insights that can lead to design refinements as well as novel product extensions. Although these services are often well-received, companies need to be aware of biases. Those participants who respond to services offered online may not be representative of the general user population, so drawing conclusions about the data collected in these types of interventions can be problematic (Crook et al., 2009).

5.6.4 Discussion groups, wikis, newsgroups, and search

Some users may have questions about the suitability of a software package for their application or may be seeking someone who has had experience using an interface feature. They are not likely to have a particular individual in mind, so e-mail does not serve their needs. Furthermore, with the international use of software products and the 24-hour, 7-day-a-week, always-on computing environment, users may encounter issues outside of traditional working hours. Many interaction designers and website managers offer users discussion groups, newsgroups, or wikis to permit posting of open messages and questions. More independent discussion groups are also hosted by various services and can easily be found using today's powerful search engines.

Discussion groups usually offer lists of item headlines, allowing users to scan for relevant topics. User-generated content fuels these discussion groups. Most anyone can add new items, but usually someone moderates the discussion to ensure that offensive, useless, outdated, or repetitious items are removed. When there are a substantial number of users who are geographically dispersed, moderators may have to work hard to create a sense of community.

With the prevalence of the internet, searching has become even more generic and ubiquitous and typically under user control. Users often input into Google (or another search engine) a phrase or a set of words describing their issues, easily yielding a long list of matches. Some may match exactly what the user is looking for without any registrations or other sign-up activities. These matches may point to wikis, discussion forums, company FAQs, and even YouTube videos.

5.6.5 Tools for automated evaluation

Software tools can be effective in evaluating user interfaces for applications, websites, and mobile devices. Even straightforward tools to check spelling or concordance of terms benefit interaction designers. Simple metrics that report numbers of pages, widgets, or links between pages capture the size of a user-interface project, but the inclusion of more sophisticated evaluation procedures can allow interaction designers to assess whether a menu tree is too deep or contains redundancies, whether labels have been used consistently, whether all buttons have proper transitions associated with them, and so on.

Research has noted some recommendations. Keep average link text to two to three words, using *sans serif* fonts and applying colors to highlight headings. One intriguing finding was that preferred websites do not always have the fastest user performance, suggesting that in e-commerce, mobile, entertainment, and gaming applications, the attractiveness may be more important than rapid task execution. Further analysis of the results could lead to conjectures about the design goals that bring about high preference. For example, users may prefer designs that are comprehensible, predictable, and visually appealing and that incorporate relevant content. Today, sophisticated users exist with high expectations. Young folks (such as *digital natives*) have grown up with computers and other mobile devices, and these devices are an integral part of their life and related activities.

In the recent past, download speeds for webpages were an issue, and people used website optimization services that could count the number of items in a page, the number of bytes in each image, and the size of the source code. These services also provided suggestions for how to revise webpages for faster performance. Today the issue is more about the number of hits and the visibility of a webpage.

Another family of tools is *run-time logging software*, which captures the users' patterns of activity. Simple reports—such as the frequency of each error message, menu-item selection, dialog-box appearance, help invocation, form-field usage, or webpage access—are of great benefit to maintenance personnel and to revisers of the initial design. Experimental researchers can also capture performance data for alternative designs to guide their decision making. Software to analyze and summarize the performance data (e.g., TechSmith's Morae) is improving steadily (see Fig. 5.9).

When evaluating mobile devices in the field, unobtrusive methods to gather data may be needed. A log-file-recording tool that captures clicks with associated timestamps and positions on the display, keeping track of items selected and display changes, capturing the page shots, and recording when a user is finished, can provide valuable information for analysis. Another approach to gather user feedback is to do a *site intercept survey*. This involves putting a small JavaScript snippet on a webpage allowing it to gather information from the users.

FIGURE 5.9

This is an example of some of the automated reports that can be created with software such as TechSmith's Morae. The item being measured is mouse clicks. This shows the view for task 2 (selected in the tabbed bar). Obviously, the other three tasks could also be displayed. These are the values for participant 4. The dropdown list box would allow the evaluator to choose the mouse clicks for other participants. Across the horizontal axis, time is shown.

Of course, gathering the data from usability evaluations is only the beginning. Making sense of the data, identifying patterns, and reaching a better understanding of what the data show are difficult and tedious tasks.

5.7 Controlled Psychologically Oriented Experiments

Scientific and engineering progress is often stimulated by improved techniques for precise measurement. Rapid progress in the design of interfaces will be stimulated as researchers and practitioners continue to evolve suitable human-performance measures and techniques. We have come to expect that automobiles will have gas mileage reports pasted to their windows, appliances will have energy-efficiency ratings, and textbooks will be given grade-level designations; soon, we will expect software packages to show learning-time estimates and user-satisfaction indices from appropriate evaluation sources.

5.7.1 Experimental approaches

Academic and industrial researchers understand that the power of the traditional scientific method can be fruitfully employed in the study of interfaces. They are conducting numerous experiments that aid in better understanding of basic design principles. The classic scientific method for interface research (as stated in Chapter 1), which is based on controlled experimentation, has this basic outline:

- Understanding of a practical problem and related theory
- Lucid statement of a testable hypothesis
- Manipulation of a small number of independent variables
- Measurement of specific dependent variables
- Careful selection and assignment of subjects
- Control for bias in subjects, procedures, and materials
- Application of statistical tests
- Interpretation of results, refinement of theory, and guidance for experimenters

The classic experimental methods of psychology are being enhanced to deal with the complex cognitive tasks of human performance with information and computer systems. The transformation from Aristotelian introspection to Galilean experimentation that took two millennia in physics is being accomplished in just over three decades in the study of human-computer interaction.

The scientific approach required for controlled experimentation yields narrow but reliable results. Through multiple replications with similar tasks, participants, and experimental conditions, reliability and validity can be enhanced. Each small experimental result acts like a tile in the mosaic of human performance with computer-based information systems.

Managers of actively used systems are also coming to recognize the power of controlled experiments in fine-tuning the human-computer interface. As proposals are made for new interfaces, novel devices, and reorganized display formats, a carefully controlled experiment can provide data to support a management decision. Fractions of the user population can be given proposed improvements for a limited time, and then performance can be compared with the control group. Dependent measures may include performance times, user-subjective satisfaction, error rates, and user retention over time.

For example, the competition over mobile device–input methods has led to numerous experimental studies of keyboard arrangements with similar training methods, standard benchmark tasks, common dependent measures that account for error rates, and strategies for testing frequent users. Such careful controls are necessary because a 10-minute reduction in learning time, a 10% speed increase, or 10 fewer errors could be a vital advantage in a competitive consumer market.

Similar controlled studies are being used in online experiments at a large scale with large web-based groups (Kohavi and Longbotham, 2015). It is important to pay attention to the size and the representativeness of the group used. One needs to be aware of novelty and primacy effects that can affect the results. Other "rules of thumb" in doing these types of studies are discussed by Kohavi et al. (2014).

5.7.2 Experimental design

A full discussion of experimental design is outside the scope of this book, although many excellent resources exist (Lazar et al., 2009; Cairns and Cox, 2011; Sauro and Lewis, 2012; MacKenzie, 2013; Tullis and Albert, 2013). Experimental design and statistical analysis are complex topics. Some basic terminology and methodologies are described, though novice experimenters would be well advised to collaborate with experienced research scientists and statisticians to develop the details properly.

In a tightly controlled experimental study, selecting the appropriate participants is important. Since conclusions and inferences are often made from the data, it is important that the sample is *representative* of the target users for the interface. Users are frequently grouped or categorized by some sort of *demographic*, such as age, gender, computer experience, or other attribute. When selecting participants from a population to create the sample, the *sampling technique* needs to be considered. Are people selected *randomly*? Is there a *stratified* subsample that should be used? Novice researchers may want to use their friends and family members, creating a *convenience* sample, but such a sample is not typically representative, may be biased, and therefore can compromise the confidence and validity of the results. The sample size is another consideration. It is important to define a *confidence level* that needs to be met for the study. A full discussion of sample sizes and confidence levels can be found in most statistics books.

Basic experimental design comes in two forms: *between-subjects* or *within-subjects*. In a *between-subjects* design, the groups are relatively similar in makeup, and each group is given a different treatment. To have a powerful effect, this design approach needs to have a relatively large number of users in each group. The large sample size usually ensures that the groups (if selected appropriately) are similar in nature, so the differences can be attributed primarily to the different treatments. If the groups are too small, the results may be related to the individual characteristics of each group. In a *within-subjects* design, each participant performs the same tasks, and the data being recorded are compared across participants. Although the sample size can be smaller, there may still be concerns about fatigue (causing performance to decrease) or practice and familiarity (causing performance to increase). It is important to *counterbalance* the tasks, since the order of the tasks can affect the results. If the variable being measured is ease of use, earlier tasks may artificially seem more difficult because the user is not yet familiar with the system; likewise, later tasks may be seen as easier, not because the tasks themselves are less complex but because of the familiarity with the system that the user has acquired.

In the design of an experimental study, different types of variables need to be considered and understood. The *independent variable* is something that is being manipulated. For example, you may have two different interface designs: one that provides access to a help system and one that does not. The *dependent variable* is something that happens as a result of the experiment and is usually measured. Examples of dependent variables include time to complete the task, number of errors, and user satisfaction. Experimental design needs to be carefully controlled so the main differences found in the dependent variables can be attributed to the independent variables, not other outside sources or confounding variables. To help control for potential systematic bias and experimental error in the study design, the researcher should apply randomization strategies, when plausible, such as random selection of participants and random assignment of participants to testing conditions. See Box 5.4 for a discussion on Simpson's Paradox. One also needs to be aware of *false positives*. These are positive results that are not really true. They could be due to experimental design issues, biased selection, data issues, or just chance.

BOX 5.4

Simpson's Paradox (Crook et al., 2009).

A study is being done with two groups (an A/B test). Users can use different browsers. The sampling is not uniform; there may be users from some browsers sampled at a higher rate. When the study is complete, the treatment group does better. Upon further analysis of the data, separating the users by browser types, the treatment is actually worse for all browser types.

With the advent of the maturing of HCI and more emphasis on the user experience, measurements are changing. Traditional quantitative approaches are still important and valid, but attention needs to be given to qualitative measures (Bazeley, 2013) and associated methodologies as well. Measuring emotional dimensions associated with the interface will provide a more holistic perspective of the user experience. Differences between users' perceptions and objective measures need to continue to be identified. Researchers need more work on validating scales and instruments for these types of measures.

Practitioner's Summary

Interface developers evaluate their designs by conducting expert reviews, usability tests (in lab settings, in the field, and "in-the-wild"), surveys, and rigorous acceptance tests. Once interfaces are released, developers carry out continuous performance evaluations by interviews or surveys or by logging users' performance in a way that respects their privacy. If you are not measuring user performance, you are not focusing on user experience and usability!

Successful interface project managers understand that they must work hard to establish a relationship of trust with the user community. As markets are opened (for example, in another country or vertical market segment), managers have to start fresh in gaining recognition and customer loyalty. Special attention may need to be devoted to novice users, users with disabilities, and other special populations (children, older adults). In addition to providing a properly functioning system, successful managers recognize the need to offer mechanisms for feedback, such as online surveys, interviews, discussion groups, consultants, suggestion boxes, newsletters, and conferences as well as participation in the common social media outlets.

Ideally, a company has a group of personnel targeted and trained in doing usability evaluations. But sometimes one person may be the only usability evangelist in the company. That position requires wearing many hats to be sure the usability needs of the company are addressed. Research has shown that including usability early in the product design cycle provides a much better return on investment (ROI) compared to tacking on usability at the end.

Researcher's Agenda

Researchers can contribute their experience with experimentation to develop improved techniques for interface evaluation and the user experience. Guidance in conducting pilot studies, acceptance tests, surveys, interviews, and discussions would benefit large-scale development groups, but additional attention needs to be given to smaller projects and incremental-type changes. Strategies are needed to cope with evaluation for the numerous specific populations of users and the diverse forms of disabilities that users may have. Experts in constructing psychological tests can help in preparing validated and reliable test instruments for subjective evaluation of the varying types of interfaces, from small mobile devices to very large displays, including specialized interfaces such as gaming. Such standardized tests would allow independent groups to compare the acceptability of interfaces. Would benchmark datasets and task libraries help standardize evaluation? How useful can researchers make automated testing against requirements documents? How many users are needed to generate valid recommendations? How can we better explain the differences between users' perceptions of a task and the objective measures? How do we select the best measure for a task? How can life-critical applications for experienced professionals be tested reliably? Is there a single usability metric that can be used and compared across types of interfaces? Can we combine performance data and subjective data and create a single meaningful result? Is there a scorecard that can be used to aid in the interpretation of usability results? Is there a theory to explain and understand the relationship between measures? Also, how do we best incorporate and evaluate qualitative data and dimensions such as fun, pleasure, joy, affect, challenge, or realism?

Input from experimental, cognitive, and clinical psychologists would help computer specialists to recognize the importance of the human aspects of computer use. Can psychological principles be applied to reduce novice users' anxiety or expert users' frustration? Could profiles of users' skill levels with interfaces be helpful in job-placement and training programs? How can good usability practices be applied to the gaming environment while preserving the challenge and excitement of the game itself? Continuously keeping the user experience in mind is an integral part of this.

Additional work is also needed on the appropriate choice of evaluation methodology. Some of the traditional methodologies need to be expanded, and non-empirical methods, such as sketches and other design alternatives, should be considered. As HCI is maturing as a discipline, two facets of HCI are emerging. One approach is micro-HCI: counting discrete items (e.g., mouse clicks) and other quantitative items. These are places where measurable performance in terms of speed and errors can be reported using controlled experiments. The second approach is macro-HCI: dealing more with the full user experience including social engagement (see Chapter 3). Changes are needed to make usability reports that are understandable, readable, and useful. Additional work on developing automated tools is needed, with attention paid to the specialized systems (mobile devices, games, personal devices) that are readily available today. The standardized usability instruments need modification and validation as they deal with different criteria and different environments. What happens if testing cannot take place in a usability lab? Perhaps the testing must be done in a field setting or "in-the-wild" to ensure validity. How can we effectively simulate the high-stress situations that users encounter in hostile environments? Satisfaction may be more broadly defined to include characteristics such as fun, pleasure, and challenge.

WORLD WIDE WEB RESOURCES

www.pearsonglobaleditions.com/shneiderman

- Additional information on usability testing and questionnaires is available on the companion website.
- Resource on usability methods and guidelines from the U.S. government: http://www.usability.gov
- Usability Methods Toolbox from James Hom (older but good information): http://usability.jameshom.com
- SUMI Questionnaire from J. Kirakowski: http://www.ucc.ie.hfrg.questionnaires/sum/index.html
- Testing Methods and Tools Guide to Usability and Software Engineering (GUSE) from University of Maryland: http://lte-projects.umd.edu/guse/testing.html
- Heuristic Evaluations from Jakob Nielsen: http://www.nngroup.com/topic/heuristic-evaluation/
- Usability First—Foraker Design: http://www.usabilityfirst.com
- Zazelenchuk's Usability Test DataLogger: http://www.userfocus.co.uk/resources/datalogger.html
- Usability information from the University of Texas: http://www.utexas.edu/learn/usability/
- A comprehensive list of UX evaluation methods from All About UX created and maintained by volunteers: http://www.allaboutux.org/all-methods
- Sample size calculator: http://www.blinkux.com/usability-sample-size
- How to Conduct Eyetracking Studies, Kara Pernice and Jakob Nielsen: http://www.nngroup.com/reports/how-to-conduct-eyetracking-studies/
- *New York Times* article on A/B testing: http://www.nytimes.com/2015/09/27/upshot/a-better-government-one-tweak-at-a-time.html?_r=0

Discussion Questions

1. Describe at least three different types of expert review methods.
2. Create a bird's-eye view of an interface you wish to investigate. Focus on detecting inconsistencies and spotting unusual patterns.
3. Compare and contrast controlled psychological experiments and usability tests in the evaluation process of user interfaces. Be sure to include the benefits and limitations of each.
4. List the advantages and disadvantages of survey questionnaires.

For questions 5–7, refer to the following instructions:
One argument against the current interface design of a popular word processor is that it has all the functional menu items appearing together, which cause the interface to be too complex. This complexity results in a confusing and frustrating experience for novice users. An alternative design is to provide different levels of functional complexity, so users can choose the level that is suitable for them, then advance to higher level as they get familiar with the tool, thus feel more comfortable and learn more efficiently. You are asked to conduct usability testing to compare these two designs.

5. Which type of usability testing should be used for this situation? Explain why.
6. List and briefly describe the steps in the usability testing you would conduct.
7. Do you think there should be a control group in your test? Justify your answer.

References

Akers, David, Simpson, Matthew, Jeffries, Robin, and Winograd, Terry, Undo and erase events as indicators of usability problems, *Proceedings of the ACM Conference on Human Factors in Computing Systems*, ACM Press, NY (2009), 659–668.

Babaian, T., Lucas, W., and Oja, M-K., Evaluating the collaborative critique method, *Proceedings of the ACM Conference on Human Factors in Computing Systems*, ACM Press, NY (2012).

Barcelos, T., Muñoz, R., and Chalegre, V., Gamers as usability evaluators: A study in the domain of virtual worlds, *Proceedings of the 11th Brazilian Symposium on Human Factors in Computing Systems*, (2012), 301–304.

Bargas-Avila, J., and Hornbæk, K., Old wine in new bottles or novel challenges? A critical analysis of empirical studies of user experience, *Proceedings of the ACM Conference on Human Factors in Computing Systems*, ACM Press, NY (2011), 2689–2698.

Barnum, Carol M., *Usability Testing Essentials*, Morgan Kaufmann (2011).

Bazeley, Patricia, *Qualitative Data Analysis: Practical Strategies*, Sage Publications (2013).

Brooke, John, SUS: A quick and dirty usability scale, in Jordan, P. W., Thomas, B., Weerdmeester, B. A., and McClelland, I. L. (Editors), *Usability Evaluation in Industry*, Taylor and Francis, London, UK (1996).

Cairns, P., and Cox, A. L. (Editors), *Research Methods for Human-Computer Interaction*, Cambridge University Press (2008, reprinted 2011).

Crook, Thomas, Frasca, Brian, Kohavi, Ron, and Longbotham, Roger, Seven pitfalls to avoid when running controlled experiments on the web, *KDD '09*, Paris, France (2009), 1105–1113.

Dimond, J., Fiesler, C., DiSalvo, B., Pelc, J., and Bruckman, A., Qualitative data collection technologies: A comparison of instant messaging, email, and phone, *ACM GROUP'12* (2012), 277–280.

Dumas, Joseph, and Loring, Beth, *Moderating Usability Tests: Principles and Practices for Interacting*, Morgan Kaufmann, Burlington, MA (2008).

Elmqvist, Niklas, and Yi, Ji Soo, Patterns for visualization evaluation, *Proceedings 2012 BELIV Workshop: Beyond Time and Errors: Novel Evaluation Methods for Visualization* (2012).

Greenberg, S., Carpendale, S., Marquardt, N., and Buxton, B., *Sketching User Experiences: The Workbook*, Morgan Kaufmann (2012).

Hartson, R., and Pyla, P., *The UX Book: Process and Guidelines for Ensuring a Quality User Experience*, Morgan Kaufmann (2012).

Joyce, G., Lilley, M., Barker, T., and Jefferies, A., Adapting heuristics for the mobile panorama, *Interaction '14* (2014).

Kirakowski, J., and Corbett, M., SUMI: The Software Usability Measurement Inventory, *British Journal of Educational Technology 24,* 3 (1993), 210–212.

Kohavi, Ron, and Longbotham, Roger, Online controlled experiments and A/B tests to appear in Sammut, Claude, and Webb, Geoff (Editors), *The Encyclopedia of Machine Learning and Data Mining* (2015).

Kohavi, Ron, Deng, Alex, Frasca, Brian, Walker, Toby, Xu, Ya, and Pohlmann, Nils, Online controlled experiments at large scale, *KDD'13*, Chicago, IL (2013), 1168–1176.

Kohavi, Ron, Deng, Alex, Longbotham, Roger, and Xu, Ya, Seven rules of thumb for web site experimenters, *KDD'14* New York, NY (2014), 1857–1866.

Korhonen, Hannu, and Koivisto, Elina M. I., Playability heuristics for mobile games, *Proceedings MobileHCI '06 Conference*, ACM Press, New York (2006), 9–15.

Lazar, J., Feng, J. H., and Hochheiser, H., *Research Methods in Human-Computer Interaction*, Wiley (2009).

Lewis, James R., Usability: Lessons learned . . . and yet to be learned, *International Journal of Human-Computer Interaction 30* (2014), 663–684.

Lewis, J., Utesch, B., and Maher, D., UMUX-LITE: When there's no time for the SUS, *Proceedings of the ACM Conference on Human Factors in Computing Systems*, ACM Press, NY (2013), 2099–2102.

Lund, A., *User Experience Management: Essential Skills for Leading Effective UX Teams*, Morgan Kaufmann (2011).

MacDonald, Craig M., and Atwood, Michael, What does it mean for a system to be useful? An exploratory study of usefulness, *DIS 2014*, Vancouver, BC, Canada (2014), 885–894.

MacKenzie, I. Scott, *Human-Computer Interaction: An Empirical Research Perspective,* Morgan Kaufmann, San Francisco, CA (2013).

Madathil, K., and Greenstein, J., Synchronous remote usability testing: A new approach facilitated by virtual worlds, *Proceedings of the ACM Conference on Human Factors in Computing Systems*, ACM Press, New York (2011), 2225–2234.

Masip, L., Granollers, T., and Oliva, M., A heuristic evaluation experiment to validate the new set of usability heuristics, *2011 Eighth International Conference on Information Technology: New Generations* (2011).

Molich, Rolf, Jeffries, Robin, and Dumas, Joseph S., Making usability recommendations useful and usable, *Journal of Usability Studies 2,* 4 (2007), 162–179.

Nielsen, Jakob, *Usability Engineering*, Academic Press, New York (1993).

Nielsen, J., Heuristic evaluation, in Nielsen, J., and Mack, R. L. (Editors), *Usability Inspection Methods*, John Wiley & Sons, New York, NY (1994).

Nielsen, J., and Budiu, R., *Mobile Usability*, New Riders (2012).

Pinelle, David, Wong, Nelson, and Stach, Tadeusz, Heuristic evaluation for games: Usability principles for video game design, *Proceedings of the ACM Conference on Human Factors in Computing Systems*, ACM Press, NY (2008), 1453–1462.

Preece, Jenny, Rogers, Yvonne, and Sharp, Helen, *Interaction Design: Beyond Human-Computer Interaction*, 4th Edition, John Wiley & Sons, West Sussex, UK (2015).

Reiss, E., *Usable Usability*, Wiley (2012).

Rogers, Y, Yuill, N., and Marshall, P., Contrasting lab-based and in-the-wild studies for evaluating multi-user technologies, in B. Price, *The SAGE Handbook on Digital Technology and Research*, SAGE Publications (2013).

Roto, Virpi, and Lund, Arnie, On top of the user experience wave: How is our work changing, *Proceedings of the ACM Conference on Human Factors in Computing Systems, Extended Abstracts*, ACM Press, NY (2013), 2521–2524.

Rubin, Jeffrey, and Chisnell, Dana, *Handbook of Usability Testing*, 2nd Edition, John Wiley & Sons, Indianapolis, IN (2008).

Ryu, Young Sam, Mobile Phone Usability Questionnaire (MPUQ) and Automated Usability Evaluation, in Julie A. Jacko (Editor), *Proceedings of the 13th International Conference on Human-Computer Interaction. Part I: New Trends*, Springer-Verlag, Berlin, Heidelberg (2009), 349–351.

Sauro, J., and Lewis, J., *Quantifying the User Experience: Practical Statistics for User Research* Morgan Kaufmann (2012).

Schmettow, M., Sample size in usability studies, *Communications of the ACM 55*, 4 (April 2012), 64–70.

Spool, Jared, Surviving our success: Three radical recommendations, *Journal of Usability Studies 2,* 4 (August 2007), 155–161.

Tullis, Thomas, and Albert, William, *Measuring the User Experience: Collecting, Analyzing, and Presenting Usability Metrics*, 2nd Edition, Morgan Kaufmann Publishers/Elsevier (2013).

Vermeeren, A., Lai-Chong Law, E., Roto, V., Obrist, M., Hoonhout, J., and Vaananen-Vainio-Mattila, K., User experience evaluation methods: Current state and development needs, *NordiCHI 2010*, Reykjavik, Iceland (2010), 521–530.

Wharton, Cathleen, Rieman, John, Lewis, Clayton, and Polson, Peter, The cognitive walkthrough method: A practitioner's guide, in Nielsen, Jakob, and Mack, Robert (Editors), *Usability Inspection Methods*, John Wiley & Sons, New York (1994).

Wilson, Chauncey, *User Interface Inspection Methods: A User-centered Design Method*, Morgan Kaufmann (2013).

This page intentionally left blank


Yellow
Green
Blue
Purple
Gray
All Tags...

CHAPTER 6

Design Case Studies

> Prototyping helps you get ideas out of your head and into something more tangible—something you can feel, experience, work through, play with and test . . . you can't afford not to prototype on your next project.
>
> **Todd Zaki Warfel**
> *Prototyping: A Practitioners Guide*, 2009

> A prototype is worth a thousand meetings.
>
> **Mike Davidson**
> Vice President of Design for Twitter*

CHAPTER OUTLINE

*http://alvinalexander.com/photos/prototype-worth-thousand-meetings

6.1 Introduction

This chapter's case studies present design contexts and applications to let readers see how tradeoffs and choices are made. Readers may find the case studies valuable for encapsulating design learning and showing the challenges of a design context so that they can be shared within teams or across an organization.

The three case studies were chosen to cover this book's design methods. One example of the design methods is whiteboard or digital sketching (Buxton, 2007; Greenberg et al., 2011), where prototype screen designs are presented for discussion and collaboration using whiteboard drawing. A number of tools and apps exist to support this technique.

User-interface designs are often proposed on a napkin at a favorite coffee shop. Wireframes (Usability.gov, 2015) and supporting wireframing tools are popular to define an interface design. Other design methods include sticky notes placed strategically on a whiteboard sketch or computer-based mockup of a screen. Higher-fidelity screen prototypes can be generated to illustrate the state of the design by adding navigation options, icons, and animation for clarification of design decisions.

Case Study 1 is titled "Iterative Design Evaluation of Automated Teller Machines (ATMs)," a study in the user-interface process of developing ATMs with details on how to perform usability testing of ATMs. This case study is a good example of how an iterative HCI process could be performed, exposing potential roadblocks while illustrating in a specific example the processes described in previous chapters regarding user-interface design and development: observe, refine, design, implement, evaluate, iterate.

Case Study 2 is titled "Design Consistency at Apple Computer" (Apple, 2015a). This case study is part of the Apple Human Interfaces Guidelines (Apple, 2015b) that results in a perspective and suggested approach for practitioners. Many product manufacturers besides Apple have developed style guidelines to ensure a consistent user interface across multiple products. For example, a company developing multiple technology products would prefer that its user interface be consistent across the product lines, follow a corporate style that reflects branding, and ensure that it is easy for a new user to master a new product from the same manufacturer. Although arguably Apple may be one of the best at this, other industries, such as automobile manufacturers and medical equipment companies, work hard to pursue this goal.

Case Study 3: "Data-Driven Design at Volvo" (Wozniak et al., 2015) shows successful collaboration methods in action to solve a diverse, distributed corporate data-analysis problem. By using user-interface development process methods, data are retrieved and presented in a tailorable format that empowers the users to achieve their business and organization goals.

BOX 1.1

See also:

- Chapter 4, Design
- Chapter 5, Evaluation and the User Experience
- Chapter 12, Advancing the User Experience

The chapter concludes with general observations and a summary that compares and contrasts all three case studies and their importance. There are many design process models and user experience evaluation approaches. While reading about the case studies in this chapter, reflect on the process steps required for a successful outcome.

6.2 Case Study 1: Iterative Design Evaluation of Automated Teller Machines (ATMs)

Most of us have become familiar with ATMs of varying styles and sizes. Drive-thru, stand-up, kiosk-style, standalone ATM structure, part of a bank wall or lobby—these machines are everywhere. One can go to another bank and pay a fee if he or she needs cash right away. An individual can travel the world and get cash in the local currency just by inserting his or her ATM/debit card, entering the PIN, and making a few choices, hopefully in a language that the user understands. Many ATMs have multi-lingual options now. Many banking mobile apps allow for many of the same transactions that an ATM provides except getting cash. So, let's limit this case study to just physical ATMs, an example of which appears as Fig. 6.1.

As for any device, the user interface for ATMs has evolved, from a more primitive electronic keypad to a magnificent, immersive experience of touchscreen displays with animated advertisements, tones signaling completion of task steps or key presses, color and font choices to improve the appearance while remaining consistent with the bank brand, and the latest security features such as small mirrors to see behind the customer, security cameras recording the customer's presence for safety, copious lighting at night, and card entry points that hinder "skimming" or copying of ATM cards by thieves and fraudsters.

When learning about usability of the user interface for an ATM, after reviewing Chapters 1–5 of this text, go visit the nearest ATM. (Disclaimer: The authors certainly understand it takes more than a few minutes to understand usability methods and techniques, but please read on regarding this usability "experiment.")

FIGURE 6.1
Sample ATM.

Designers could run a stopwatch as the user withdraws a specified amount at multiple ATMs, record their movements, watch them move from keypad for PIN entry to touchscreen for the withdrawal step(s) with prompting for receipt—the experienced user is like a one-person band playing multiple instruments—to get that end result (cash in hand, with or without receipt, ATM card returned, account updated, quickly and safely). Visualize the statistical data that can be captured from this usability "experiment":

- Time to complete all tasks over a statistically significant set of ATMs
- Time expended for these ATM steps or "subtasks":
  1. Entrance into ATM (approach ATM, read instructions to get started, insert card, enter PIN, continue following prompted instructions)
  2. Enter commands to make withdrawal
  3. Receive cash, optional receipt, and card returned (with the preferred goal of leaving a positive balance in the account)
- Objective and subjective user feedback and contextual observation regarding user performance of the above ATM steps

To complicate things, let's add an eye-tracker, key-logging tool to record keyboard and/or touchscreen data entries, record any errors, document navigation steps taken, and have the user enunciate steps taken with commentary in a think-aloud protocol—for example, "I am now going to insert my ATM card into the machine." Consider having a team record this event in a video to analyze later as is done in user experience labs. An excellent set of guiding principles for user experience appears in Hartson and Pyla (2012).

The amount of data to analyze is growing! The previous chapter (Chapter 5) discusses how to structure this usability evaluation to make this process practical and finite.

Designers who study neighborhood ATMs and review current literature from ATM developers or other vendors can develop a useful competitive feature analysis. One intriguing design for an ATM kiosk in developing countries is discussed in Birnie (2011). Numerous ATM screenshot examples and ATM designs can be found by a quick web search, illustrating style alternatives worldwide of current ATM machine design.

Look at the ATM design for accessibility, i.e., universal usability (see Chapter 2). Consider some of the guidelines, principles, and theories that drive the design (Chapter 3), often resulting in a style guide that merges these concepts with product branding to ensure an end result that fits the business objective for the ATM. Of course, manage the design process in an organized, well-defined, user-centered, iterative fashion (Chapter 4).

Once this usability experimentation and literature search is complete, designers could enter the next life-cycle (design) phase. Think of this as an incremental continuous improvement. The data collected can be analyzed to arrive at concrete, data-driven design interventions that may improve the user experience. These alternative designs can then be sketched and prototyped as discussed in Buxton (2007) and Greenberg et al. (2011). Make sure to review Chapters 4–5 of this book for design and evaluation processes.

Designs can then be documented, tradeoffs between alternative designs evaluated, and specifications written for an improved ATM design. Iterative design is the best approach here, with design prototypes developed, evaluated, and improved. Again, striving to make this process complete yet finite is the challenge. Typically, a delivery deadline will drive the depth and fidelity of any prototyping effort—for example, the next-generation prototype ATM needs to appear and be operational at a trade show on a specific date and location. Some clients require "capability demonstrations," where the increasing fidelity of the prototype is shown in "proof of concept" demonstrations following a planned, incremental development strategy.

Sales commitments are made, final implementation continues, ATMs built, delivered, and fitted into a physical structure and integrated into the banking network, bank personnel are trained, customers are notified, and so on, to bring these products online.

Observations

Does the analysis of the usability of the newly installed ATM stop here? Certainly not. Continue gathering feedback from customers and monitoring implementation success. Consider rolling out test sites (e.g., beta testing) to first ensure new designs are accepted on a smaller scale. These business decisions are tightly coupled to the design and usability results discussed here. Ultimately, as in many user-interface designs, the success of a product is often judged by the user interface.

At this point in the life cycle of the ATM case study, the following possible scenario could occur: In the ideal situation, everything works perfectly and the bank clients love the new design. System performance is terrific, cost per transaction drops, profits are up, and customers flock to the bank to use the new ATMs.

Realistically, some changes may need to be made. There could be numerous unanticipated user transition and acceptance issues. The bank could hire someone to independently develop an alternative user experience. The bank captures data from the deployed systems to methodically (like a software upgrade) roll out improvements to the ATM network. The feedback loop continues.

The remainder of this chapter focuses on two specific design case studies and how the organizations approached their user-interface challenges.

6.3 Case Study 2: Design Consistency at Apple Computer

Case Study 2 examines the process and decisions reflected in an Apple document titled "From Desktop to iOS" (Apple, 2015a, 2015b). In this analysis, Apple examined products and design decisions made for Keynote® (for presentations), Mail (e-mail for iPhone), and web content.

The case study reviews style guidelines from the Apple Development Guidelines and how these were applied in bringing apps to iOS-enabled devices. For more information on related issues, see the iOS Human Interface Guidelines in the iOS Developer Library that Apple provides (Apple, 2015b). Following are a few samples of the referenced iOS Human Interface Guidelines (explanation, use, and screenshot illustrations are given for the guidelines):

- Take advantage of the whole screen
- Reconsider visual indicators of physicality and realism
- Let translucent user interface elements hint at the content behind them
- Let color simplify the user interface
- Ensure legibility by using the system fonts
- Use depth to communicate

There are guidelines for icons and image design, iOS technologies, user interface elements, and more.

Keynote has presentation development tools, graphics and toolbars for rapid generation of presentations. An example screenshot appears as Fig. 6.2.

FIGURE 6.2
Sample Keynote display with help text.

Presentation (graphics) styles as well as human interaction for touchscreen devices are included in the case study, illustrating the product's ease of use. iOS device–enabled features apply direct manipulation and gesturing interaction (see Chapter 7) that are easy to use.

Next the Apple study looks at Mail for iPhones. Fig. 6.3 illustrates Apple's intuitive, predictable navigation for Mail that has a user interface consistent with the Apple product line.

The case study finishes with a discussion on Safari on iOS devices, where again the mobile web-viewing user experience on iOS devices is easy to use and consistent with the product line and gives web designers valuable insight as to the portability and ease of development of web content for iOS devices. References to iOS Design Strategies consistent with the teachings in this text are included—specify the app, determine who the users are, identify desired features (requirements collection), preliminary and detailed design, build and development, evaluation and testing.

Observations

Some general observations are worth noting with respect to this case study. The years of experience with the referenced Apple Human Interface Guidelines are brought to bear on the problem. There is a consistent style across all products

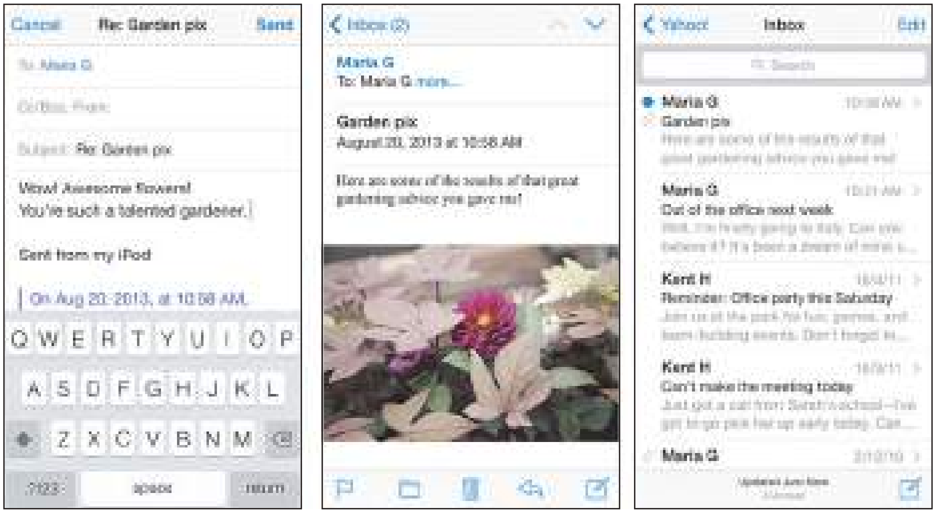


FIGURE 6.3

Sample iPhone Mail screens.

and devices that makes device operation comfortable and intuitive. Rapid device technology improvements (lighter weight, faster, more colors, more pixels, improved throughput, etc.) result in a constant reevaluation of the user interface and improvements of the guidelines applied. Still, the principles discussed in this text also hold true in this case study: universal usability, guidelines based on principles and theory, iterative user-centered design processes, and a keen appreciation of the user experience and of style.

The following case study shows a successful collaboration in a large corporation utilizing user-interface development process methods to solve a data-analysis challenge.

6.4 Case Study 3: Data-Driven Design at Volvo

The development of Volvo's big data service provides a terrific example of a case study with big data analytics used in the corporate world that contains a strong user-interface design component (Wozniak et al., 2015). But first a definition: *big data* is defined (Google, 2015) as

> Extremely large data sets that may be analyzed computationally to reveal patterns, trends, and associations, especially relating to human behavior and interactions.

In this case study, stakeholders were identified and empowered to help design the service (toolset) that would be used by the company. A diverse set of stakeholders (including many users) proved to be a success-oriented approach for this participatory design team. Stakeholders included the internal IT organization, an invited (external) expert on big data implementations, database engineers, and business intelligence analysts. Workshops were held for the stakeholders as well as users of the data such as the organizations in charge of vehicle maintenance. The workshop attendees (representative stakeholders) strived to define how the results might appear and how they could be applied to the various stakeholder group missions. Attendees were encouraged to "think outside the box" in terms of potential uses for the data. They had to make sure the data were indeed collectable and would make sense to improve organization performance.

Taking huge datasets of Volvo truck service data and essentially prototyping the analysis output that could be performed worked successfully to identify the needed data in useful formats. In the workshop, a low-fidelity prototype was developed (see Fig. 6.4).

Through a series of refinements with representative users worldwide, this prototype evolved into something useful for all concerned. Fig. 6.5 shows a sample final version. Users could bring up advanced information about their products (vehicles) such as service history statistics, perform queries on market-specific issues, see maps of vehicle usage, highlight interesting values, and more.

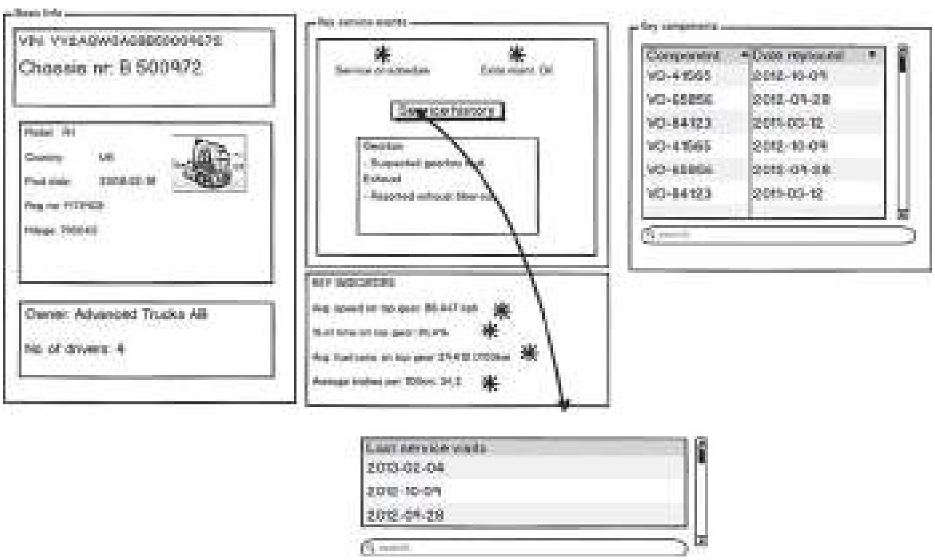


FIGURE 6.4

Low-fidelity prototype resulting from big data analysis of truck service statistics.

FIGURE 6.5

Fully functional dashboard prototype.

A big data service was developed with user-customizable reports. Customization played a crucial role in the design. Users involved in the process knew they would be able to choose features and rearrange views based on market-to-market differences. The study showed how following a user-interface development paradigm or development methodology led to a successful result that had buy-in from a distributed set of users. Indeed, if brokering organizational support is desired for a new innovation or product, a design-thinking workshop of the sort practiced in this use case can be an effective technique.

The performers of the study learned to first identify sources of data while empowering the stakeholders of the data to choose what data they could use (and how) in order to get their jobs done. Also, there were discussions in the workshops to analyze the data outputs. The final output was corporate big data policies that led to stakeholder-customizable report formats to better improve internal corporate communication and decision making. The tool is now used in all European Volvo truck dealerships.

Observations

There are some general observations worth making here. What was essentially a process for developing a big data analysis strategy, service design, and supporting tools for a company to use to increase internal communication and profitability turned out to apply methods and processes taken from the world of user experience design and designing user interfaces! The authors applied a well-known methodology and an interface design paradigm—the process worked.

Often, user-interface processes are applied to business processes without the persons leading the change realizing its origin. For example, to analyze patient flow in a hospital (patient recordkeeping, scheduling of resources, service bottlenecks, prioritization of patient needs, etc.) may sound to some like a simple database or queueing problem or an application of business-process reengineering. However, when humans are mixed with sources of diverse data often in a time-sensitive environment, quick and easy access to critical data can make an organization run more smoothly, better allocate resources, and lead to better decisions to improve customer satisfaction and patient outcomes. "Knowing thy user," optimizing access to meaningful data, iteratively gathering feedback data, getting stakeholder buy-in, and so on, can all be accomplished using user-interface design and development methods as was performed in this case study.

6.5 General Observations and Summary

This chapter's case studies are "a tip of the iceberg" in what can be accomplished by designers of user-interface systems. The case studies were chosen strategically to highlight design contexts, various applications, and incremental continuous improvement.

The ATM design example illustrated where what may have started out as a relatively straightforward task turned into a methodical study of how to improve a user interface to the machines and was not only accepted but embraced by the general banking customers. Clearly a competitive edge and source of profitability are the extension of the banking functions for customers via well-designed ATMs.

The Apple guidelines case study shows one company's approach to a consistent, easy-to-to use style for all the company's products and iOS-enabled devices. Lastly, the Volvo study shows how following a good user-interface design process can result in a successful conclusion with a large, data-intensive problem.

Additional sources for interesting user-interface case studies can be found on the web and in Snyder (2003), Righi and James (2007), Karat and Karat (2010), and Warfel (2011).

Practitioner's Summary

Interface designers are aware of the challenge of working with multi-disciplinary teams while striving for consensus in a timely manner to address the requirements for a new or updated system. The challenge here is that many use varying applications of interface design methodologies that are not standardized. What might work for one company, organization, or industry may not work for another.

Make sure to do some preliminary work up front to appreciate and understand the differences in development methodologies and to apply what makes sense for the application. This same rule applies to any software development task. Organizations can benefit from methods used elsewhere but must be carefully managed in order to achieve a successful result within schedule constraints. Defining the user and characterizing end user needs make up the engine to drive the successful user experience analysis.

Review interface designs for value-sensitive design issues—designs that center on human well-being, human dignity, justice, welfare, and human rights. Ensure interface designs meet universal usability—the design of information and communications products and services that are usable for every citizen (Friedman et al., 2013).

Researcher's Agenda

There is ample opportunity for research and experimentation with different interface design methodologies and how they interface with software development process models. Not starting from scratch, there are often examples on the web of similar development challenges that can be extrapolated for a need or application. It would be beneficial to develop a characterization of the nuances in different user-interface development methodologies and how they interact with current software development processes.

WORLD WIDE WEB RESOURCES

www.pearsonglobaleditions.com/shneiderman

Case study examples have a significant presence on the internet and are growing. Check out ACM SIGCHI "CHI" conferences, which hold practitioner's sessions. SIGCHI publishes "CHI Extended Abstracts" with example case studies available via the ACM Digital Library.

- ACM SIGCHI "CHI" conferences: http://www.sigchi.org/conferences

Discussion Questions

1. Consider additional requirements and technology to further complicate your analysis of an Automated Teller Machine (ATM) design:
   - Use eye-tracker data to further analyze the product.
   - Consider accessibility (universal usability) issues such as lighting, physical placement of ATM, etc.
   - Consider user profile issues, e.g., if a user is using an ATM for the first time.
   - Requirement to perform beta and/or market tests.
   - Are there other stress factors such as a looming time deadline or a personal safety issue?
2. Review the iOS Human Interface Guidelines in the iOS Developer Library at https://developer.apple.com/library/ios/documentation/UserExperience/Conceptual/MobileHIG/. In groups of two, select one guideline that makes perfect sense and seems easy to be incorporated into a design, and select another guideline that is much less clear, requiring further explanation or analysis to be incorporated into a design. Share these ideas with the class to see if there is any trend or pattern on the easy-to-do versus the harder-to-do guidelines.
3. At present, the drive to use Big Data to define or enhance corporate strategies seems to be a global business trend. State an example of where this data can improve a business, focusing on user-interface aspects.
4. Cite a past experience where user-interface development methods might apply to another system development activity that might not have a strong user-interface component.

References

Apple, *From Desktop to iOS* (2015a). Available at https://developer.apple.com/library/ios/documentation/UserExperience/Conceptual/MobileHIG/DesktopToiOS.html#//apple_ref/doc/uid/TP40006556-CH51-SW1.

Apple, *iOS Human Interface Guidelines* (2015b). Available at https://developer.apple.com/library/ios/documentation/UserExperience/Conceptual/MobileHIG/.

Birnie, S., The pillar ATM: NCR and community centered innovation, *DPPI '11 Proceedings of the 2011 Conference on Designing Pleasurable Products and Interfaces*, ACM (2011).

Buxton, W., *Sketching User Experiences: Getting the Design Right and the Right Design (Interactive Technologies),* New York: Morgan Kaufmann (2007), 77–80, 135–138.

Friedman, B., Kahn Jr., P. H., Borning, A., and Huldtgren, A., Value sensitive design and information systems, in Doorn et al. (Editors), *Early Engagement and New Technologies: Opening Up the Laboratory*, Springer (2013), 55–95.

Google, Big Data (2015). Available at https://www.google.com/#q=what+is+big+data.

Greenberg, S., Carpendale, S., Marquardt, N., and Buxton, B., *Sketching User Experiences: The Workbook,* San Francisco: Morgan Kaufmann (2011), 29–66.

Hartson, R., and Pyla, P., *The UX Book: Process and Guidelines for Ensuring a Quality User Experience*, Morgan Kaufmann (2012).

Karat, C., and Karat, J., Designing and evaluating usable technology in industrial research three case studies, *Synthesis Lectures on Human-Centered Informatics 3,* 1 (2010), 1–118.

Righi, C., and James, J., *User-Centered Design Stories: Real-World UCD Case Studies (Interactive Technologies),* San Francisco: Morgan-Kaufman (2007).

Snyder, C., *Paper Prototyping: The Fast and Easy Way to Design and Refine User Interfaces*, San Francisco: Morgan Kaufmann (2003).

Usability.gov, Wireframing (2015). Available at http://www.usability.gov/how-to-and-tools/methods/wireframing.html.

Warfel, T. Z., *Prototyping: A Practitioner's Guide*, Rosenfeld Media (2011).

Wozniak, P., Valton, R., and Fjeld, M., Volvo single view of vehicle: Building a Big Data service from scratch in the automotive industry, *CHI EA '15: Proceedings of the 33rd Annual ACM Conference Extended Abstracts on Human Factors in Computing Systems*, ACM (2015), 671–678.

This page intentionally left blank

PART 3

INTERACTION STYLES

PART OUTLINE

Interaction styles are the "bread and butter" of the interface. Part 3 covers the actions, objects, and approaches used by the designers to assemble the interface. Various conventions have been developed and should be followed when designing interfaces. One goal is to use familiar objects and metaphors. This helps minimize the learning curve for the user and allows users to quickly become productive.

Computers originally were controlled mostly by keyboard and mouse. As computing advanced, more direct (or "natural") pointing and gesture interfaces emerged as well as the desire to "talk" to the interface using speech. Chapter 7 discusses the principles of direct manipulation and how they are used in the interface. This chapter also goes into immersive environments including augmented and virtual reality. Chapter 8 discusses fluid navigation. Navigation is critical as it allows users to find the information they want or more generally express their choices. The use of voice commands and human-language technology in general is growing quickly, but users can also easily learn and use structured command languages. Chapter 9 discusses language and its use. The artifacts that are used to control the interface are discussed in Chapter 10 with devices. Novel devices are being developed every day. The last chapter in this section is Chapter 11, which discusses communication and collaboration. Today, almost no one uses a computer just as a single, isolated individual. Social media applications fill our world as users participate in Facebook, post tweets and restaurant reviews, or edit Wikipedia pages.

This book is all about design, but this set of chapters discusses the tools and artifacts needed to create exciting, stimulating, and usable interaction styles to enhance user experiences.

CHAPTER 7

Direct Manipulation and Immersive Environments

"Leibniz sought to make the form of a symbol reflect its content. 'In signs,' he wrote, 'one sees an advantage for discovery that is greatest when they express the exact nature of a thing briefly and, as it were, picture it; then, indeed, the labor of thought is wonderfully diminished.'"

Frederick Kreiling
"Leibniz," *Scientific American*, May 1968

CHAPTER OUTLINE

7.1 Introduction

Certain interactive systems generate a glowing enthusiasm among users that is in marked contrast with the more common reaction of reluctant acceptance or troubled confusion. The enthusiastic users report the following positive feelings:

- Mastery of the interface
- Competence in performing tasks
- Ease in learning originally and in assimilating advanced features
- Confidence in the capacity to retain mastery over time
- Enjoyment in using the interface
- Eagerness to show off the interface to novices
- Desire to explore more powerful aspects

These feelings convey an image of a truly pleased user. The central ideas in such satisfying interfaces, now widely referred to as *direct-manipulation interfaces* (Shneiderman, 1983), are visibility of the objects and actions of interest; rapid, reversible, incremental actions; and replacement of typed commands by a pointing action on the object of interest. Direct-manipulation ideas are at the heart of many contemporary and advanced non-desktop interfaces. Game designers continue to lead the way in creating visually compelling 3-D scenes with characters (sometimes designed and user-created) controlled by novel pointing devices. At the same time, interest in remote-operated (teleoperated) devices has blossomed, enabling operators to look through distant microscopes or fly drones. As the technology platforms mature, direct manipulation increasingly influences designers of mobile devices and webpages. It also inspires designers of information-visualization systems that present thousands of objects on the screen with dynamic user controls (Chapter 16).

Newer concepts that extend direct manipulation include virtual reality, augmented reality, and other tangible and touchable user interfaces. Augmented reality keeps users in the normal surroundings but adds a transparent overlay with information such as the names of buildings or visualizations of hidden objects. Tangible and touchable user interfaces give users physical objects to manipulate so as to operate the interface—for example, putting several plastic blocks near to each other to create an office floor plan. Virtual reality puts users in an immersive environment in which the normal surroundings are blocked out by a head-mounted display that presents an artificial world; hand gestures allow users to point, select, grasp, and navigate. All of these concepts are being applied not only in individual interactions but also in wider artificial worlds, creating collaborative efforts and other types of social-media interactions.

This chapter defines the principles, attributes, and problems of direct manipulation, including a way to categorize direct manipulation (Section 7.2). Some examples of direct-manipulation use are provided in Section 7.3. Section 7.4 discusses 2-D and 3-D interfaces. Teleoperation and presence are covered in Section 7.5. Lastly, augmented and virtual reality are discussed Section 7.6. Although the tenets of direct manipulation still hold true, regardless of the sophistication of the technology, the technology in this chapter is advancing rapidly. The references for this chapter include a combination of books and articles. The articles are taken from the recent conference proceedings showing some of the innovations and interesting projects being developed in research labs of industries and academia. Many pundits and popular press sources (Kushner, 2014; Kofman, 2015; Metz, 2015; Mims, 2015; Stein, 2015) feel the time for virtual and augmented reality is now. Researchers are looking into the theoretical challenges and opportunities in virtual worlds (de Castell et al., 2012) and continuing to improve upon the gaming experience (Kulshreshth and LaViola, 2015).

See also:
Chapter 10, Devices
Chapter 16, Data Visualization

7.2 What Is Direct Manipulation?

Direct manipulation as a concept has been around since before computers. The metaphor of direct manipulation works well in computing environments and was introduced in the early days of Xerox PARC and then widely disseminated by Shneiderman (1983). Direct-manipulation designs can provide the capability for differing populations and easily stretch across international boundaries. Section 7.2.1 explains the three principles of direct manipulation and advantages of using direct manipulation. Section 7.2.2 provides a way of discussing direct manipulation using a translational concept of strength. Section 7.2.3 discusses some problems with direct manipulation. Section 7.2.4 discusses the continuing evolution of direct manipulation.

A favorite example of direct manipulation is driving an automobile. The scene is directly visible through the front window, and performance of actions such as braking and steering has become common knowledge in our culture. To turn left, for example, the driver simply rotates the steering wheel to the left. The response is immediate and the scene changes, providing feedback to refine the turn. Now imagine how difficult it would be trying to accurately turn a car by typing a command or selecting “turn left 30 degrees” from a menu. The graceful interaction in many applications is due to the increasingly elegant application of direct manipulation. Although there is lively discussion on the impact of driverless cars and their uses, research still continues. Driverless cars may soon respond to commands like "take me to Baltimore airport," but they are a long way from matching the skills of drivers at the wheel while navigating snow-covered roads or police hand signals at accident sites.

Before designing for current devices, it makes sense to reflect where early design has been. In the early days of office automation, there was no such thing as a direct-manipulation word processor or a presentation system like PowerPoint. Word processors were command-line–driven programs where the user typically saw a single line at a time. Keyboard commands were used along with inserting special commands to provide instructions for viewing and printing the documents often as a separate operation. Similarly, with presentation programs, specialized commands were used to set the font style, color, and size. Obviously, these were very limited compared to the numerous font families available today. Most users today are used to a WYSIWYG (What You See Is What You Get) environment enhanced by direct-manipulation widgets.

7.2.1 The three principles and attributes of direct manipulation

The attraction of direct manipulation is apparent in the enthusiasm of the users. The designers of the examples, provided in Section 7.3, had an innovative inspiration and an intuitive grasp of what users would want. Each example has problematic features, but they demonstrate the potent advantages of direct manipulation, which can be summarized by three principles:

1. Continuous representations of the objects and actions of interest with meaningful visual metaphors
2. Physical actions or presses of labeled interface objects (i.e., buttons) instead of complex syntax
3. Rapid, incremental, reversible actions whose effects on the objects of interest are visible immediately

Simple metaphors or analogies with a minimal set of concepts—for example, pencils and paintbrushes in a drawing tool—are a good starting point. Mixing metaphors from two sources may add complexity that contributes to confusion. Also, the emotional tone of the metaphor should be inviting rather than distasteful or inappropriate. Since the users are not guaranteed to share the designer's understanding of the metaphor, analogy, or conceptual model used, ample testing is required.

Using these three principles, it is possible to design systems that have these beneficial attributes:

- Novices can learn basic functionality quickly, usually through a demonstration by a more experienced user.
- Experts can work rapidly to carry out a wide range of tasks, even defining new functions and features.

- Knowledgeable intermittent users can retain operational concepts.
- Error messages are rarely needed.
- Users can immediately see whether their actions are furthering their goals, and if the actions are counterproductive, they can simply change the direction of their activity.
- Users experience less anxiety because the interface is comprehensible and because actions can be reversed easily.
- Users gain a sense of confidence and mastery because they are the initiators of action, they feel in control, and they can predict the interface's responses.

In contrast to textual descriptors, dealing with visual representations of objects may be more "natural" and in line with innate human capabilities: Action and visual skills emerged well before language in human evolution. Psychologists have long known that people grasp spatial relationships and actions more quickly when they are given visual rather than linguistic representations. Furthermore, intuition and discovery are often promoted by suitable visual representations of formal mathematical systems.

7.2.2 Translational distances with direct manipulation

The effectiveness and reality of the direct-manipulation interface are based on the validity and strength of the metaphor chosen to represent the actions and objects. Using familiar metaphors creates easier learning conditions for users and lessens the number of mistakes and incorrect actions. Adequate testing is needed to validate the metaphor. Special attention needs to be paid to the user characteristics such as age, reading level, educational background, prior experiences, and any physical disabilities.

One way of trying to understand and categorize the direct-manipulation metaphor is by looking at the *translational distance* between users and the representation of the metaphor, which will be referred to as strength. Strength can be perceived along a continuum from weak to immersive (See Box 7.1). This can be further described as the level of indirectness between the user's physical actions and the actions in the virtual space.

BOX 7.1

Examples of translational distances (strength).

- Weak—early video game controllers (Fig. 7.5)
- Medium—touchscreens, multi-touch (Fig. 7.1)
- Strong—data glove, gesturing, manipulating tangible objects (Fig. 7.2)
- Immersive—virtual reality, i.e, oculus rift (Fig. 7.14)

Weak direct manipulation is what can be described as basic direct manipulation. There is a mouse, trackpad, joystick, or similar device translating the user's physical action into action in the virtual space using some mapping function. The translational difference is large because interaction is completely indirect. For example, the user moves the mouse on a 2-D desk within a small circumscribed region and the mouse moves on the screen (again 2-D). Because this mapping function is not always fully understood and processed correctly by the user, sometimes the user will actually run the mouse off the surface of the desk. Weak direct manipulation was used with early game controllers that provided buttons and joysticks, where the action of the controllers needed to be learned explicitly by the players.

Medium direct manipulation is the next step moving along the continuum. The translational distance is reduced. Instead of communicating with the virtual space with the device, the user reaches out and touches, moves, and grabs the entities in the on-screen representation. Examples of this include touchscreens (mobile, kiosk, and desktop). This is still limited by the glass of the screen; the world is beyond the glass. This direct-manipulation strength supports pointing

FIGURE 7.1

Three users working concurrently on a large tabletop touch device. They can use their hands/figures to manipulate the objects on the device. Note the different hand gestures being used. (www.reflectivethinking.com) and tapping, but other activities that would include the third dimension, like reaching into the device, cannot be accommodated by the simple metaphor. Instead, creating these other actions requires stepping outside the metaphor with a new artifact such as double-tap and assigning a corresponding action to it. This again requires learning on the user's part. Multi-touch (Fig. 7.1) allows new actions to be assigned to various combinations of finger touches. The two-finger actions like zoom in/out are intuitive, but others must be learned and take longer to discover. This accounts for why a young child can easily learn to tap, change screens, and touch on a tablet (the intuitive actions) but doesn't have the skills to rearrange the icons on the screen (the learned actions).

Strong direct manipulation involves actions such as gesture recognition with various body parts. It may be the user's hand, foot, head, or full body (whatever controls the action) that is "virtually" placed inside the physical space (Fig. 7.2). The users can see their hand in the 3-D space and can grasp, throw, drop,

FIGURE 7.2

A tangible user interface for molecular biology, developed in Art Olson's Laboratory at the Scripps Research Institute, utilizes autofabricated molecular models tracked with the Augmented Reality Toolkit from the University of Washington Human Interface Technology Lab. The video camera on the laptop captures the molecule's position and orientation, enabling the molecular modeling software to display information such as the attractive/repulsive forces surrounding the molecule.

manipulate, and so forth. The users themselves still remain on the outside looking in. This works well when the spaces are small and simple, but when the spaces get bigger, the users need to move themselves outside the initial metaphor and enter another mode, such as move mode, and then traverse to the new region. See Chapter 10 for more on devices.

The notion of tangible and immersive user interfaces—in which users grasp physical objects to manipulate a graphical display that represents the object are becoming quite popular. Tangible devices use haptic interaction skills to manipulate objects and convert the physical form to a digital form (Ishii, 2008).

The last dimension is *immersive* direct manipulation. Here is where direct manipulation is combined with virtual reality (see Section 7.6). The users put on glasses or some other device and they are inside the space. The users can see themselves and can walk/fly through the space by walking, leaning in, and so forth—the scenery changes with the moves.

7.2.3 Problems with direct manipulation

Graphical user interfaces were a setback for vision-impaired users, who appreciated the simplicity of linear command languages. However, screen readers for interfaces, speech-enabled devices, page readers for browsers, and audio designs for mobile devices enable vision-impaired users to understand some of the spatial relationships necessary to achieve their goals.

Direct-manipulation designs may consume valuable screen space and thus force valuable information off-screen, requiring scrolling or multiple actions. This is an issue in the mobile world, where screen space is very limited.

Another issue is that users must learn the meanings of visual representations and graphic icons. Titles that appear on icons (flyover help) when the cursor is over them offer only a partial solution. The visual representation may sometimes be misleading. Users may grasp the analogical representation rapidly but then may draw incorrect conclusions about permissible actions, overestimating or underestimating the functions of the computer-based analogy. Ample testing must be carried out to refine the displayed objects and actions and to minimize negative side effects.

For experienced typists, taking a hand off the keyboard to move a mouse or point with a finger may take more time than typing the relevant command. This problem is especially likely to occur if the users are familiar with a compact notation, such as for arithmetic expressions, that is easy to enter from a keyboard but may be more difficult to select with a mouse. While direct manipulation is often defined as replacing typing of commands with pointing with devices, sometimes the keyboard is the most effective direct-manipulation device. Rapid keyboard interaction can be extremely attractive for expert users, but the visual feedback must be equally rapid and comprehensible.

Small mobile devices have limited screen sizes. A finger pointing at a device may partially block the display, rendering a good portion of the device not visible. Also, if the icons are small because of the limited screen size, they may be hard to select or, because of limited resolution and viewing capabilities (especially for older adults), not clearly distinguishable, resulting in their meanings becoming lost or confused.

Some direct-manipulation principles can be surprisingly difficult to realize in software. Rapid and incremental actions have two strong implications: a fast perception/action loop (less than 100 ms) and reversibility (the undo action). A standard database query may take a few seconds to perform, so implementing a direct-manipulation interface on top of a database may require special programming techniques. The undo action may be even harder to implement, as it requires that each user action be recorded and that reverse actions be defined. It changes the style of programming because a nonreversible action is implemented by a simple function call whereas a reversible action requires recording the inverse action.

7.2.4 The continuing evolution of direct manipulation

A successful direct-manipulation interface must present an appropriate representation or model of reality. With some applications, the jump to visual language may be difficult, but after using visual direct-manipulation interfaces, most users and designers can hardly imagine why anyone would want to use a complex syntactic notation to describe an essentially visual process. It is hard to conceive of learning the commands for the vast number of features in modern word processors, drawing programs, or spreadsheets, but the visual cues, icons, menus, and dialog boxes make it possible for even intermittent users to succeed. See Box 7.2 for a summary of the advantages and disadvantages of direct manipulation.

BOX 7.2

Advantages and disadvantages of direct manipulation.

Direct Manipulation

Advantages	Disadvantages
• Visually presents task concepts	• May be hard to program
• Allows easy learning	• Accessibility requires special attention
• Allows easy retention	
• Allows errors to be avoided	
• Encourages exploration	
• Affords high subjective satisfaction	

Users are trying to better understand all the data and other visual content that are now available. One way to manage this information is through the use of a *dashboard* (Few, 2013). Being able to see a large volume of information (big data) at one time and to directly manipulate it and observe the impact visually is a powerful concept. Businesses and companies are bombarded by volumes of data every day. The ability to organize this user-generated data into a useful graphical format can help them manage resources and spot trends (Chapter 16). Dashboards provide ways for users to manipulate data using the various widgets provided. Companies such as Tableau Software, SAP Lumira, and IBM Cognos provide this capability as do smaller user-oriented companies like dashboardsbyexample.

Weiser's (1991) influential vision of ubiquitous computing described a world where computational devices were everywhere—in your hands, on your body, in your car, built into your home, and pervasively distributed in your environment. The 1993 special issue of *Communications of the ACM* (Wellner et al., 1993) showed provocative prototypes that refined Weiser's vision. It offered multiple visions of beyond-the-desktop designs that used freehand gestures and small mobile devices whose displays changed depending on where users stood and how they pointed the devices. Almost 25 years later, Weiser's full vision has not yet been realized, but the social-media aspect of ubiquitous computing has blossomed.

Touchable displays from the small to the large (as large as wall size [Figs. 10.20 and 10.21] or even mall size) are becoming available as well. Interaction is all accomplished without users entering a long string of commands; instead, users physically manipulate the items of interest with their hands. An application of this is often seen on news programs, where the commentator can move the objects of interest on the screen and drill down to more detailed levels. Another application is virtual maps, which can be manipulated and zoomed by using hand motions as a multi-touch interface (Han, 2005). On a touchable display, interactions with both hands seem quite natural (although with small displays, issues of occlusion can be problematic).

There will certainly be many future variations of and extensions to direct manipulation, but the basic goals will remain similar: comprehensible interfaces that enable rapid learning, predictable and controllable actions, and appropriate feedback to confirm progress. Direct manipulation has the power to attract users because it is rapid, and even enjoyable. If actions are simple, reversibility is ensured, retention is easy, anxiety recedes, users feel in control, and satisfaction flows in.

7.3 Some Examples of Direct Manipulation

No single interface has every admirable attribute or design feature—such an interface might not be possible. Each of the examples discussed here, however, has sufficient features to win the enthusiastic support of many users.

7.3.1 Geographical systems including GPS (global positioning systems)

For centuries, travelers have relied on maps and globes to better understand the Earth and geographical systems. As graphic- and image-capture capabilities increased (both real-world and human-generated), it was a natural progression to create systems to represent both a current location—"where we are"—and a target location—"where we want to go." Of course, as prices dropped, these types of systems became available as commercial GPS systems for cars, for walking, and even for the mobile phone. Being able to directly see the alternatives on the devices as well as how to move from the current location to the target location including manipulating the routes is another application of direct manipulation.

Google Maps™, MapQuest, Google Street View, Garmin, National Geographic, and Google Earth™ combine geographic information from aerial photographs, satellite imagery, and other sources to create a vast database of graphical information that can easily be viewed and displayed. In some areas, the detail can go down to an individual house on a street or even inside a building (Fig. 7.3). With the well-populated databases of geographic *points of interest*, these systems provide an easy-to-use facility to point and select the nearest gas station or specific type of restaurant. Some systems provide real-time traffic to facilitate alternative routing in traffic-laden situations.

FIGURE 7.3

This is a screenshot from Google Street View of the inside of the University Center at Nova Southeastern University in Florida. On the bottom is a scrollable image of other views on campus. In the bottom left corner is a more conventional static map showing the physical street location of the campus. Users can move the "person" to a different location on campus, and the views will change accordingly.

7.3.2 Video games

For many people, the most exciting, well-engineered, and commercially successful application of the direct-manipulation concepts lies in the world of video games. The early but simple and popular game *Pong*® (created in 1972) required the user to rotate a knob that moved a white rectangle on the screen. A white spot acted as a ping-pong ball that ricocheted off the wall and had to be hit back by the movable white rectangle. Users developed speed and accuracy in placing the "paddle" to keep the increasingly speedy ball from getting past, while the computer speaker emitted a ponging sound when the ball bounced. Watching someone else play for 30 seconds is all the training that a person needs to become a competent novice, but many hours of practice are required to become a skilled expert. The interface objects were a single paddle, a ball, a single player, and some rudimentary sound. Games have come a long way with various controls, including full body, multiple objects of interest (both good and evil), full stereo sounds, detailed graphical environments, changing backgrounds, and the possibility of multiple players sitting physically next to one another or virtually across the globe.

Some cataloguers state that we are in the eighth generation of video games. Parkin (2014) provides an illustrated history of five decades of video games. Last generation's Nintendo Wii, Sony PlayStation 3, and Microsoft Xbox 360™ have given way to this generation's Nintendo Wii U, Sony PlayStation 4, and Microsoft Xbox One in a very short time, and continued advances are expected. These gaming platforms have brought powerful 3-D graphics hardware to the home and have created a remarkable international market. Gaming experiences are being enhanced by combining 3-D user-interface technologies, such as stereoscopic 3-D, head tracking, and finger-count gestures (Kulshreshth and LaViola, 2015). For a detailed survey of visual, mixed, and augmented reality gaming, refer to Thomas (2012).

Wildly successful games include violent first-person shooters, fast-paced racing games, and more sedate golfing games. Small handheld game devices still exist, but now users are playing games on their phones and other mobile devices. Multi-player games on the internet have also caught on with many users providing the additional opportunity for social encounters and competitions. Gaming magazines and conferences attest to the widespread interest. In Rochester, New York, part of the Museum of Play houses the International Center for History of Electronic Games (http://www.museumofplay.org/icheg).

There is a wide genre of games, and the borders between genres are becoming blurred. Some games are single-player games; others have multiple players. For a list of gaming genre acronyms, see Box 7.3. Players can be in the same physical space or a different physical space but shared virtual space. Players themselves can be virtual. For a more complete taxonomy of gaming systems, see Pagulayan et al. (2012). In conducting research with player performance and

BOX 7.3

Gaming genre acronyms.

The computer world is filled with a list of gaming genre acronyms. Some of the more widely used acronyms include:

- AA—action adventure games
- ARPG—action role play games
- FPR—first-person shooter
- MMORPG—massively multi-player online role-playing games
- MOBA—massive online battle arena
- RPG—role-playing games
- RTS—real-time shooter

experience, the games with multiple players seem to hold more interest in the social connection with others, teamwork, and collaboration. The single-player games seem to focus more on the game narrative and the characters, and players show more interest in the degree of immersion (Johnson et al., 2015).

Game environments provide intriguing, successful applications of 3-D representations. These include first-person action games in which users patrol city streets or race down castle corridors while shooting at opponents as well as role-playing fantasy games with beautifully illustrated island havens or mountain strongholds. Many games are socially enriched by allowing users to choose avatars to represent themselves. Users can choose avatars that resemble themselves, but often they choose bizarre characters or fantasy images with desirable characteristics such as unusual strength or beauty (Boellstorff, 2008).

Some web-based game environments may involve millions of users and thousands of user-constructed "worlds," such as schools, shopping malls, or urban neighborhoods. Game devotees may spend dozens of hours per week immersed in their virtual worlds, chatting with collaborators or negotiating with opponents. *World of Warcraft* (developed and published by Blizzard Entertainment) has been the mainstay and most popular of the MMORPG games with more than 5.6 million subscribers as of 2015 (Fig. 7.4). New games are constantly hitting the market, and the competition is fierce. A relatively new entry to the market (2012), *Guild Wars 2* (developed by Arena Net and published by NCsoft) already has sold more than 5 million copies. This game is slightly different from other MMORPG games because the game is responsive to individual player actions, which is more common in single-player role-playing games.

The Nintendo Wii, introduced in 2006, changed the demographics of the gaming world. Instead of young children (typically boys), older adults were using the

FIGURE 7.4
A woman playing *World of Warcraft*. She is using both her keyboard and mouse. She also can hear the sounds of the game via her headset.

Wii to play games like tennis and bowling. It also became an early fitness/wellness platform. With the introduction of the Kinect by Microsoft for Xbox in 2010 and then for Windows in 2012, more worlds opened up, and with a software development kit (SDK), developers can create their own worlds. These interfaces have been referred to as a *natural* user interface because the entire body can be used, but the possible actions still remain limited and need to be learned. The early Wii controller was modified with the addition of a wrist strap, since gamers were so immersed in play they sometimes accidentally hurled the controller at the screen. There is no syntax to remember, and therefore there are no syntax-error messages. Error messages in general are rare because the results of actions are obvious and can be reversed easily: If users move themselves too far to the left, they merely use the natural inverse action of moving back to the right. These principles, which have been shown to increase user satisfaction, could be applied to other environments. Examples of various game controllers are shown in Fig. 7.5. Customized controllers exist for games such as *Guitar Hero* (Fig. 1.8), flight control (Fig. 10.9), and Leap Motion (Fig. 10.16).

FIGURE 7.5
Various game controllers. Some are very specific and include a steering wheel or joystick; others use a series of buttons and direction arrows. The Wii Controller with the wrist strap is shown in the upper right corner. Although these game controllers do provide direct-manipulation actions, users still have to learn the meaning of the various buttons.

Most games continuously display a numeric score so that users can measure their progress and compete with their previous performance, with friends, or with the highest scorers. Typically, the 10 highest scorers get to store their initials in the game for public display. This strategy provides one form of positive reinforcement that encourages mastery. Studies with elementary-school children have shown that continuous display of scores is extremely valuable. Machine-generated feedback—such as "Very good" or "You're doing great!"—

is not as effective, since the same score carries different meanings for different people. Most users prefer to make their own subjective judgments and perceive the machine-generated messages as an annoyance and a deception. Providing this combination of behavioral data and attitudinal data adds to the immersion quality of the game (Pagulayan et al., 2012).

Although the marketing focus and consumer popularity have concentrated on action-type games, there are other game environments, and gaming (or gamification) has become a popular metaphor used in training and evaluation. Simulation and educational games abound. Games have been developed for young children (pre-readers) where the intuitiveness of the icons and real-world–type interfaces (buttons, sliders, finger pointing, etc.) control the game. Females seem more interested in role-playing games and games with narratives. A whole new generation of female gamers now exists. Games are also used for wellness benefits (Calvo and Peters, 2014; Jones et al., 2014). Researchers are trying to better understand how users think and get into their *flow state* (Csikszentmihalyi, 1990; Ossola, 2015). Gaming can be used to learn and enhance physical skills, modify behaviors, and increase wellness. Although there are some negative implications of gaming, McGonigal (2011) offers some rules for the positive impact of gaming: limit yourself to no more than 21 hours a week; play games face to face with friends and family; and play cooperative games or games that have a creator mode.

Studying game design is fun (Lecky-Thompson, 2008), but there are limits to the applicability of the lessons. Game players are engaged in competition with the system or with other players, whereas applications-systems users prefer a strong internal locus of control, which gives them the sense of being in charge. Likewise, whereas game players seek entertainment and focus on the challenge, application users focus on their tasks and may resent too many playful distractions. The random events that occur in most games are meant to challenge the users; in non-game designs, however, predictable system behavior is preferred. Throughout this book, we discuss the user experience (UX); the gaming world now designs for the player experience (PX). Research is continuing with the development of a growing set of metrics to measure PX (Johnson et al., 2015). Additional research is ongoing in the quantification and evaluation of playfulness providing meaningful and memorable experiences (Lucero et al., 2014).

Courses and majors (or minors) in video game design exist. Some are in computer science departments, but others show the more interdisciplinary nature of the subject and can be found in media design, visual communication, and art departments. The important take-away is to use clear affordances, good instructions, and informative feedback; limit the complexity; and be aware of human variability (Fisher et al., 2014). All these are basic tenets of HCI design as described in Section 3.3.4 (The Eight Golden Rules of Interface Design).

7.3.3 Computer-aided design and fabrication

Most computer-aided design (CAD) systems for automobiles, electronic circuitry, aircraft, or mechanical engineering use principles of direct manipulation. Building and home architects now have at their disposal powerful tools, provided by companies such as Autodesk, that provide components to handle structural engineering, floor plans, interiors, landscaping, plumbing, electrical installation, and much more. With such applications, the designer may see a circuit schematic on the screen and, with mouse clicks, be able to move components into or out of the proposed circuit. When the design is complete, the computer can provide information about current, voltage drops, and fabrication costs and warnings about inconsistencies or manufacturing problems. Similarly, newspaper-layout artists or automobile-body designers can easily try multiple designs in minutes and can record promising approaches until they find even better ones. The pleasure of using these systems stems from the capacity to manipulate the object of interest directly and to generate multiple alternatives rapidly.

There are large manufacturing companies using AutoCAD® and similar systems, but there are also other specialized design programs for kitchen and bathroom layouts, landscaping plans, and other homeowner-type situations. These programs allow users to control the angle of the sun during the various seasons to see the impact of the landscaping and shadows on various portions of the house. They allow users to view a kitchen layout and calculate square footage estimates for floors and countertops and even print out materials lists directly from the software. Some of the players in the field of interior-design software for residential and commercial markets include Floored, Inc. (Fig. 7.6), 2020 Spaces, and Home Designer Software. Their products are designed to work across multiple environments, desktop to web; they provide various views (top-down, architectural, front-view) to generate a more realistic overview of the design for the client.

Related applications are for computer-aided manufacturing (CAM) and process control. Honeywell's Experion® Process Knowledge System Orion provides the manager of an oil refinery, paper mill, or power-utility plant with a colored schematic view of the plant. The schematic may be displayed on multiple displays or on a large wall-sized map, with red lines indicating any sensor values that are out of the normal range. With a single click, the operator can get a more detailed view of the troubling component; with a second click, the operator can examine individual sensors or can reset valves and circuits. A basic strategy for this design is to eliminate the need for complex commands that the operator might need to recall only during a once-a-year emergency. The visual overview provided by the schematic facilitates problem solving by analogy because the linkage between the screen representations and the plant's temperatures or pressures is so close. The latest version of this software provides capabilities for

FIGURE 7.6

An office space layout from a company called Floored, Inc. This 3-D virtual CAD representation helps designers lay out office space. Items can be moved around between and within rooms; the design will be re-created to reflect any changes (http://www.floored.com).

virtualization and cloud support and includes customized dashboards to show status.

Another emerging use of direct manipulation involves home automation. Since so much of home control involves floor plans, direct-manipulation actions naturally take place on a display of the floor plan with selectable icons for each status indicator (such as a burglar alarm, heat sensor, or smoke detector) and for each activator (such as controls for opening and closing curtains or shades, for air conditioning and heating, or for audio and video speakers or screens). For example, users can route a recorded TV program being watched in the living room to the bedroom and kitchen by merely dragging the on-screen icon into those rooms, and they can adjust the volume by moving a marker on a linear scale. The action is usually immediate and visible and can be easily reversed as well.

With the advent of these types of systems, not only are graphical, sophisticated 3-D displays generated, but with 3-D printing technology, actual workable models can be generated. These models provide a more realistic view for clients and customers. These models can include an overall outside view or even be broken down to show component parts if necessary. The cost saving of these models versus building the actual structure or device can be enormous coupled

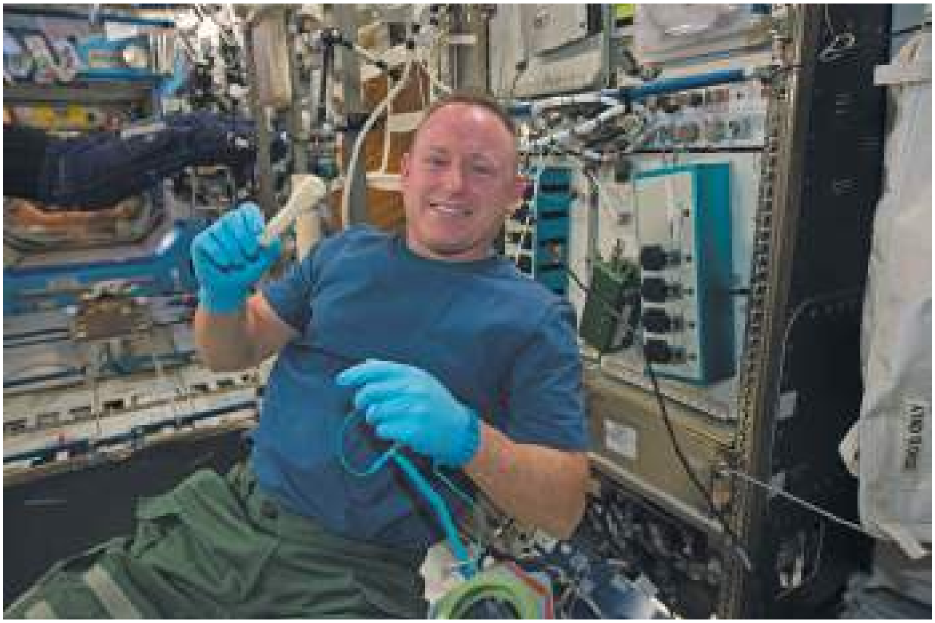


FIGURE 7.7

Astronaut Bruce Wilmore onboard the International Space Station with the ratchet wrench that was created with Made in Space's 3-D printer. This device was designed, qualified, tested, and printed in space in less than one week.

with the ease for incremental or larger modification or changes. 3-D printers have been installed on the NASA space station, where actual parts can be fabricated (Fig. 7.7).

7.3.4 Direct-manipulation programming and configuration

Performing tasks by direct manipulation is not the only goal. It should be possible to do programming by direct manipulation as well, at least for certain problems. How about moving a drill press or a surgical tool through a complex series of motions that are then repeated exactly? Automobile seating positions and mirror settings can be set as a group of preferences for a particular driver and then adjusted as the driver settles in place. Likewise, some professional television-camera supports allow the operator to program a sequence of pans or zooms and then to replay it smoothly when required.

Programming of physical devices by direct manipulation seems quite natural, and an adequate visual representation of information may make direct-manipulation programming possible in other domains. Spreadsheet packages such as Excel™ have rich programming languages and allow users to create portions of programs by carrying out standard spreadsheet actions. The result of the actions is stored in another part of the spreadsheet and can be edited, printed, and stored in a textual form. Database programs such as Access™ allow users to create buttons that when activated will set off a series of actions and commands and even generate a report. Similarly, Adobe Photoshop records a history of user actions and then allows users to create programs with action sequences and repetition using direct manipulation.

It would be helpful if the computer could recognize repeated patterns reliably and create useful macros automatically while the user was engaged in performing a repetitive interface task. Most cellphones have buttons that can be programmed to call home or call the doctor or another emergency number. This allows the user to encounter a simpler interface and be shielded from the details of the tasks.

7.4 2-D and 3-D Interfaces

Some designers dream about building interfaces that approach the richness of 3-D reality. They believe that the closer the interfaces are to the real world, the easier usage will be. This extreme interpretation of direct manipulation is a dubious proposition, since user studies show that disorienting navigation, complex user actions, and annoying occlusions can slow performance in the real world as well as in 3-D interfaces (Cockburn and McKenzie, 2002). Many interfaces (sometimes called 2-D interfaces) are designed to be simpler than the real world by constraining movement, limiting interface actions, and ensuring visibility of interface objects. However, the strong utility of "pure" 3-D interfaces for medical, architectural, product design, and scientific visualization purposes means that they remain an important challenge for interface designers. So the power of 3-D interfaces lies in applying them in the appropriate domain or context where the added dimension provides more understanding and improves task outcomes.

An intriguing possibility is that "enhanced" interfaces may be better than 3-D reality. Enhanced features might enable outside of real human capabilities, such as faster-than-light teleportation, flying through objects, multiple simultaneous views of objects, and x-ray vision. Playful game designers and creative applications developers have already pushed the technology further than those who seek merely to mimic reality.

For some computer-based tasks—such as medical imagery (Fig. 7.8), architectural drawing, computer-assisted design, chemical-structure modeling (Fig. 7.2), and scientific simulations—pure 3-D representations are clearly helpful and have become major industries. However, even in these cases, the successes are often due to design features that make the interface better than reality.

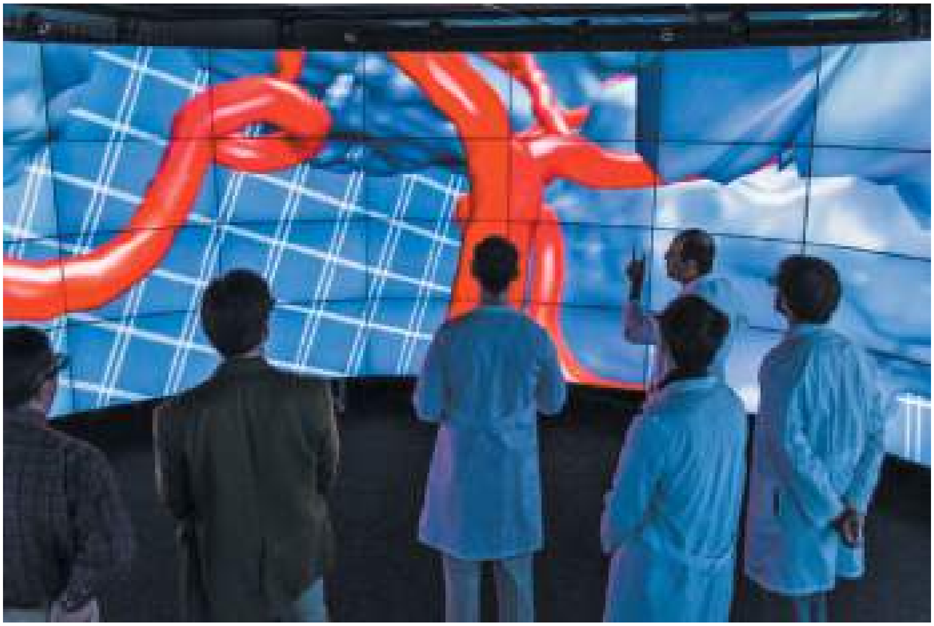


FIGURE 7.8
By using a medical simulation inserted into a large-scale visualization (using CAVE technology), physicians were able to find a solution that would not have been possible with the actual surgery. (http://www.nsf.gov/news/news_summ.jsp?cntn_id=126209)

Users can magically change colors or shapes, duplicate objects, shrink/expand objects, group/ungroup components, send them by various electronic means, and attach floating labels. Users can go back in time and even undo recent actions.

Among the many innovations, there have been questionable 3-D prototypes, such as for air-traffic control (showing altitude by perspective drawing only adds clutter when compared to an overview from directly above), digital libraries (showing books on shelves may be nice for browsing, but it inhibits searching and linking), and file directories (showing tree structures in three dimensions sometimes leads to designs that increase occlusion and navigation problems). Other questionable applications include ill-considered 3-D features for situations in which simple 2-D representations would do the job. For example, adding a third dimension to bar charts may slow users and mislead them (Hicks et al., 2003), but they are such an attraction for some users that they are included in most business graphics packages (Cognos, SAS/GRAPH, SPSS/SigmaPlot).

A modest use of 3-D techniques is to add highlights to 2-D interfaces, such as buttons that appear to be raised or depressed, windows that overlap and leave shadows, or icons that resemble real-world objects. These may be enjoyable, recognizable, and memorable because of improved use of spatial memory, but they can also be visually distracting and confusing because of additional visual complexity.

This enumeration of features for effective 3-D interfaces might serve as a checklist for designers, researchers, and educators:

- Use occlusion, shadows, perspective, and other 3-D techniques carefully.
- Minimize the number of navigation steps required for users to accomplish their tasks.
- Keep text readable (better rendering, good contrast with background, and no more than 30-degree tilt).
- Avoid unnecessary visual clutter, distraction, contrast shifts, and reflections.
- Simplify user movement (keep movements planar, avoid surprises like going through walls).
- Prevent errors (that is, create surgical tools that cut only where needed and chemistry kits that produce only realistic molecules and safe compounds).
- Simplify object movement (facilitate docking, follow predictable paths, limit rotation).
- Organize groups of items in aligned structures to allow rapid visual search.
- Enable users to construct visual groups to support spatial recall (placing items in corners or tinted areas).

Breakthroughs based on clever ideas seem possible. Enriching interfaces with stereo displays, haptic feedback, and 3-D sound may yet prove beneficial in more than specialized applications. Bigger payoffs are more likely to come sooner if these guidelines for inclusion of enhanced 3-D features are followed:

- Provide overviews so users can see the big picture (plan view display, aggregated views).
- Allow teleportation (rapid context shifts by selecting destination in an overview).
- Offer x-ray vision so users can see into or beyond objects.
- Provide history keeping (recording, undoing, replaying, editing).
- Permit rich user actions on objects (save, copy, annotate, share, send).
- Enable remote collaboration (synchronous, asynchronous).
- Give users control over explanatory text (pop-up, floating, or excentric labels and screen tips) and let them view details on demand.
- Offer tools to select, mark, and measure.
- Implement dynamic queries to rapidly filter out unneeded items.
- Support semantic zooming and movement (simple action brings object front and center and reveals more details).

- Enable landmarks to show themselves even at a distance.
- Allow multiple coordinated views (users can be in more than one place at a time and see data in more than one arrangement at a time).
- Develop novel 3-D icons to represent concepts that are more recognizable and memorable.

3-D environments are greatly appreciated by some users and are helpful for some tasks (Laha et al., 2012). They have the potential for novel social, scientific, and commercial applications if designers go beyond the goal of mimicking 3-D reality. Enhanced 3-D interfaces could be the key to making some kinds of 3-D teleconferencing, collaboration, teleoperation, and telepresence popular. Of course, it will take good design of 3-D interfaces (pure, constrained, or enhanced) and more research on finding the payoffs beyond the entertaining features that appeal to first-time users. Success will come to designers who provide compelling content, relevant features, appropriate entertainment, and novel social-media structure support. By studying user performance and measuring satisfaction, those designers will be able to polish their designs and refine guidelines for others to follow.

7.5 Teleoperation and Presence

Teleoperation has two parents: direct manipulation in personal computers and process control, where human operators control physical processes in complex environments. Typical tasks are operating power or chemical plants, controlling manufacturing, surgery, flying airplanes or drones, or steering vehicles. If the physical processes take place in a remote location, we talk about *teleoperation* or *remote control*. To perform the control task remotely, the human operator may interact with a computer, which may carry out some of the control tasks without any interference by the human operator.

There are great opportunities for the remote control or teleoperation of devices if acceptable user interfaces can be constructed. When designers can provide adequate feedback in sufficient time to permit effective decision making, attractive applications in manufacturing, medicine, military operations, and computer-supported collaborative work are viable. Home-automation applications extend remote operation of various devices to security and access systems, energy control, and operation of appliances. Scientific applications in space, underwater, or in hostile environments enable new research projects to be conducted economically and safely. The recent introduction of affordable drones will be yet another facet of teleoperation.

In traditional direct-manipulation interfaces, the objects and actions of interest are shown continuously; users generally point, click, or drag rather than type, and feedback indicating change is immediate. However, when the devices being operated are remote, these goals may not be realizable, and designers must expend additional effort to help users to cope with slower responses, incomplete feedback, increased likelihood of breakdowns, and more complex error-recovery procedures. The problems are strongly connected to the hardware, physical environment, network design, and task domain.

A typical remote application is *telemedicine*, or medical care delivered over communication links (Sonnenwald et al., 2014). Telemedicine can be used more broadly to allow physicians to examine patients remotely and surgeons to carry out operations across continents. Telehealth is being widely used in the Veteran's Administration (Fig. 7.9).

Veterans can come into the local VA office where technology visits with the various medical personnel can be conducted via Telehealth. Cameras with

FIGURE 7.9

Erica Taylor, Nurse Director for the Telehealth Program at Landstuhl Regional Medical Center, demonstrates using the Telehealth cart otoscope to conduct a real-time tympanic membrane exam. On the screen is Physician Assistant Steven Cain, who from a remote location can see and evaluate the patient and provide an appropriate plan of care. Photo by Phil Jones.

FIGURE 7.10

When doing robotic surgery, the surgeon sits at the computer console and controls the robotic camera and surgical instruments remotely. Various devices on the controller can be adjusted by the surgeon including adjustments/magnifiers to clearly see the field of view.

high-resolution images can allow the doctor to see the physical condition as well as the added benefit of seeing the affect of the patient. A trained medical person can be in the office with the patient to help facilitate the examination. Other medical applications include robotic surgery. *Robotic surgery* is an alternative to conventional surgery that enables a smaller incision and more accurate and precise surgical movements. The robotic platform expands the surgeon's capabilities and provides a highly magnified 3-D image (Fig. 7.10). In addition, the surgeon has control over hand, wrist, and finger movement through robotic instrument arms. The surgeon is comfortably seated across the operating room at a console rather than being over the patient, and the system damps out some involuntary movements that can be problematic.

The architecture of remote environments introduces several complicating factors:

- *Time delays*. The network hardware and software cause delays in sending user actions and receiving feedback: a *transmission delay*, or the time it takes for the command to reach the microscope (in our example, transmitting the command over the network), and an *operation delay*, or the time until the microscope responds. These delays in the system prevent the operator from knowing the current status of the system.

- *Incomplete feedback*. Devices originally designed for direct control may not have adequate sensors or status indicators. For instance, the microscope can transmit its current position, but it operates so slowly that it does not indicate the *exact* current position.
- *Unanticipated interferences*. Since the operated devices are remote, unanticipated interferences are more likely to occur than with physically present direct-manipulation environments. For instance, if a local operator accidentally moves the slide under the microscope, the positions indicated might not be correct. A breakdown might also occur during the execution of a remote operation without a good indication of this event being sent to the remote site.

One solution to these problems is to make explicit the network delays and breakdowns as part of the system. The user sees a model of the starting state of the system, the action that has been initiated, and the current state of the system as it carries out the action. It may be preferable for users to specify a destination (rather than a motion) and wait until the action is completed before readjusting the destination if necessary. Avenues for continuous feedback also are important.

Teleoperation is also commonly used by the military and by civilian space projects. Military applications for unmanned aircraft gained visibility during the recent wars in Afghanistan and Iraq. Reconnaissance drones and teleoperated missile-firing aircraft were widely used. Agile and flexible mobile robots exist for many hazardous duty situations (Murphy, 2014). Military missions and harsh environments, such as undersea and space exploration, are strong drivers for improved designs.

Telepresence was initially defined by Marvin Minsky (1980), but today the operative term is *presence*. The concept was that of not being remote but giving the feeling of "being there." Advances are being made with telepresence, and today's technologies and the internet-connected world have opened up additional possibilities. The commercial market is seeing a set of technologies called mobile remote presence (MRP) systems (Fig. 7.11). These are advancing video conferencing systems and allowing remote workers to have a feeling of presence. These devices facilitate formal communications as well as more informal chats in hallways. Some of the companies creating these devices include Suitable Technologies Beam, Mantarobot, Doublerobotics, and VGO. The controlling of these devices is another application of direct manipulation. Another application that extends the idea of video conferencing, made popular by Skype and other technologies, is a shared work space called ImmerseBoard, where the users are co-located but can work on the same screen (Fig. 7.12).

FIGURE 7.11
Three people having a conversation in a work environment, two are participating using MRP devices.

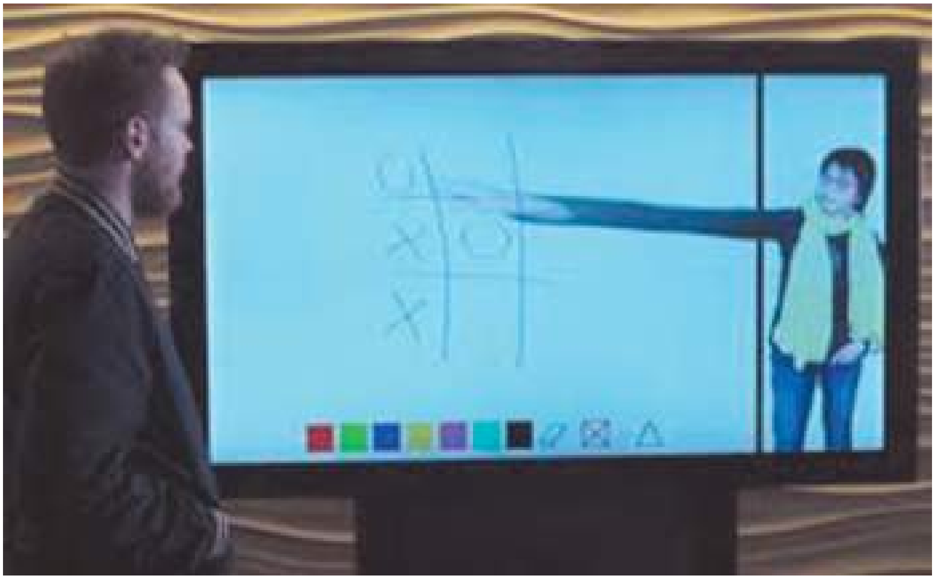


FIGURE 7.12
ImmerseBoard allows two users to be co-located and work on the same shared screen (Higuchi et al., 2015).

Robotics is a subfield of telepresence. Robots are being used in medical settings, office settings, education, and other specialized applications. New usage norms are being established for these types of devices and interactions (Lee and Takayama, 2011). The remote coworkers are often referred to as pilots. They can wander the hallways or "just hang out." Frameworks are being created with various design dimensions to better understand presence (Rae et al., 2015). It is important to understand the perspective of the users and especially that of the remote user. Doing various tasks with remote users in this type of set up can increase cognitive load. The remote person needs to concentrate on the task at hand as well as operating and positioning the device properly (Rae et al., 2014). Kristoffersson et al. (2013) provide an in-depth review of mobile robotic presence. Future work needs to be done on how mobility affects remote collaboration and on better understanding the design of mobility features. For individuals with limited mobility, robotics can facilitate more active participation. A full discussion of robotics and HCI is beyond the scope of this book.

7.6 Augmented and Virtual Reality

Flight-simulator designers work hard to create the most realistic experience for fighter and airline pilots. The cockpit displays and controls are taken from the same production line that creates the real ones. Then the windows are replaced by high-resolution computer displays, and sounds are choreographed to give the impression of engine start or reverse thrust. Finally, the vibration and tilting during climbing or turning are created by hydraulic jacks and intricate suspension systems. This elaborate technology may cost $100 million, but even so, it is a lot cheaper, safer, and more useful for training than the $400-million jet that it simulates. (And for training actual pilots, the reasonable flight simulators that millions of home computer game players have purchased won't quite do the trick!) Flying a plane is a complicated and specialized skill, but simulators are available for more common—and some surprising—tasks under the alluring name of *virtual reality* or the more descriptive *virtual environments*.

The gurus of virtuality are promoting immersive experiences. The miniaturization of electronics has provided less bulky gear to do that exploring. As computer systems continue to run faster, the obstacles that were in the way of immersive experiences are disappearing and the technology is becoming more affordable. Head-mounted displays are available from various manufacturers: Oculus Rift, Razer OSVR, HTC Vive, Sensics, Sony Glasses, and

FIGURE 7.13
Image-guided surgery can be done with the surgeon's hand attached to multiple sensors that can mimic the hand and finger positions and create accurate control. In the past, gloves were often used to attach the sensors and did not offer the flexibility and accuracy of the directly attached sensors. (http://polhemus.com/micro-sensors)

Polhemus. Bulky gloves are being replaced by more-lightweight materials (Fig. 7.13) and less-cumbersome connections (Fig. 7.14). Companies are advancing this technology very quickly. Magic Leap has just applied for a patent for a contact lens to facilitate augmented or virtual reality (Kokalitcheva, 2015).

The direct-manipulation principles outlined in Section 7.2.1 may be helpful to people who are designing and refining virtual and augmented reality environments. When users can select actions rapidly by pointing or gesturing

FIGURE 7.14
Oculus Rift head gear. This is an example of a virtual reality head-mounted display.

FIGURE 7.15

This figure shows the reality-virtuality continuum initially sketched by Milgram and Kishino in 1994. It still holds true today. Mixed reality is the reality that has some aspects of augmented reality within a virtual environment.

and display feedback occurs immediately, users have a strong sense of causality. Interface objects and actions should be simple so that users view and manipulate task-domain objects.

Graphics researchers have been perfecting image displays to simulate lighting effects, textured surfaces, reflections, and shadows. Data structures and algorithms for zooming in or panning across an object rapidly and smoothly are now practical on common computers and even some mobile devices. The immersive environment has some problems, including simulator sickness, nausea, and discomfort from wearing head-mounted gear and other equipment. Some of these problems are minimized by less-jumpy graphic transitions. Better understanding of the usability challenges, such as how much reality should be incorporated and when and how it can improve the user experience, is needed (McGill et al., 2015).

As our systems become more sophisticated, the distinction between different levels of virtuality blurs. It is best portrayed as originally conceived by Milgram and Kishino (1994): a continuum (Fig. 7.15). The last two sections of this chapter discuss augmented reality (Section 7.6.1) and then virtual reality (Section 7.6.2).

7.6.1 Augmented reality

Augmented reality enables users to see the real world with an overlay of additional information; for example, while users are looking at the walls of a building, their semitransparent eyeglasses may show the location of electrical wires and studwork. Medical applications, such as allowing surgeons or their assistants to look at patient while they see an overlay of a sonogram or other pertinent information to help locate a tumor, also seem compelling (Fig. 7.16). Augmented reality could show users how to repair equipment or guide visitors through cities (Fig. 7.17). Augmented reality strategies also enable users to

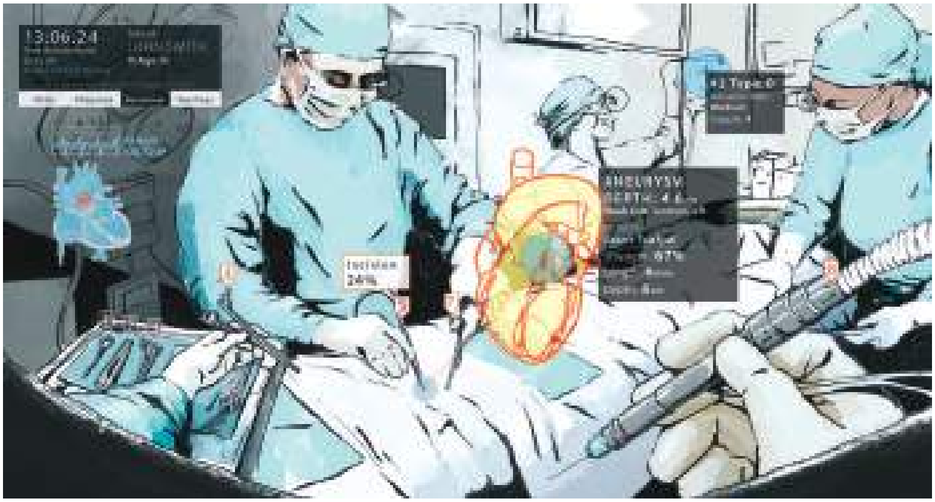


FIGURE 7.16
Virtual reality might be used to help surgeons or their assistants during surgery, by showing pertinent information superimposed on a view of the real world. (http://augmentarium.umiacs.umd.edu)

manipulate real-world artifacts to see results on graphical models (Poupyrev et al., 2002; Ishii, 2008) with applications such as manipulating protein molecules to understand the attractive/repulsive force fields between them. Using augmented reality systems to enhance social pretend play by young children (ages 4–6) promotes reasoning about emotional states as well as communication and divergent thinking (Bai et al., 2015).

An interior designer walking through a house with a client should be able to pick up a window-stretching tool or pull on a handle to try out a larger window or to use a room-painting tool to change the wall colors while leaving the windows and furniture untouched. Companies like IKEA are providing augmented reality tools so customers can visualize the products via their catalog in their own homes and rooms (Fig. 7.18).

7.6.2 Virtual reality

The presence aspect of virtual reality breaks the physical limitations of space and allows users to act as though they are somewhere else. Practical thinkers immediately grasp the connection to remote direct manipulation, remote control, and remote vision, but the fantasists see the potential to escape current

FIGURE 7.17

Using augmented reality overlays, the HERE City Lens app shows various points of interest on a mobile phone. Icons represent the types of places (food, shopping, etc.) and distances from the current location. In addition, links are provided to user reviews.

FIGURE 7.18

Customers can use their personal mobile devices to pull up objects from the IKEA catalog and see how the various items would look in their own house.

reality and to visit science-fiction worlds, cartoonlands, previous times in history, galaxies with different laws of physics, or unexplored emotional territories.

There have been many medical successes using virtual environments. For example, virtual worlds can be used to treat patients with a fear of heights by giving them an immersive experience with control over their viewpoint and movement. The safe immersive environment enables phobia sufferers to accommodate themselves to frightening stimuli in preparation for similar experiences in the real world. Another dramatic result is that immersive environments provide distractions for patients so that some forms of pain are controlled (Fig. 7.19). The immersive virtual reality environment has been used to treat military personnel suffering with PTSD (Fig. 7.20). Virtual worlds can be used for positive computing (Calvo and Peters, 2014) and wellness issues (Fig. 7.21).

FIGURE 7.19

A patient using UW HITLab/Harborvview's SnowWorld pain distraction at Shriners Children's Burn Center Galveston. UW designer/researcher Hunter Hoffman's latest version of SnowWorld was created for the UW by gifted worldbuilders at www .firsthand.com using www.3ds.com Virtual World Development Software. The immersive experience seems to lessen the painful experiences.

FIGURE 7.20
Soldiers can "re-live" portions of their combat experiences in a virtual reality setting with full immersion and sounds. Some systems even provide full immersion to include shaking and movement to make the experience as realistic as possible. Working with trained therapists, the soldier can be slowly desensitized from the traumatic experiences. (http://ict.usc.edu)

The opportunities for artistic expression and public-space installations are being explored by performance artists, museum designers, and building architects. Creative installations include projected images, 3-D sound, and sculptural components, sometimes combined with video cameras and user control by mobile devices. Other creative ideas include virtual dressing rooms where users can try on clothes on a model of themselves. The possibilities are truly endless.

Further information on virtual and augmented reality can be found in the wide assortment of textbooks available (Fuchs et al., 2011; Boellstorff et al., 2012; Kipper and Rampolla, 2012; Craig, 2013; Hale and Stanney, 2014; Barfield, 2015, Jerald, 2016). Billinghurst et al. (2014) recently compiled a comprehensive survey of augmented reality that gives both history of the field and details about the technologies and tools, including future research directions. The field is

FIGURE 7.21

Image of a virtual meditative world for engaging in meditation activities. The virtual world has sounds that change with each chakra (stage) of the meditation process. This is an application of positive computing. (http://nsuworks.nova.edu/gscis_etd/65/)

changing rapidly, and although avatars and virtual worlds still exist and are being explored (Blascovich and Bailenson, 2011), other virtual worlds like Second Life have almost disappeared (Boellstorff, 2008).

Practitioner's Summary

Among interactive systems that provide equivalent functionality and reliability, some systems have emerged to dominate the competition. Often, the most appealing systems have an enjoyable user interface with customized user-generated content that offers a natural representation of the task objects and actions—hence the term *direct manipulation* (Box 7.2). These interfaces are easy to learn, to use, and to retain over time. Novices can acquire a simple subset of the actions and then progress to more elaborate ones. Actions are rapid, incremental, and reversible, and they can be performed with physical movements instead of complex syntactic forms. The results of actions are visible immediately, and error messages are needed less often.

Using direct-manipulation principles in an interface does not ensure its success. A poor design, slow implementation, or inadequate functionality can undermine acceptance. For some applications, other approaches may be more appropriate. However, great potential exists for multiple and varied applications of direct-manipulation concepts. Compelling demonstrations of virtual and augmented reality are being applied in a growing set of applications with enhanced social interactions. Iterative design (Chapter 4) is especially important in testing advanced direct-manipulation systems because the novelty of these approaches may lead to unexpected problems for designers and users.

Researcher's Agenda

Research needs to refine our understanding of the contributions of each feature of direct manipulation: analogical representation, incremental action, reversibility, physical action instead of syntax, immediate visibility of results, characteristics such as translational distances, and graphic displays. Reversibility is easily accomplished by a generic undo action, but designing natural inverses for each action may be more attractive. Complex actions are well-represented with direct manipulation, but multi-layer design strategies for graceful evolution from novice to expert usage could be a major contribution. For expert users, direct-manipulation programming is still an opportunity, but good methods of history keeping and editing of action sequences are needed as well as increased attention to user-generated content. Better understanding of touchable interfaces and their uses as well as research on two-handed versus one-handed operations are needed. The allure of 3-D interaction is great, but researchers need to provide a better understanding of how and when (and when not) to use features such as occlusion, reduced navigation, and enhanced 3-D actions such as teleportation or x-ray vision and what are the best widths for field of view. Providing better semantic understanding of 3-D images can provide information for visually impaired users to better understand their environment. The impact of immersion on gaming and virtual worlds using rich social-media interactions across various ages and activities needs to be understood better.

Beyond the desktops and laptops, there is the allure of presence, virtual environments, augmented realities, and context-aware devices. Research is needed into how presence affects behaviors and interactions including privacy issues. The playful and enjoyable aspects will certainly be pursued, but the real challenge is to find the practical designs and a better understanding of "being there" when looking at 3-D worlds, both as individuals and as collaborators and players in the enriched social-media environments. A new set of tools is needed to investigate and better understand digital games research and its implications, both good and bad.

WORLD WIDE WEB RESOURCES

www.pearsonglobaleditions.com/shneiderman

Other Resources

Journals

- *Presence* (teleoperators and virtual environments): http://www.mitpressjournals.org/loi/pres
- *Virtual Reality*—Springer: http://www.springer.com/computer/image+processing/journal/10055
- *International Journal of Virtual Reality*: http://www.ijvr.org/
- *International Journal of Virtual Technology and Multimedia*: http://www.inderscience.com/jhome.php?jcode=ijvtm

Conferences

- VRST ACM Symposium on Virtual Reality Software and Technology: http://vrlab.buaa.edu.cn/vrst2015/
- IEEE Virtual Reality: http://ieeevr.org/2016/
- IEEE Symposium of Mixed and Augmented Reality, IEEE and ACM Symposium on Augmented Reality: http://ismar.vgtc.org

Additional information of this topic can be found in multimedia journals and conferences as well as journals and conferences that emphasize visualization.

Discussion Questions

1. Describe three principles of direct manipulation.
2. Give four benefits of direct manipulation. Also list four problems of direct manipulation.
3. Explain the differences between various kinds of direct manipulation with respect to translational distances.
4. An airline company is designing a new online reservation system. They want to add some direct-manipulation features. For example, they would like customers to click a map to specify the departure cities and the destinations, and to click on the calendar to indicate their schedules. From your point of view, list four benefits and four problems of the new idea compared with their old system, which required the customer to do the job by typing text.

5. Explain how virtual reality can be used for medical purposes.
6. List an example of teleoperation or virtual reality. Consider what a future application (that does not presently exist) might do. Be creative!

References

Bai, Zhen, Blackwell, Alan F., and Coulouris, George, Exploring expressive augmented reality: The fingAR puppet system for social pretend play, *Proceedings of the ACM Conference on Human Factors in Computing Systems*, ACM Press, New York (2015), 1035–1044.

Barfield, Woodrow (Editor), *Fundamentals of Wearable Computers and Augmented Reality*, 2nd Edition, CRC Press (2015).

Billinghurst, Mark, Clark, Adrian, and Lee, Gun, A survey of augmented reality, *Foundations and Trends in Human Computer Interaction 8*, 2–3 (2014), 73–272.

Blascovich, Jim, and Bailenson, Jeremy, *Infinite Reality: Avatars, Eternal Life, New Worlds, and the Dawn of the Virtual Revolution*, William Morrow (2011).

Boellstorff, Tom, *Coming of Age in Second Life: An Anthropologist Explores the Virtually Human*, Princeton University Press (2008).

Boellstorff, Tom, Nardi, Bonnie, Pearce, Celia, and Taylor, T. L., *Ethnography and Virtual Worlds: A Handbook of Method*, Princeton University Press (2012).

Calvo, Rafael, and Peters, Dorian, *Positive Computing: Technology for Wellbeing and Human Potential*, MIT Press (2014).

Cockburn, Andy, and McKenzie, Bruce, Evaluating the effectiveness of spatial memory in 2D and 3D physical and virtual environments, *Proceedings of the ACM Conference on Human Factors in Computing Systems*, ACM Press, New York (2002), 203–210.

Craig, Alan B., *Understanding Augmented Reality: Concepts and Applications*, Morgan Kaufmann (2013).

Csikszentmihalyi, Mihaly, *Flow: The Psychology of Optimal Experience*, Harper & Row (1990).

de Castell, Suzanne, Taylor, Nicholas, Jenson, Jennifer, and Weiler, Mark, Theoretical and methodological challenges (and opportunities) in virtual worlds research, *The International Conference on the Foundations of Digital Games 12* (2012), 134–140.

Few, Stephen, *Information Dashboard Design*, 2nd Edition, Analytics Press (2013).

Fisher, Kristie, Nichols, Tim, Isbister, Katherine, and Fuller, Tom, Quantifying "magic": Learnings from user research for creating good player experiences on Xbox Kinect, *International Journal of Gaming and Computer-Mediated Simulations 6*, 1 (January–March 2014), 26–40.

Fuchs, Phillippe, Moreau, Guillaume, and Guitton, Pascal, *Virtual Reality Concepts and Technologies*, CRC Press (2011).

Hale, Kelly S., and Stanney, Kay M. (Editors), *Handbook of Virtual Environments: Design, Implementation, and Applications*, 2nd Edition, CRC Press (2014).

Han, Jefferson Y., Low-cost multi-touch sensing through frustrated total internal reflection, *Proceedings UIST '05 Conference*, ACM Press, New York (2005), 115–118.

Hicks, Martin, O'Malley, Claire, Nichols, Sarah, and Anderson, Ben, Comparison of 2D and 3D representations for visualising telecommunication usage, *Behaviour & Information Technology 22*, 3 (2003), 185–201.

Higuchi, Keita, Chen, Yinpeng, Chou, Philip A., Zhang, Zhengyou, and Liu, Zicheng, ImmerseBoard: Immersive telepresent experience using a digital whiteboard, *Proceedings of the ACM Conference on Human Factors in Computing Systems*, ACM Press, New York (2015), 2383–2392.

Ishii, Hiroshi, Tangible user interfaces, in Sears, Andrew, and Jacko, Julie (Editors), *The Human-Computer Interaction Handbook*, 2nd Edition, Lawrence Erlbaum Associates, Hillsdale, NJ (2008), 469–487.

Jerald, Jason. *The VR Book: Human-Centered design for virtual reality*. Morgan & Claypool (2016).

Johnson, Daniel, Nacke, Lennart E., and Wyeth, Peter, All about that base: Differing player experiences in video game genres and the unique case of MOBA games, *Proceedings of the ACM Conference on Human Factors in Computing Systems*, ACM Press, New York (2015), 2265–2274.

Jones, Christian, Scholes, Laura, Johnson, Daniel, Katsikitis, Mary, and Carrs, Michelle C., Gaming well: Links between videogames and flourishing mental health, *Frontiers in Psychology 5* (March 2014), Article 260.

Kipper, Greg, and Rampolla, Joseph, *Augmented Reality: An Emerging Technologies Guide to AR*, Syngress (2012).

Kofman, Ava, Dueling realities, *The Atlantic* (June 9, 2015).

Kokalitcheva, Kia, Magic Leap files for a big pile of patents, including for a sci-fi contact lens, *Fortune* (September 1, 2015).

Kristoffersson, Annica, Coradeschi, Silvia, and Loutfi, Amy, A review of mobile robotic telepresence, *Advances in Human-Computer Interaction* (2013), Article 902316.

Kulshreshth, Arun, and LaViola, Joseph J. Jr., Exploring 3D user interface technologies for improving the gaming experience, *Proceedings of the ACM Conference on Human Factors in Computing Systems*, ACM Press, New York (2015), 125–134.

Kushner, David, Is it live, or is it VR? Virtual reality's moment, *IEEE Spectrum* (January 2014), 34–37.

Laha, B., Sensharma, K., Schiffbauer, J. D., and Bowman, D.A., Effects of immersion on visual analysis of volume data, *IEEE Transactions on Visualization and Computer Graphics 18*, 4 (April, 2012).

Lee, Min Kyung, and Takayama, Leila, "Now, I have a body": Uses and social norms for mobile remote presence in the workplace, *Proceedings of the ACM Conference on Factors in Computing Systems*, ACM Press, New York (2011), 33–42.

Lecky-Thompson, Guy W., *Video Game Design Revealed*, Charles River Media, Boston, MA (2008).

Lucero, Andres, Karapanos, Evangelos, Arrasvuori, Juha, and Horhonen, Hannu, Playful or gameful? Creating delightful user experiences, *ACM Interactions*, May–June (2014), 34–39.

McGill, Mark, Boland, Daniel, and Murray-Smith, Roderick, A dose of reality: Overcoming usability challenges in VR head-mounted displays, *Proceedings of the ACM Conference on Human Factors in Computing Systems*, ACM Press, New York (2015), 2143–2152.

McGonigal, Jane, *Reality Broken: Why Games Make Us Better and How They Can Change the World*, Penguin Press (2011).

Metz, Rachel, What's it like to try Magic Leap's take on virtual reality? *MIT Technology Review* (March/April 2015).

Milgram, P., and Kishino, F. A., Taxonomy of mixed reality visual displays, *IECE Transactions on Information and Systems (Special Issue on Networked Reality) E77-D*, 12, (1994), 1321–1329.

Mims, Christopher, Virtual reality isn't just about games, *The Wall Street Journal* (August 2, 2015).

Minsky, Marvin, Telepresence, *OMNI Magazine* (June 1980).

Murphy, Robin R., *Disaster Robotics*, MIT Press (2014).

Ossola, Alexandra, Could analyzing how humans think make better video games? *Popular Science* (February 17, 2015).

Pagulayan, Randy J., Keeker, Kevin, Fuller, Thomas, Wixon, Dennis, Romero, Ramon, and Gunn, Daniel V., User-centered design in games, in Jacko, Julie (Editor), *The Human-Computer Interaction Handbook*, 3rd Edition, Taylor and Francis/CRC Press (2012), Chapter 34, 795–824.

Parkin, Simon, *An Illustrated History of 151 Video Games,* Lorenz Books (2014).

Poupyrev, Ivan, Tan, Desney S., Billinghurst, Mark, Kato, Hirokazu, Regenbrecht, Holger, and Tetsutani, Nobuji, Developing a generic augmented-reality interface, *IEEE Computer 35*, 3 (March 2002), 44–50.

Rae, Irene, Mutlu, Bilge, and Takayama, Leila, Bodies in motion: Mobility, presence, and task awareness in telepresence, *Proceedings of the ACM Conference on Human Factors in Computing Systems*, ACM Press, New York (2014), 2153–2162.

Rae, Irene, Venolia, Gina, Tang, John C., and Molnar, David, A framework for understanding the designing telepresence, *Proceedings Computer Supported Cooperative Work and Social Computing (CSCW)* (2015), 1552–1566.

Shneiderman, Ben, Direct manipulation: A step beyond programming languages, *IEEE Computer 16,* 8 (August 1983), 57–69.

Sonnenwald, Diane H., Söderholm, Hanna Maurin, Welch, Gregory F., Cairns, Bruce A., and Fuchs, Henry, Illuminating collaboration in emergency health care situations: Paramedic-physician collaboration and 3D telepresence technology, *Information Research 19,* 2 (June 2014).

Stein, Joel, Inside the box, the surprising joy of virtual reality, *Time* (August 17, 2015).

Thomas, Bruce H., A survey of visual, mixed, and augmented reality gaming, *ACM Computers in Entertainment 10*, 3 (November 2012), Article 3.

Weiser, M., The computer for the 21st century, *Scientific American* 265, 3 (1991), 94–104.

Wellner, P., Mackay, W., and Gold, R., Computer augmented environments: Back to the real world, *Communications of the ACM 36*, 7 (July 1993), 24–27.


PLEASE

CHAPTER 8

Fluid Navigation

> A man is responsible for his choice and must accept the consequences, whatever they may be.
>
> **W. H. Auden**
> *A Certain World*, 1970

CHAPTER OUTLINE

8.1 Introduction

This chapter addresses design issues related to navigation, which can be defined as enabling users to know where they are and to steer themselves to their intended destination. In short, navigation is about getting work done or having fun through a series of actions, much like sailors who steer their boat to a harbor. Navigation is key to successfully operating interactive applications, such as installing a mobile app, filling in a survey, or purchasing a train ticket (task navigation). It is also the key to finding information on a website or browsing social media (web navigation) or to finding the action needed in a desktop application (command menu navigation).

Navigation harnesses users' ability to rapidly skim choices, recognize what is relevant, and select what they need to realize their intentions. The goal for designers is to enable fluid navigation that allows users to gracefully and confidently get to where they want to go, explore novel possible routes, and backtrack when necessary. Navigation depends on recognition of landmarks that travelers use to guide their choices, which differs greatly from search, which requires users to describe what they want by typing keywords in a blank search box (see Chapter 15).

While the search box is the main technique to initiate the process of finding information in vast information spaces (like the internet or digital libraries), navigation techniques such as small or large menus, embedded links, or tool palettes are the workhorses of navigation. Users indicate their choices with a touch, tap, or swipe of the finger or by using a pointing device (see Chapters 7 and 10) and get immediate feedback indicating what they have done. Navigation by selection is an interaction style that is especially effective for users who are novice or first-time users, are knowledgeable intermittent users, or need help in structuring their decision-making processes. However, with careful design of complex menus and rapid interaction, menu selection can be appealing even to expert frequent users. These strategies can be used in combination with command languages (see Section 9.5), allowing users to transition smoothly from novice to expert because menus offer cues to elicit recognition rather than forcing users to recall the syntax of a command from memory. Careful design, keyboard shortcuts, and gestures allow expert users to navigate quickly through large information structures.

A loose definition of menus is used here as a representation of available choices, which can describe the rich array of techniques designers use to present choices and guide users as they select what they want. Arrays of check boxes or form fill-in can be seen as primarily data-entry techniques, but those techniques contribute to the experience of steering an application or website navigation (e.g., to complete a survey, sign up for a service, or make a purchase), so they are discussed in this chapter as well. Similarly, dialog boxes contribute to allowing users to express their choices, so dialog box design is described at the end of the chapter.

Very early studies demonstrated the importance of organizing menus in a meaningful structure, resulting in faster selection time and higher user satisfaction (see Section 8.4). Navigation may follow a linear sequence (e.g., in a wizard or survey), a hierarchical structure that is natural and comprehensible (e.g., an ebook split into chapters, a store into departments, or the animal kingdom into species), or a network structure when choices may be reachable by more than one path (e.g., websites).

By harnessing the latest versions of HTML or CSS, even webpages and mobile applications now include smooth animations and sleek graphic design that turn basic menus into custom widgets that help define the entire look and feel of a website or application. When links and menus or choices and commands are designed using familiar terminology or recognizable visual elements and are organized in a meaningful structure and sequence, users can navigate complex information structures easily with a few mouse clicks or taps of the finger or smoothly scroll through sleek presentations of the possible next steps to accomplish their tasks. Carefully selected gestures can add a sense of delight and fluidness to the navigation on touchscreen devices.

Of course, just because a designer uses slick graphical menus, elegant form fill-in, or well-known gestures does not guarantee that the interface will be appealing and easy to use. Effective interfaces emerge only after careful consideration of and testing for numerous design issues, such as task-related organization, phrasing of items, sequence of items, graphic layout and design, responsive design to adapt to various sizes of devices, shortcuts for knowledgeable frequent users, online help, and error correction (Bailly et al., 2015).

This chapter starts by reviewing the rich array of available techniques for allowing users to specify their choices, from single techniques to the combinations of multiple techniques (Section 8.2). Section 8.3 discusses issues related to small displays. Content organization is discussed in Section 8.4. Finally, Section 8.5 discusses the needs of audio menus, and form fill-in and dialog boxes are covered in Section 8.6.

See also:

- Chapter 10, Devices
- Chapter 12, Advancing the User Experience
- Chapter 14, Documentation and User Support (a.k.a. Help)
- Chapter 16, Data Visualization

8.2 Navigation by Selection

Choices can be presented explicitly, in that there is an orderly enumeration of the items with little extraneous information, or they can be embedded in text or graphics and still be selectable. Embedded links of webpages were first popularized in the Hyperties system (Koved and Shneiderman, 1986), which was used for early commercial hypertext projects and became the inspiration for the *hotlinks* of the World Wide Web. Highlighted names, places, or phrases became menu items embedded in text that informs users and helps to clarify the meaning of the menu items. Graphical techniques are a particularly attractive way to present choices while providing context to help users specify what they want. For example, maps can orient users about the geography of the area before users select an item of interest, and calendars or timelines can inform users of availability and constraints before a date or time is selected (e.g., see HIPMUNK in Fig. 1.7). Interactive visualization of information can also help analysts navigate large amount of data in a fluid visual manner (Elmqvist et al., 2011 and Chapter 16).

The simplest case of explicit menus is a *binary menu* for yes/no, true/false choices (Fig. 8.1).

Another example of a simple menu is the grid menu popularized by mobile devices, with a small set of icons and labels (Fig. 8.2).

When users need to make a series of choices (e.g., in a survey or to select parameters of an application), there are well-established methods of presenting choices.

Radio buttons support single-item selection from a multiple-item menu (Fig. 8.3), while *check boxes* allow the selection of one or more items in a menu. A multiple-selection menu is a convenient method for handling multiple binary

FIGURE 8.1

A simple menu with two choices. A short explanation is provided. Buttons are large enough to be easy to select and have informative labels, and one answer has been highlighted as the most likely answer.

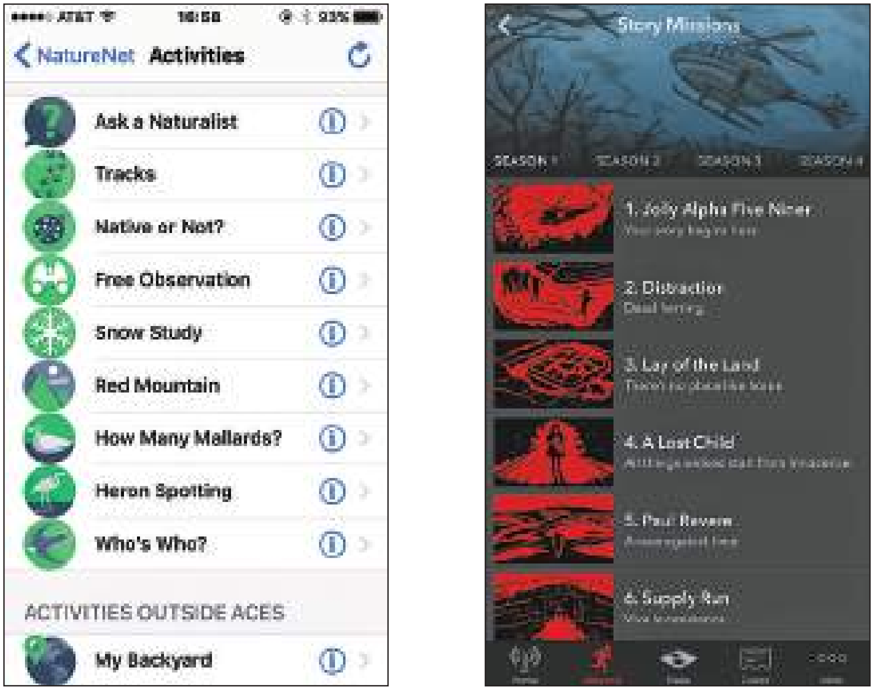


FIGURE 8.2

Two examples of simple menus. On the left, the NatureNet citizen science app shows the nine functions of the main menu. On the right, the Zoombies, Run! app lists the possible missions of Season 1 of the immersive running game and audio adventure.

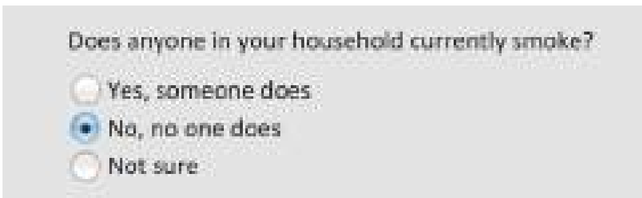


FIGURE 8.3

Three radio buttons constitute a menu that steers users to appropriate information in a health risk assessment website.

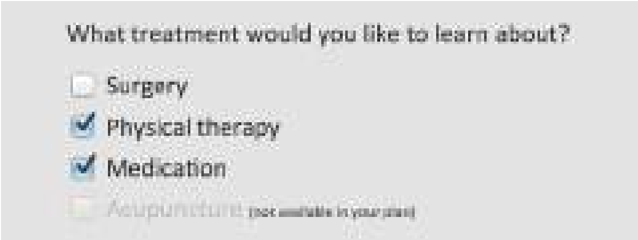


FIGURE 8.4

Check boxes allow users to indicate their preferences about treatment they would like to discuss. Feedback is provided by a check mark. Unavailable choices can be grayed out.

choices, since the user is able to scan the full list of items while deciding (Fig. 8.4). Unavailable choices can be grayed out.

8.2.1 Menu bars, pop-up menus, toolbars, palettes, and ribbons

Menu bars are typically found at the top of the each application (Fig. 8.5 but also Fig. 1.2 or 1.4) or both at the top and on the side of the screen. Common items in desktop or tablet applications are File, Edit, View, and Help, and menus that follow this order will seem familiar to most users. Clicking on a menu title brings up a list of related items, and users can then make a selection by moving the pointing device over the items (which respond by highlighting) and clicking on the desired choice. Since positional constancy is such a strong principle, when an item is not available for selection, it is important to gray it out rather than removing it from the list (e.g., "Copy to Scrapbook" in Fig. 8.5).

The increasing ease of creating custom widgets allows designers to create endless variants of the original menu bars. Preserving readability and ensuring that users will be able to identify menus as such are important goals when creating these new designs. Many rely on multiple menu bars, placing menus at the top but also on the side and bottom of the screen or webpage. When placed on the side, submenus can open in place using an *accordion menu* style expansion, or to the side. Accordion menus work well when the submenus have few items and do not force users to scroll too far to collapse the accordion, but accordions may also increase user disorientation when the indenting scheme is unclear or the menu structure is more than two or three levels deep. Large submenus are better expanded below or to the side (e.g., the REI

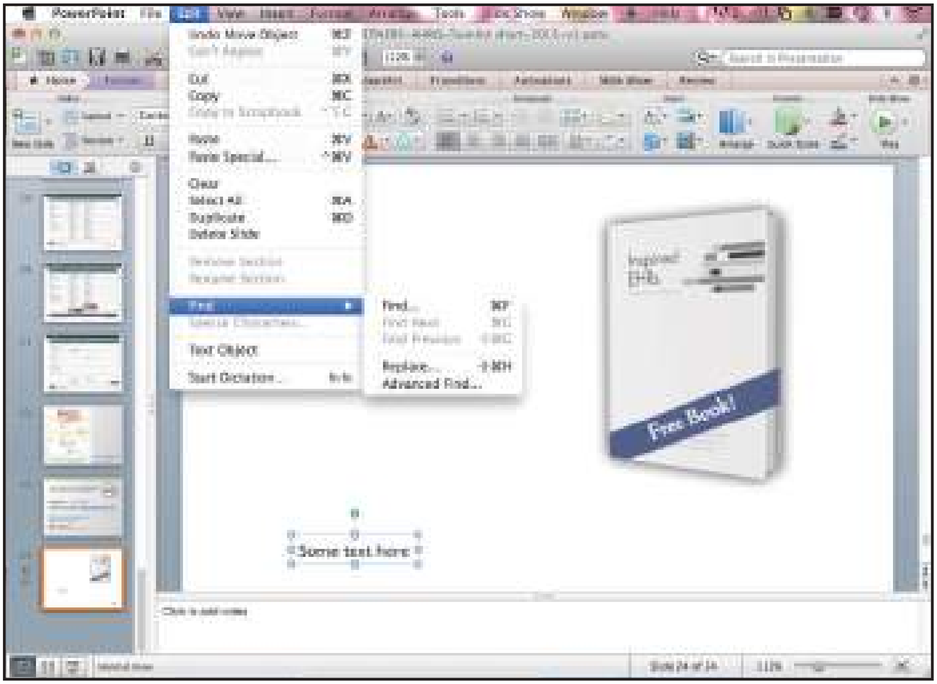


FIGURE 8.5

On the top menu bar of Microsoft PowerPoint, the Edit cascading *pull-down menu* (also called *pulled-right*) is open, followed by the Find menu. The menus allow users to explore the functions of the application. To facilitate discovery and learning, icons and keyboard shortcuts are indicated on the right of the menu items (for example, ⌘C for Copy or ⌘F for Find). A small black triangle indicates that selection of the menu item will lead to a submenu. Three dots (...) indicate that the selection will lead to a dialog box. Partially hidden behind the Edit menu, the application *ribbon* is visible, revealing the large number of choices available in the selected tab (Format).

website lists all the Cycle subcategories in a large menu that expands to the right, filling most of the screen; see Fig. 8.6).

The limited screen space of mobile devices leads designers to strive to limit the number of menu items. To leave more room for content, most or all menu items can be moved into a separate screen that is accessible from a main menu icon, sometimes called the *hamburger menu* icon  for its shape and which can be placed on every screen (Fig. 8.7).

Toolbars, *iconic menus*, and *palettes* can offer many actions that users can select with a click and apply to a displayed object (Fig. 1.10). A large number of toolbars can be overwhelming, so users need to be able to customize which toolbars

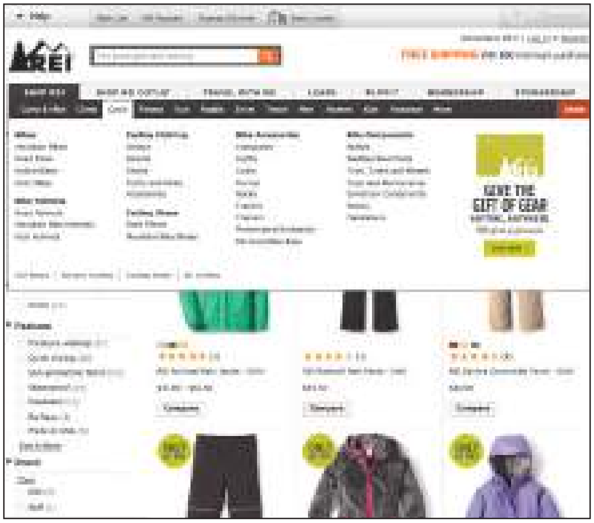
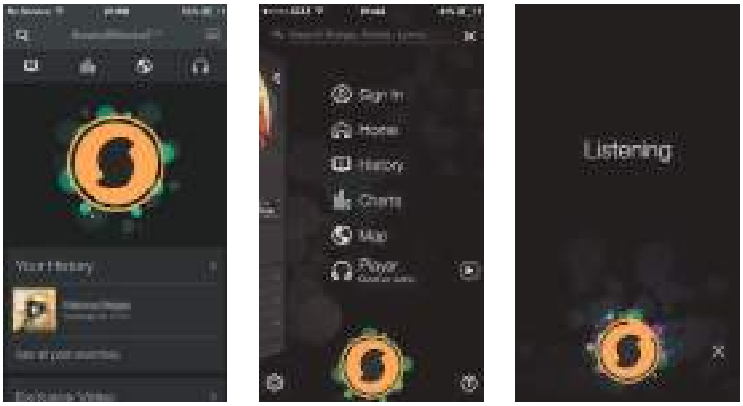


FIGURE 8.6

In the REI website, the categories for “Cycle” are expanded all at once below the top menu, showing 34 items organized in a meaningful hierarchy as a large menu.



A B C

FIGURE 8.7

The main menu of Soundhound has only six items, but it is still too much to be displayed on every page, so a main menu “hamburger” icon appears at the top right of all appropriate pages; for example, it appears in A but not in the recording screen C, where only the X close icon is visible. The main menu (B) animates from the right, so most users will learn that a swipe to the left also opens the main menu and a swipe to the right will close it. In B, part of the previous screen is still visible on the left, reinforcing the suggestion that swiping can be used.

are visible and to control the placement of those toolbars. Tool palettes (such as color wheels or layers) may be separated from the menus and moved so they do not hide the content. Users who wish to conserve screen space can eliminate most or all of the toolbars and palettes. Dense menus with many small icons can be overwhelming for novice users but appreciated by experts because of their small footprint and quick access.

Pop-up menus appear on the display in response to a click or tap with a pointing device. When the content of the pop-up menu depends on the cursor position, it is called a *context menu*. Since the pop-up menu covers a portion of the display, there is strong motivation to keep the menu text short (so that it does not cover the context of the menu). Pop-up menus can be hard to discover, so alternative access may need to be provided. Pop-up menus can also be organized in a circle to form *pie menus* also called *marking menus* (Figs. 2.5 and 8.8). Those menus have the advantage that the average distance to travel to select an item is smaller than linear menus, and with practice they can be used without visual attention if users memorize the direction of the item (which is easier with four to eight items). This is particularly useful in design applications that require constant menu selections (Fig. 8.8).

Ribbons were introduced by Microsoft in Office 2007. Ribbons attempt to replace menus and toolbars by one-inch tabs grouping commands by task (Fig. 8.5). While this approach might be beneficial for new users, expert users had difficulties adapting to the reorganized menus and finding items they knew existed before, highlighting the challenge of versioning and menu reorganization in professional applications. Ribbons also reduce the screen space for the document, which is a drawback for many users.

8.2.2 Shortcuts and gestures for rapid interaction

For rapid selection, *keyboard shortcuts* (also sometime called *hotkeys*, such as Ctrl-C on PCs or ⌘-C on Macs for Copy) are essential for expert users using desktop computers (Fig. 8.5). Users can memorize the keystrokes for the menu items they use often and thus speed up the interaction considerably. The first letter of the command is often used for the shortcut to favor memorability, but caution is required to avoid collisions. If at all possible, shortcuts should be used consistently across applications; for example, Ctrl-S on a PC or ⌘-S on a Mac is usually used for Save and Ctrl-P or ⌘-P for Print. Keyboard shortcuts should be indicated next to their corresponding menu items and in the tooltip of the menu icons. Learning shortcuts is one of the useful paths to reaching expert performance (Cockburn et al., 2014), but many users never even attempt to learn them. Using a modifier key to reveal all the available shortcuts at once was found to be helpful to increase their use (Malacria, 2013).

Since typing keyboard shortcuts become impractical or impossible with touchscreen devices, other techniques are being devised for smart phones and

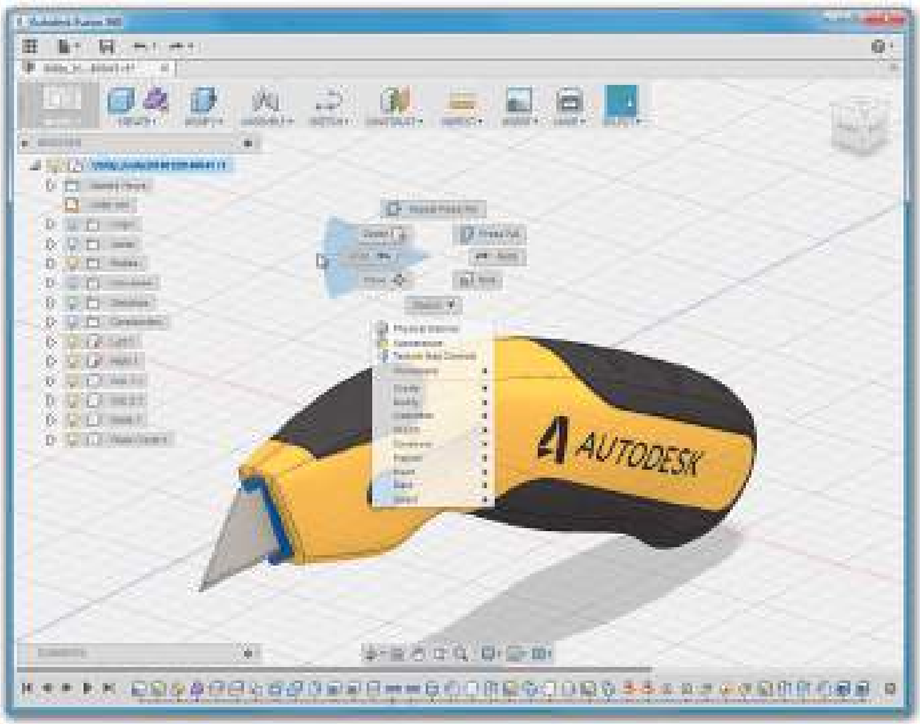


FIGURE 8.8

Fusion 360TM, an AutodeskTM 3D Computer Aided Design tool, allowed an engineer to design a utility knife. A click on the background of the image brings a pop-up marking menu with eight context-dependent menu items arranged in a circle (as well as a conventional linear menu below it). Sliding the mouse to the left selects the Undo command, now highlighted by a pie-shaped gray background. When the click + move is done rapidly, the menu itself doesn't appear on the screen, allowing rapid command selection via simple gestures (http://www.autodesk.com/products/fusion-360).

tablets. Gestures often serve as a shortcut for rapid selection (Box 8.1). First made widely available by the Apple iPhone, gestures have transformed navigation with tablets and smart phones. Still, they can be hard to discover and learn and have few or no affordances. Redundancy is recommended, i.e., an alternate traditional way of selecting the action may be needed instead of relying solely on gestures (see Fig. 8.7). Careful design and use of gestures (Wigdor and Wixon, 2011; Zhai et al., 2012) can lead to fluid navigation for expert users but cause frustration when actions are triggered inadvertently. Newer approaches take advantage of the multi-touch capabilities of touchscreens; for example, a two-finger swipe to the right might be associated with the back button of a browser.

BOX 8.1

Examples of Common Gestures and Their Effects

Gestures can speed interaction, and their directness is compelling, but they are hard to discover. Gestures may have different actions when applied on an object, on the background, or toward the edge of the screen, which can be frustrating when applied inadvertently; therefore, it is important to ensure easy reversal of actions. Consistent application of gestures remains an issue.

- *Tap*: select
- *Long press:* varied, from magnified cursor (iOS) to showing a tooltip (Windows 8)
- *Double tap*: varied (e.g., zoom [iOS])
- *Small swipe*: varied (e.g., move location or order of objects, reveal a delete button)
- *Large swipe*: usually scroll
- *Rapid swipe or fling*: fast scroll with inertia
- *Pinch and spread*: zoom in and out
- *Variation with two or more fingers*: varied effects

FastTap allows users to select commands by combining a thumb tap (to display the menu) and an index finger tap to select (Gutwin et al., 2014). As users learn the location of menu items relative to their thumb, they can select rapidly before the menu is even displayed. Allowing users to customize the gestures may help users remember them and provide better accessibility than pre-defined gestures, but users have limited understanding of the recognizer's ability to recognize gestures they propose, often leading to poorly recognized gestures (Oh and Findlater, 2013).

Other aspects of design contribute to rapid navigation, such as error prevention, avoiding scrolling, and laying out menus on the screen such that the distance traveled to perform the most common tasks is minimized (see Chapters 12 and 13).

8.2.3 Long lists

Sometimes the list of menu items may be longer than the 30 to 40 lines that can reasonably fit on a display. One common solution is to create a tree-structured menu (Section 8.4.1), but sometimes the desire to limit the interface to one conceptual menu is strong—for example, when users must select a state from the 50 states in the United States or a country from an extensive list of possibilities. Typical lists are alphabetically ordered, but categorical lists may be useful. The principles of menu-list sequencing apply (Section 8.4.2).

Scrolling menus, combo boxes, and fisheye menus *Scrolling menus* display the first portion of the menu and an additional menu item, typically an arrow that leads to the next set of items in the menu sequence. The scrolling (or paging) menu might continue with dozens or thousands of items. Allowing users to type the letter "*M*" to scroll directly to the first word starting with the letter "*M*" will reduce manual scrolling, but this feature is not always discovered. Similarly, typing *M* twice can move to the second word starting with "*M*". *Combo boxes* make this option more evident by combining a scrolling menu with a text-entry field.

Users can type in leading characters to scroll quickly through the list. Another alternative is the *fisheye menu*, which displays all of the menu items on the screen at once but shows only items near the cursor at full size; items further away are displayed at a smaller size. Fisheye menus have been made popular by Apple's Mac OS X (Fig. 1.2) and are attractive for menus of 10 to 20 items where the zoom ratio remains small and all items are readable at all times. When the number of items is such that smaller items become unreadable, fisheye menus have the potential to improve speed over scrolling menus, but hierarchical menus remain faster (Hornbæk and Hertzum, 2007). Fisheye menus can be an eye-catching option but are not recommended as a default menu style for long lists.

Sliders and alphasliders When the available choices are continuous numerical values, a slider is a natural choice to allow the selection of a single value. Ranges of values can also be selected with double-sided (range) sliders. Users select values by using a finger or pointing device to drag the slider thumb (scroll box) along the scale (see Fig. 1.7). When greater precision is needed, the slider thumb can be adjusted incrementally by clicking on arrows located at each end of the slider. A similar technique that allows users to select a name or category among even large numbers of ordered items is an *alphaslider* (see Fig. 8.9). Because of their compactness, sliders, range sliders, and alphasliders are often used in the control panels of interactive visualization systems (Chapter 16). When results are available in real time, a sweep of the slider thumb allows rapid comparisons between the results of dozens of choices within seconds (without having to even look at the slider). This would be very tedious with a standard menu that requires users to start the selection process again for each new value.

Two-dimensional mega menus Alternatively, menus that fill all the available space might be used. Two-dimensional mega menus give users a good overview of the choices, reduce the number of required actions, and allow rapid selection. The ease of scrolling on touchscreens has encouraged designers to make heavy use of *scrollable two-dimensional menus* in webpage design (e.g., http://www.pinterest.com or the NASA website; see Fig. 8.10). Website competitions

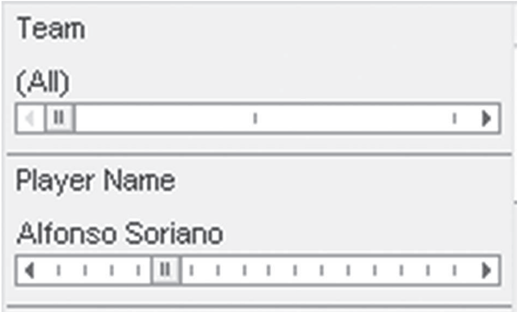


FIGURE 8.9

An alphaslider (also called an item slider) in the Spotfire visualization tool from Tibco. The alphaslider allows users to select one item from a large number of categorical items and rapidly step through the other items (http://spotfire.tibco.com).

FIGURE 8.10

The NASA website consists of a large scrollable 2-dimensional menu. Below the main menu, each square or rectangle is a large button. Scrolling gives access to dozens of items easily updated and rearranged. This adaptive grid design scales down nicely to the small displays. On the right, the same page is displayed on an Android phone. The grid now appears as a single column of items.

(e.g., http://www.awwward.com or http://www.webbyawards.com/) gave the 2015 awards to sites with home pages filled with bright photos and snazzy graphic interspersed with selectable objects. The top section of the webpage that is visible at first (called the *topfold* or *above the fold*) remains critical as users need the confidence that the site answers their needs before they start to scroll down (why go further if not impressed?), but users can be exposed to potentially hundreds of selectable zones or menu items within seconds of scrolling, which remains entirely in users' control.

In stark contrast, some designers choose the more sober style of a text-only large 2-dimensional menu (e.g., craigslist in Fig. 8.11). Compact text menus allow users to rapidly scan hundreds of choices without dizzying effects or reorientation. This utilitarian solution is appealing for websites with little or no competition (e.g., company intranets) or home pages of sites whose success comes entirely from direct access to their lower-level pages through search

FIGURE 8.11

The craigslist home page is a text-only, 2-dimensional mega menu. It allows users to rapidly read hundreds of choices with little or no scrolling required. Items are organized hierarchically. (http://www.craigslist.org/)

engines. Similarly, a *site map* lists every single page of a website and is useful as a table of contents.

With such compact text-oriented designs—as well as with all other more graphic-oriented designs—accessibility issues need to be addressed (Fig. 2.1).

Users browsing user-generated content such as photo or document collections also need to choose among non-curated lists of terms or tags attached to items in the collections. *Tag clouds* were fashionable until recently as compact 2-dimensional text menus. In tag clouds, the larger the font size of the tag, the more items are available. While attractive and fun, tag clouds are often misinterpreted because longer tags have more prominence than short ones and users believe that the position in the tag cloud has meaning even when it does not. To address this problem, tag indexes are now gaining popularity with tags sorted by number of items so users make no mistakes when looking for the tags that have the most items (Fig. 8.12). A horizontal layout may be convenient when the list is long, but arranging the tags vertically will facilitate scanning of the list.

8.2.4 Linear versus simultaneous presentation

Often, a sequence of interdependent menus can be used to guide users through a series of choices. For example, a pizza-ordering interface might include a linear sequence of menus in which users choose the size (small, medium, or large), thickness (thick, normal, or thin crust), and finally toppings. Other familiar examples are online examinations that have sequences of multiple-choice test items, each made up as a menu, or *wizards* (a Microsoft term) that steer users through software installation by presenting a sequence of menu options. Linear sequences guide users by presenting one decision at a time and are effective for

FIGURE 8.12

Awwwards.com gives awards to a large number of websites, which are tagged. A tag index at the top of the page displays all the tags sorted by total count. The counts are indicated in parenthesis. The green-colored tags are the popular tags that have been selected more often (which most likely will lead to even more selection).

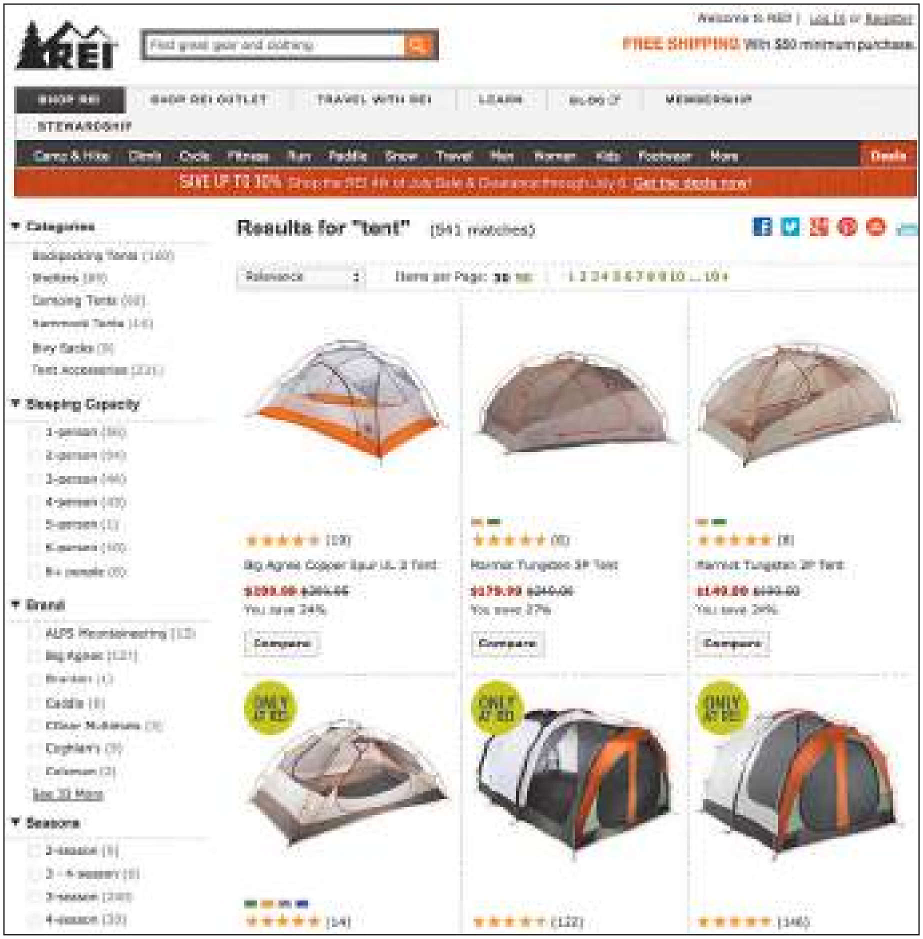


FIGURE 8.13

Faceted search interfaces allow shoppers looking for tents to narrow the list of results by indicating their choices in the simultaneous menus on the left: categories, sleeping capacity, brand, seasons, and so on. Results can be laid out in a row or a grid, and sorting can be done by price or rating. (http://www.rei.com/)

novice users performing simple tasks. They may be the only possible option for a small display.

Simultaneous menus present multiple active menus (also called filters) on a screen at the same time and allow users to enter choices in any order. They require more display space; however, experienced users performing complex tasks benefit from simultaneous menus. *Faceted search menus* are a very powerful application of simultaneous menus now used extensively in online shopping, library catalogs, and other database searches (see Fig. 8.13 and Chapter 15).

8.3 Small Displays

While most designs adapt fairly easily from desktop displays to the larger tablets (once the design has been reviewed for touchability), small displays make most desktop designs impractical, and dumbed-down designs are very likely to fail. Small displays require a radical rethinking of what functionalities should be included and often lead to novel interface and menu designs specifically adapted to particular devices and applications.

The smaller the screen, the more temporal the interface becomes (all the way to entirely linear audio interfaces when no display is available). For example, linear sequences of menus are possible, while simultaneous menus are much harder to fit in. On tiny devices (such as watches or fitness wearables), a *deck of card menu* can be used, where each single tap advances to the next choice and a long press or two-finger press may select the item to access more information. Animated attention-catching *ticker menus* have also been used. Users don't need to manually scroll or page through the menu items, and with a single touch they can stop the scrolling and select an item in view. On the other hand, having to wait for an item to appear or reappear will be frustrating to some users, especially as the number of items grows.

Temptation is great to include menu items just because they fit, but successful designs limit the number of functions to the most essential ones (Box 8.2 and Fig. 8.14). They may push other features in less accessible parts of the interface, relegate them to counterpart applications on desktop or tablet, or eliminate the features altogether. An often-mentioned rule of thumb for small devices is "less is more."

BOX 8.2

Design considerations for small displays.

- Simplify: Less is more.
- Strive to reduce or eliminate data entry.
- Learnability is key.
- Consider use frequency and importance.
- Plan for interruptions.
- Use of contextual information.
- Make clear what is selectable and what is not.
- Leave room for scroll and swipe gestures to avoid inadvertent actions.
- Consider relegating less important functions to other platforms.

FIGURE 8.14

Small devices have very focused functionalities and few selectable areas. Discoverability is often an issue.

Apps need to be learned in a few seconds or risk being abandoned. Sequencing menu items by frequency of use can be more useful than sequencing by category or alphabetical order, as speed of access to the most commonly used options is critical. For example, it is likely that flight status and check-in are more common than booking a flight on a mobile device. This can be verified by logging usage data.

Designers also need to allow users to deal with interruptions and distractions in their environment; for example providing an automatic Save function addresses interruption issues (e.g., when the phone rings) and simplifies the interface.

Concise writing and careful editing of titles, labels, and instructions will lead to simpler and easier-to-use interfaces. Every word counts on a small screen, and even unnecessary letters or spaces should be eliminated. Consistency remains important, but clear differentiation of menu types helps users remain oriented when no context can be provided. Tiny icons are difficult to design and are rarely used, as they take up space and require labels anyway. On the other hand, large color icons, such as those used in car navigation systems or the main screen of most smart phones, can be used successfully because they can be recognized at a glance once they have been learned.

Data entry is a difficult challenge for small devices and should be avoided as much as possible. The use of contextual information such as location (e.g., with a global positioning system [GPS]) or the proximity to objects (e.g., using radio-frequency identification [RFID] tags or scanned QR codes) complemented with simple touch widgets may facilitate the navigation to relevant information. For example, using the current location as default when searching for a hotel using a smart phone will eliminate data entry in many situations. Making all phone numbers and e-mail addresses selectable for easy calls or e-mails, addresses loadable in maps, and dates a tap away from calendars can dramatically shorten navigation time. In certain cases, hand-off to another larger device may be the best approach (e.g., the login and password function on a watch app can be handed off to be executed on a nearby phone or laptop.)

Position information relative to the user's body can also lead to new modes of interaction with menus of small devices. For example, users can move the device in front of them horizontally or vertically to scroll through long lists or pan across maps. Using the back of the device as a touch-sensitive pad might help enrich selection mechanisms.

Responsive menus that adapt to different screen sizes remain a challenge. Less important functions can be removed or relegated to other platforms (e.g., deleting names in a directory). Different stylings may allow more buttons to fit in small spaces, but they need to be large enough to allow easy selection on touch-screen devices. Menu labels can be carefully abbreviated or replaced by icons only. Menus can appear in a different location or be bundled on a separate screen (e.g., using the hamburger menu icon (Fig. 8.7)). One successful strategy is to design for mobile first instead of dumbing down the design for larger displays (Wroblewski, 2011).

Designing for older *feature phones* will open the door to a wider audience—for example, in emerging markets (Medhi et al., 2013, Fig. 2.1). Such phones typically have dedicated hard buttons to control the connect and disconnect functions and up and down buttons to navigate lists as well as "soft" keys with matching on-screen labels that change dynamically depending on the context. Soft keys are extremely useful as they allow designers to provide direct access to the next-most-logical command at every step. Consistent placement of the commands will speed interaction—for example, user selections on the left side and back or exit options on the right side.

8.4 Content Organization

Meaningful grouping and sequencing of choices, along with careful editing of titles and labels and appropriate layout design, can lead to easier-to-learn menus and increased navigation speed. In this section, This section reviews the content-organization issues and provides guidelines for design. This area of design has been heavily researched in the context of traditional menus for desktop applications, but most results are useful for website and phone application designs (Krug, 2014). Webpages act as large menus where items are the embedded links or buttons that can be used to navigate to another page.

Some lessons can be learned from restaurant menus. Restaurant menus separate appetizers, main dishes, desserts, and beverages to help customers organize their selections. Menu items should fit logically into categories and have readily understandable meanings. Restaurateurs who list dishes with idiosyncratic names such as "veal Amélie", unfamiliar labels such as "wor shu op", or vague terms such as "house dressing" should expect that customers will be puzzled or anxious and that waiters will waste time providing explanations. Similarly, for computer menus, the categories should be comprehensible and distinctive so that users are confident in making their selections and have a clear idea of what the result will be. Computer menus become more difficult to design than restaurant menus when the number of choices and the level of complexity increase—and there are no waiters to turn to for help.

8.4.1 Structure and breath versus depth

When a collection of items grows, designers can form categories of similar items, creating a *tree structure* (Box 8.3). Some collections can be partitioned easily into mutually exclusive groups with distinctive identifiers. For example, the products in an online grocery store can be organized into categories such as produce, meat, dairy, cleaning products, and so on. Produce can then be organized into vegetables, fruits, and nuts, while dairy is organized into milk, cheese, yogurt, and so on.

Even these groupings may occasionally lead to confusion or disagreement. Classification and indexing are complex tasks, and in many situations, there is no single solution that is acceptable to everyone. Card sorting exercises are useful to engage users and reach an initial design, which can then be refined with usability or A/B testing (see Chapter 5). Over time, as the structure is improved and as users gain familiarity with it, success rates will improve.

Tree-structured menu systems have the power to make large collections of data available to novice or intermittent users. If each menu has 10 items, a menu tree with four levels has the capacity to lead an untrained user through a

BOX 8.3

Rules for forming menu trees.

Grouping menu items in a tree such that they are comprehensible to users and match the task structure is sometimes difficult. The problems are akin to putting kitchen utensils in order; steak knives go together and serving spoons go together, but where do you put butter knives or carving sets? Problems include overlapping categories, extraneous items, conflicting classifications in the same menu, unfamiliar jargon, and generic terms.

- Use task semantics to organize menus.
- Limit the number of levels (i.e., prefer broad–shallow to narrow–deep).
- Create groups of logically similar items: e.g., Level 1: countries, Level 2: states, Level 3: cities.
- Form groups that cover all possibilities: e.g., age ranges: [0–9] [10–19] [20–29] and [>= 30].
- Make sure that items are non-overlapping: e.g., use "Concerts" and "Sports" over "Entertainment" and "Events".
- Arrange items in each branch by natural sequence (not alphabetically) or group related items.
- Keep ordering of items fixed (or possibly duplicate frequent items in dedicated sections of the menu).

collection of 10,000 destinations. That number would be excessively large for a word processor but is realistic in a newspaper, a library, or an enterprise web portal.

If the groupings at each level are natural and comprehensible to users and if users know the target, menu traversal can be accomplished in a few seconds—it is faster than flipping through a book. On the other hand, if the groupings are unfamiliar and users have only vague notions of the items that they're seeking, they may get lost for hours in the tree menus. Terminology from the user's task domain can help orient the user: Instead of using a title that is vague and emphasizes the computer domain, such as "Main Menu Options", use terms such as "Friendlibank Services" or simply "Games".

Menus using large indexes, such as library subject headings or comprehensive business classifications, are challenging to navigate, making search a valuable alternative (Chapter 15).

The *depth*, or number of levels, of a menu tree depends in part on the *breadth*, or number of items per level. If more items are put into the main menu, the tree spreads out and has fewer levels. This shape may be advantageous, but only if clarity is preserved. Several authors urge using four to eight items per menu, but at the same time, they urge using no more than three to four levels. With large menu applications, one or both of these guidelines must be compromised.

Many empirical studies have dealt with the depth/breadth tradeoff (Cockburn and Gutwin, 2008), and the evidence is strong that breadth should be preferred over depth as long as users can anticipate target location at each level. The navigation problem (getting lost or using an inefficient path) becomes more and more treacherous as the depth of the hierarchy increases. Of course, screen clutter must be considered in addition to the semantic organization. Given sufficient screen space, it is possible to show a large portion of the menu structure and to allow users to rapidly point in the flattened tree structure (Figs. 8.6 and 8.11).

Although tree structures are appealing, sometimes network structures are more appropriate. For example, in online shopping, it might make sense to provide access to banking information from both the personal profile and the checkout section of a link structure. A second motivation for using menu networks is that it may be desirable to permit paths between disparate sections of a tree rather than requiring users to begin a new traversal from the main menu. It is helpful to provide site maps and to preserve the notion of levels, as users may feel more comfortable if they have a sense of how far they are from the main menu.

8.4.2 Sequence, phrasing, and layout

Sequence Once the items in a menu have been chosen, the designer is still confronted with the choice of *presentation sequence*. If the items have a natural sequence—such as days of the week, chapters in a book, or sizes of eggs—the decision is trivial. Many cases have no task-related ordering, though, so the designer must choose from either alphabetic order, grouping of related items, and most frequently used items first. Categorical organization is generally preferable over alphabetical. Using frequency of use does speed up selection of the topmost items, but the loss of a meaningful ordering for low-frequency items may be disruptive, so it is best limited to small lists. Varying the sequence adaptively to reflect the current pattern of use has been shown to be disruptive, increasing confusion and selection time. In addition, users may become anxious that other changes might occur at any moment, undermining the users' learning of menu structures. To avoid disruption and unpredictable behavior, it is wise to allow users to specify if and when they want the menu restructured. A sensible compromise is to extract three or four of the most frequently selected items and put them near the top while preserving the order of the remaining items. This *split-menu* strategy proved appealing, statistically significantly improved performance, and has been adopted by commercial software (Fig. 8.15).

Adaptable menus (i.e., providing users with control over the sequence of menu items) is an attractive alternative to adaptive menus that adapt

FIGURE 8.15

Example of adaptive split menus in Microsoft Office. A font-selection menu lists the theme fonts and then the recently used fonts near the top of the menu (as well as in the full list), making it easier to quickly select the popular fonts. A thin line separates the sections.

automatically. One study compared the Microsoft Word version using adaptive menus with a variant providing users with the ability to switch between two modes of operation: the normal full-featured mode and a personal mode that users could customize by selecting which items were included in the menus (McGrenere et al., 2007). Results showed that participants were better able to learn and navigate through the menus with the personally adaptable version. Preferences varied greatly among users, and the study revealed some users' overall dissatisfaction with adaptive menus but also the reluctance of others to spend significant time customizing the interface. Novel approaches have used ephemeral adaptation (Findlater et al., 2009) to help users quickly identify important commands. With this technique, a small subset of menu items is immediately shown when the menu is displayed, while the remaining items are gradually faded into view over a few hundred milliseconds.

Phrasing For single menus, a simple descriptive title that identifies the situation is all that is necessary. For tree-structured menus, choosing titles is more difficult. One helpful rule is to use the words used for the menu items as the titles for the submenu or next pages. For example, it is reassuring to users to find that when they select "Business and financial services", they are shown a display that is titled "Business and financial services". It might be unsettling to get a display titled "Managing your money", even though the intent is similar. For webpages, a distinctive short title displayed as browser tab label will help users return to the page after they visit other tabs. A distinctive icon improves the tab label as well.

Just because an interface has words, phrases, or sentences as menu choices is no guarantee that it is comprehensible or provides adequate information scent (see Section 3.4 on theories).

Individual words (for example, "expunge") may not be familiar to some users, and often two menu items may appear to satisfy the user's needs when only one actually does (for example, "disconnect" or "eject"). This enduring problem has no perfect solution, but designers can gather useful feedback from colleagues, users, pilot studies, acceptance tests, and user-performance monitoring. The following directives may seem obvious but are listed here because they are so often violated:

- *Use familiar and consistent terminology.* Carefully select terminology that is familiar to the designated user community and keep a list of these terms to facilitate consistent use.
- *Ensure that items are distinct from one another.* Each item should be distinguished clearly from other items. For example, "Slow tours of the countryside", "Journeys with visits to parks", and "Leisurely voyages" are less distinctive than are "Bike tours", "Train tours to national parks", and "Cruise-ship tours".
- *Use consistent and concise phrasing.* Review the collection of items to ensure consistency and conciseness. Users are likely to feel more comfortable and to be more successful with "Animal", "Vegetable", and "Mineral" than with "Information about animals", "Vegetable choices you can make" and "Viewing mineral categories".
- *Bring the keyword to the fore.* Try to write menu items such that the first word aids the user in recognizing and discriminating between items—use "Size of type" instead of "Set the type size". Then, if the first word indicates that this item is not relevant, users can begin scanning the next item.

Layout While the layout of applications and websites can be assisted by the use of templates and website management tools, designers who establish guidelines for consistency across dozens or hundreds of screens will reduce users' anxiety by offering predictability (see Section 3.2). The following elements can be included:

- *Titles.* Some people prefer centered titles, but left justification is also acceptable.
- *Item placement.* Typically, items are left justified, with the item number or letter preceding the item description. Blank lines may be used to separate meaningful groups of items. If multiple columns are used, a consistent pattern of numbering or lettering should be used (for example, it is easier to scan down columns than across rows). See also Section 12.2 on display design.
- *Instructions.* The instructions should be identical in each menu and should be placed in the same position. This rule includes instructions about traversals, help, or function-key usage.
- *Error messages.* If the users make unacceptable choices, the error messages should appear in a consistent position and should use consistent terminology and syntax. Graying out unacceptable choices will help reduce errors.

Since disorientation is a potential problem, techniques to indicate position in the menu structure can be useful. In books, different fonts and typefaces may indicate chapter, section, and subsection organization. Similarly, in menu trees, as the user goes down the tree structure, the titles can be designed to indicate the level or distance from the main menu. Graphics, fonts, typefaces, or highlighting techniques can be used beneficially. For example, this set of headers from the Library of Congress collections webpages gives a clear indication of progress down the tree:

BROWSE BY TOPIC
Sports, Recreation & Leisure
Baseball
Baseball Cards 1887-1914

When users want to do a traversal back up the tree or to an adjoining menu at the same level, they will feel confident about what action to take.

8.5 Audio Menus

Audio menus found in *interactive voice response* (IVR) systems (Lewis, 2010) are useful when hands and eyes are busy, such as when users are driving or testing equipment and are ubiquitous in phone surveys or services and public-access situations that need to accommodate blind or vision-impaired users, such as information kiosks or voting machines.

With audio menus, instruction prompts and lists of options are spoken to users, who respond by using the keys of a keyboard or phone or by speaking. While visual menus have the distinct advantage of persistence, audio menus have to be memorized. Similarly, visual highlighting can confirm users' selections, while audio menus have to provide a confirmation step following the selection. As the list of options is read to them, users must compare each proposed option with their goal and place it on a scale from no match to perfect match. To reduce dependence on short-term memory, it is preferable to describe the item first and then give the number. A way to repeat the list of options and an exit mechanism must be provided (preferably by detecting user inaction).

Complex and deep menu structures should be avoided. A simple guideline is to limit the number of choices to three or four to avoid memorization problems, but this rule should be re-evaluated in light of the application. For example, a theater information system will benefit from using a longer list of all the movie titles rather than breaking them into two smaller, arbitrarily grouped menus. Dial-ahead capabilities allow repeat users to skip through the prompts. For example, users of a drugstore telephone menu might remember that they can dial 1 followed by 0 to be connected to the pharmacy immediately without having to listen to the store's welcome message and the list of options.

Voice recognition has finally reached an acceptable recognition rate and enables users to speak their options instead of pressing letter or number keys (see Section 9.2). Most systems still use numbered options to allow both keypad and voice entry (e.g., "To hear the options again, press or say nine"), but it leads to longer prompts and longer task-completion times.

To develop successful audio menus, it is critical to know the users' goals, make the most common tasks easy to perform rapidly, and keep prompts to a minimum (e.g., avoid permanent "Listen carefully, as our menu options have recently changed."). See Chapter 9, in particular Section 9.2, for more discussion of interactive voice response (IVR) systems.

8.6 Form Fill-in and Dialog Boxes

Selection is effective for choosing an item from a set of choices, but if the entry of names or numeric values is required, typing becomes more attractive. When many fields of data are necessary, the appropriate interaction style is form fill-in (Fig. 8.16). The combination of form fill-ins, menus, and custom widgets such as calendars or maps supports rapid navigation for a vast array of applications from airline-ticket booking to triage of new patients in the emergency room.

8.6.1 Form fill-in

There is a paucity of empirical research on form fill-in, but several design guidelines have emerged from practitioners (Jarrett and Gaffney, 2008). Software tools simplify design, help to ensure consistency, ease maintenance, and speed implementation, but even with excellent tools, the designer must still make many complex decisions.

The elements of form fill-in design include the following:

- *Meaningful title.* Identify the topic and avoid computer terminology.
- *Comprehensible instructions.* Describe the user's tasks in familiar terminology. Be brief; if more information is needed, make a set of help screens available to the novice user. A useful rule is to use the word "type" for entering information

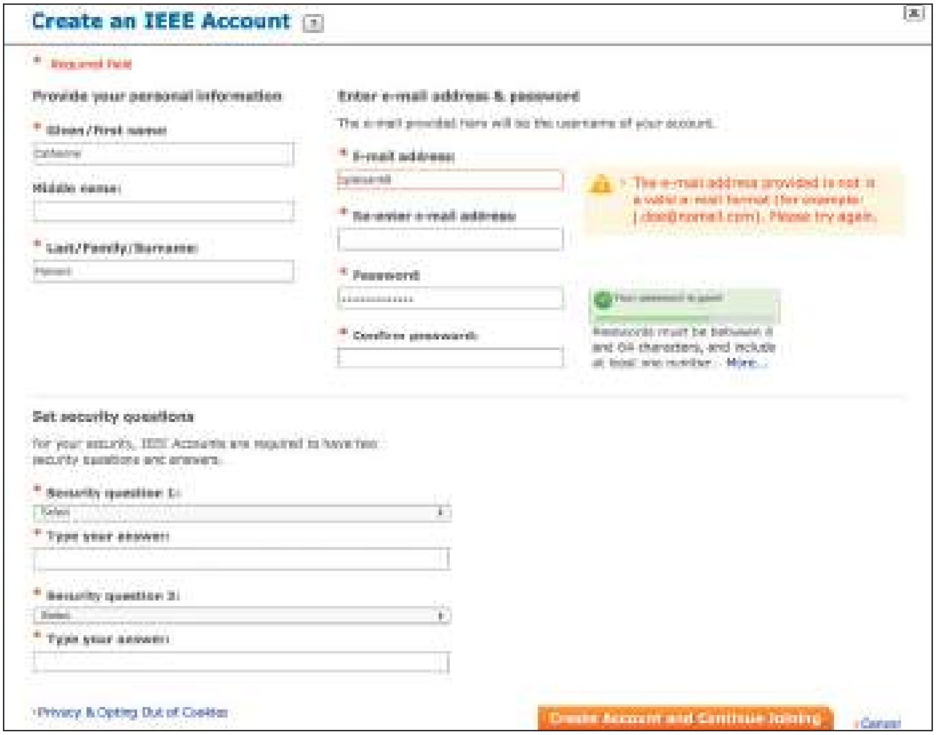


FIGURE 8.16

This form fill-in allows users to enter information when joining the IEEE Society. Fields are grouped meaningfully, and field-specific rules such as password requirements are provided next to the fields. The information is validated as it is provided (as opposed to when the form is submitted), and error messages explain how to correct problems (http://www.ieee.org).

and the word "press" for special keys such as the Tab, Enter, or cursor-movement (arrow) keys. Since "Enter" often refers to the special key with that name, avoid using it in the instructions (for example, do not use "Enter the address"; instead, stick with "Type the address"). Once a grammatical style for instructions is developed, be careful to apply that style consistently.

- *Label the fields.* Place the label in a consistent location (e.g., top or left of the field). A less desirable location is to place labels inside the fields, using a grayed-out font. It saves space, but the labels disappear as soon as users start typing, requiring users to remember what is needed, which often leads to errors.
- *Limit data entry.* Make sure all fields are really needed. Carefully set default values (e.g., use the current location). This is particularly important for small displays (see Box 8.4)—for example, using only the zip code instead of the city

BOX 8.4

Additional form fill-in guidelines for small displays.

- Include only critical data fields.
- Break long forms in multiple smaller ones.
- Use sensible defaults (e.g., current location or date).
- Place short labels on top of the fields, not to their left.
- Set the touch keyboard to match the data (e.g., numeric keyboard to enter a number).

and state. Maybe only a single phone number is enough, instead of asking for several alternatives. Some fields may be removed entirely and reserved only for large devices.

- *Explanatory messages for fields.* Information about a field (e.g., "Your e-mail address will be the user name of your account") or its permissible values should appear in a standard position, such as next to or below the field, preferably using a different font and style.
- *Error prevention.* Where possible, prevent users from entering incorrect values. For example, in a field requiring a whole number, do not allow the user to enter letters or decimal points.
- *Error recovery.* Summarize errors at the top of the page. Highlight errors in the form. If users enter unacceptable values, indicate permissible values for the field; for example, if the zip code is entered as 28K21 or 2380, the message might be "Zip codes should have 5 digits".
- *Immediate feedback.* Immediate feedback about errors is preferable. When feedback can be provided only after the entire form has been submitted, the location of the field needing correction should be made clearly visible (for example, by displaying the error message in red next to the field in addition to general instructions at the top of the form).
- *Logical grouping and sequencing of fields.* Related fields should be adjacent and should be aligned with blank spaces for separation between groups. The sequencing should reflect common patterns—for example, city followed by state followed by zip code.
- *Visually appealing layout of the form.* Alignment creates a feeling of order and comprehensibility. For example, the field labels "Name", "Address", and "City" can be right-justified so that the data-entry fields are vertically aligned. This layout allows the frequent user to concentrate on the entry fields and to ignore the labels.

- *Familiar field labels.* Common terms should be used. If "Home Address" were replaced by "Domicile", many users would be uncertain or anxious about what to enter.
- *Consistent terminology and abbreviations.* Prepare a list of terms and acceptable abbreviations and use the list diligently, making additions only after careful consideration. Instead of varying between such terms as "Address", "Employee Address", "ADDR.", and "Addr.", stick to one term, such as "Address".
- *Visible space and boundaries for data-entry fields.* Users should be able to see the size of the field and to anticipate whether abbreviations or other trimming strategies will be needed. An appropriately sized box can show the maximum field length.
- *Convenient cursor movement.* Provide a mechanism for moving the cursor between fields using the keyboard, such as the Tab key or cursor-movement arrows.
- *Required fields clearly marked.* For fields that must be filled in, the word "Required" or some other indicator (e.g., *) should be visible. Optional fields should follow required fields whenever possible.
- *Privacy and data sharing information.* Users will be anxious sharing their personal information and want to know how the data will be used and who will have access to it.
- *Accessibility.* For example, make sure the forms are navigable with a screen reader.
- *Completion signal.* It should be clear to the users what they must do when they are finished filling in the fields. Generally, designers should avoid automatic form submission when the final field is filled in because users may wish to review or alter previous field entries. When the form is very long, multiple Submit or Save buttons can be provided at different points in the form.

These considerations may seem obvious, but often designers will omit the title or an obvious way to signal completion or will include unnecessary computer file names, strange codes, unintelligible instructions, unintuitive groupings of fields, cluttered layouts, obscure field labels, inconsistent abbreviations or field formats, awkward cursor movement mechanisms, confusing error-correction procedures, or hostile error messages.

8.6.2 Format-specific fields

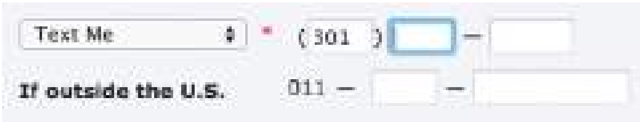
Using custom widgets and direct-manipulation interaction techniques can facilitate data entry and reduce errors. Calendars can be used to enter dates, seating maps can help users select airplane seats, and menus using photographs might clarify choices of pizza style.

Apps for touchscreen devices need to open the keyboard with the appropriate preset; for example, when a number is requested, the numerical keyword should appear by default. For e-mail addresses, the "@" and "." buttons need to be shown. For URLs, the ":" and "/" will be handy.

Alphabetic fields are customarily left-justified on entry and on display. Numeric fields may be left-justified on entry but then become right-justified on display. When possible, avoid entry and display of leftmost zeros in numeric fields (with zip codes being an exception). Numeric fields with decimal points should line up on the decimal points.

Pay special attention to such common fields as these:

- *Telephone numbers.* Offer a form to indicate the subfields:



  Be alert to special cases, such as the addition of extensions or the need for nonstandard formats for international numbers. When the user has typed all the needed digits of a field, the cursor should jump to the leftmost position of the next field.

- *Dates.* Providing a pop-up graphical calendar showing the current month will reduce the number of errors in some cases, but users may still want to type in the numbered field if moving the calendar to the correct date requires a large number of clicks (e.g., to enter a date of birth). Different formats for dates are appropriate for different tasks, and European rules differ from American rules. An acceptable standard may never emerge. Instructions need to show an example of correct entry. For example:

```
Date: _ _/_ _/_ _ _ _ (04/22/2016 indicates April 22, 2016)
```

  For many people, examples such as this one are more comprehensible than abstract descriptions like *MM/DD/YYYY*.

- *Times.* Even though the 24-hour clock is convenient, many people in the United States find it confusing and prefer a.m. and p.m. designations.
- *Dollar amounts (or other currency).* The currency sign should appear on the screen so users enter only the amount. If a large number of whole-dollar amounts are to be entered, users might be presented with a field such as

```
Deposit amount: $_ _ _ _ _._ _
```

with the cursor to the left of the decimal point. As the user types, the numbers should shift left, calculator style. To enter an occasional cents amount, the user can place the cursor on the right field (but remember that countries have different conventions for entering numbers—for example, many countries use a comma instead of a decimal point).

- *Passwords.* When asked to type a password, users also need a means to retrieve or change the password if they have forgotten it, but it is also important to avoid malicious use of that functionality. Designers who work with a security team will reach a higher level of security that matches the importance of the data and application (Bonneau et al., 2015; Shay et al., 2015) (Box 8.5). For example, two-factor identification (e.g., password and a code sent to a separate device) is strongly recommended for a bank application or an e-mail password change, but users will be annoyed if such procedures are required for unimportant accounts with little or no personal information. When asking users to create a new password, having them enter the password twice helps users catch typos and provides an opportunity to practice typing the password just created. Providing guidance and explanations of why a proposed password is not acceptable will help users generate stronger passwords (with possibly a meter that reflects the strength of the password).
- *CAPTCHAs.* A CAPTCHA (acronym for Completely Automated Public Turing test to tell Computers and Humans Apart) requires users to type text presented graphically to be illegible to computers. Including an audio option is necessary to make the CAPTCHA accessible to users with visual impairments. Newer versions observe user behavior with the CAPTCHA to predict whether a human or a robot is interacting (Fig. 8.17).

BOX 8.5

Guidelines for password creation.

- Use two-factor authentication for secure accounts.
- Indicate the rules for password creation.
- Ask for the password to be entered a second time.
- Hide the password with **** by default for privacy.
- Provide an option to unhide the password.
- Provide feedback encouraging strong password selection.

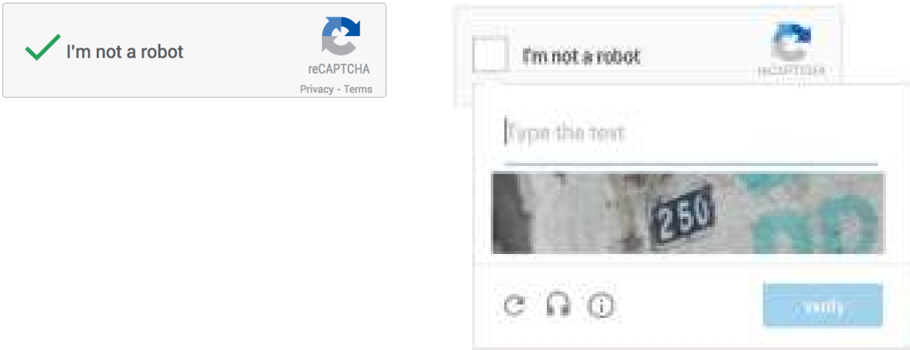


FIGURE 8.17

Google introduced a new reCAPTCHA in 2014. Observing the interaction, it predicts whether a human or a robot is clicking on the box but presents a more difficult CAPTCHA when in doubt. An audio version can play hard-to-understand words instead of the visual hard-to-read text.

8.6.3 Dialog boxes

Many tasks are interrupted to request users to select options, perform limited data entry, or review alerts and error messages (see Section 12.8). The most common solution is to provide a dialog box (Fig. 8.18).

Dialog-box design combines menu-selection and form fill-in issues with additional concerns about consistency across potentially hundreds of dialog boxes and relationships with other items on the screen. A guidelines document for dialog boxes will help strive for consistency. Dialog boxes should have meaningful titles to identify them, and the titles should have consistent visual properties. Dialog boxes are often shaped and sized to fit each situation, but distinctive sizes or aspect ratios may be used to signal errors, confirmations, or components of the application.

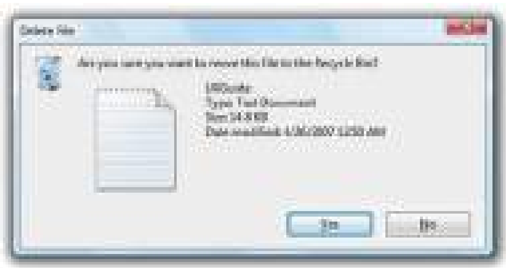


FIGURE 8.18

This dialog box includes a binary menu with two choices ("Yes" and "No"). The blue highlighting on Yes indicates that this selection is the default and that pressing Return will select it. Specific keyboard shortcuts can be made available. Escape closes the dialog box. Typing the letter "N" will select No, as indicated by the underlined letter "N".

Since dialog boxes usually pop up on top of some portion of the screen, there is a danger that they will obscure relevant information. Therefore, dialog boxes should be as small as is reasonable to minimize the overlap and visual disruption. Dialog boxes should appear near, but not on top of, the related screen items: When a user clicks on a city on a map, the dialog box about the city should appear just next to the click point. The classic annoying example is to have the Find or Spell Check box obscure a relevant part of the text. When multiple large displays are used, placing the dialog box in multiple locations simultaneously can result in faster interaction (Hutchings and Stasko, 2007).

Dialog boxes should be distinct enough that users can easily distinguish them from the background but should not be so harsh as to be visually disruptive. On desktop computers, keyboard shortcuts are essential to speed the response to dialog boxes. A common convention is to use Escape to cancel and close the dialog box and Enter to select the default command when appropriate. Dialog boxes do not always require users to answer or close them (e.g., the Find box in many applications can remain open after the search is performed). Modal dialog boxes require users to indicate their choice immediately, but non-modal dialog boxes allow users to continue their work and return to the dialog box again at a later time. When an alert is critical, dialog boxes may require immediate attention (Fig. 8.19) (https://sbmi.uth.edu/nccd/SED/Briefs/sedb-mu03.htm).

FIGURE 8.19

This dialog box is used to alert clinicians who try to prescribe the drug Warfarin because it increases the risk of bleeding in patients already on aspirin. Several possible actions are proposed. Overriding the alert is possible but requires confirmation by clicking a check box. Because of the severity of the alert, this is a modal dialog box and requires immediate action.

When tasks are complex, multiple dialog boxes may be needed, leading some designers to choose to use a tabbed dialog box in which two or more protruding tabs in one or several rows indicate the presence of multiple dialog boxes. This technique carries with it the potential problem of too much fragmentation; users may have a hard time finding what they want underneath the tabs. A smaller number of larger dialog boxes may be advantageous, since users usually prefer doing visual searches to having to remember where to find a desired control.

Practitioner's Summary

Designers who focus on organizing the structure and sequence of menus are more likely to match the users' tasks, priorities, and environment. If each menu is a meaningful, task-related unit, then the individual items will be distinctive and comprehensible. Favor broad and shallow hierarchical menus. For users who make frequent use of the system, shortcuts and gestures will greatly increase the speed of interaction. Permit simple traversals to the previously displayed menu and to the main menu. Remember that audio menus and menus designed for small devices require careful rethinking of what functions to include. For such menus, carefully limit the number of items, and consider frequency of use as a criterion for sequencing menu items. Gestures are useful for fluid interaction but are hard to discover and learn and often require complementary means of interaction. Consider direct-manipulation graphical widgets such as calendars or maps to facilitate data entry with form fill-in. Such widgets, along with immediate feedback and dynamic help, will help reduce errors and speed data entry.

Be sure to conduct usability tests and to involve human-factors specialists in the design process. When the interface is implemented, collect usage data, error statistics, and subjective reactions to guide refinement. Consider user-adaptable menu designs.

Researcher's Agenda

Experimental research could help to refine the design guidelines concerning organization of menus. How can differing communities of users be satisfied with a common organization when their information needs are markedly different? Should users be allowed to tailor the structure of the menus, or is there greater advantage in compelling everyone to use the same structure and terminology? Should a tree structure be preserved even if some redundancy is introduced? What's the best way to progressively introduce new users to large menu structures? How can users be encouraged to discover and learn new gestures or keyboard shortcuts? What further improvements will speed menu selection on small and very large displays? Can better guidance and feedback during password creation improve usability and security?

Research opportunities abound, and the quest for novel menu-selection strategies for small and large displays continues. Implementers would benefit from advanced software tools to automate the organization of menus (e.g., Bailly and Oulasvirta, 2014) and facilitate the design of responsive menus and their evolutionary refinements.

WORLD WIDE WEB RESOURCES

www.pearsonglobaleditions.com/shneiderman

- An extensive review of menu techniques: http://www.gillesbailly.fr/menua/
- Major suppliers describe the use of gestures in their guidelines: Google's Android, Apple's iOS, and Microsoft's Windows 8: https://www.google.com/design/spec/patterns/gestures.html,

  https://developer.apple.com/library/ios/documentation/UserExperience/Conceptual/MobileHIG/InteractivityInput.html#//apple_ref/doc/uid/TP40006556-CH55-SW1,

  https://msdn.microsoft.com/en-us/library/windows/desktop/dd940543(v=vs.85).aspx
- Stories of "less is more" for mobile devices: http://www.fastcompany.com/1816610/sharing-app-bump-30-slashes-most-features-proves-less-really-can-be-more
- Design patterns suggested by the UK government: https://www.gov.uk/service-manual/user-centred-design/resources/patterns
- Design winners in various categories (website, tablet, smart phone, etc.): http://www.awwwards.com
- Website accessibility example: http://www.raisingthefloor.com

The most interesting experience is browsing the web to see how designers have laid out menus or form fill-ins in online commerce, government websites, and intranets.

Discussion Questions

1. A telephone-based menu system is being designed for a magazine subscription service system. There are seven magazines available—*National Geographic, Travel and Leisure, Entrepreneur, Time, Golf, U.S. News & World Report*, and *Fortune*. Describe three reasonable orderings of the voice menus and justify each.
2. What are the elements of form fill-in design?
3. Design a touch screen music jukebox, which allows the user to select from a menu of the five most popular songs of the week. Draw a sketch of this interface for each of the following menu types—binary menu, multiple-item menu, check boxes, pull-down menus. Argue which design serves the user best.
4. You are in charge of designing a menu tree for navigating 1,250 books in a digital library. Present an argument of whether the menu should have larger depth (number of levels) or breadth (number of items per level).
5. Frequent menu users can become annoyed if they must make several menu selections to complete a simple task. Suggest two ways you can refine the menu approach to accommodate expert or frequent users.
6. When users are navigating through a menu structure, they may become disoriented. The authors suggest techniques to help alleviate this disorientation such as indicating the current position in the menu. Draw a sketch of how you can show users their position for an online car showroom, assuming the user has browsed with the following path:

   Main Menu → Mid-size Cars → Honda → Accord
7. Data entry is challenging for small devices. What are some of the ways in which this issue can be addressed?

8. Critique the design of the dialog box below. This dialog box is used to alert clinicians who try to prescribe the drug Warfarin because it increases the risk of bleeding in patients already on Aspirin.

References

Bailly, G., Lecolinet, E., and Nigay, L., Visual menu techniques , Research Report hal-01258368, Telecom ParisTech (2016) https://hal.archives-ouvertes.fr/hal-01258368

Bailly, G., and Oulasvirta, A., Toward optimal menu design, *Interactions 21*, 4 (2014), 40–45.

Bonneau, J., Herley, C., van Oorschot, P. C., and Stajano, F., Passwords and the evolution of imperfect authentication, *Communications of the ACM 58*, 7 (2015), 78–87.

Cockburn, A., Gutwin, C., Scarr, J., and Malacria, S., Supporting novice to expert transitions in user interfaces, *ACM Comput. Surv. 47*, 2 (2014), 36 pages.

Cockburn, A., and Gutwin, C., A predictive model of human performance with scrolling and hierarchical lists, *Human Computer Interaction 24*, 3 (2008), 273–314.

Elmqvist, N., Vande Moere, A., Jetter, H.-C., Cernea, D., Reiterer, H., and Jankun-Kelly, T., Fluid interaction for information visualization, *Information Visualization 10*, 4 (2011), 327–340.

Findlater, L., Moffatt, K., McGrenere, J., and Dawson, J., Ephemeral adaptation: The use of gradual onset to improve menu selection performance, *Proceedings of the SIGCHI Conference on Human Factors in Computing Systems*, ACM Press, New York (2009), 1655–1664.

Gutwin, C., Cockburn, A., Scarr, J., Malacria, S., and Olson, S. C., Faster command selection on tablets with FastTap, *Proceedings of the SIGCHI Conference on Human Factors in Computing Systems*, ACM Press, New York (2014), 2617–2626.

Hornbæk, K., and Hertzum, M., Untangling the usability of fisheye menus, *ACM Transactions on Computer-Human Interaction 14,* 2 (2007), 6.

Hutchings, D. R., and Stasko, J., Consistency, multiple monitors, and multiple windows, *Proceedings SIGCHI Conference on Human Factors in Computing Systems*, ACM Press, New York (2007), 211–214.

Jarrett, C., and Gaffney, G., *Forms That Work: Designing Web Forms for Usability*, Morgan Kaufmann (2008).

Koved, L., and Shneiderman, B., Embedded menus: Menu selection in context, *Communications of the ACM 29* (1986), 312–318.

Krug, S., *Don't Make Me Think: A Common Sense Approach to Web and Mobile Usability*, New Riders (2014).

Lewis, J., *Practical Speech User Interface Design,* CRC Press (2010).

Malacria, S., Bailly, G., Harrison, J., Cockburn, A., and Gutwin, C., Promoting hotkey use through rehearsal with ExposeHK, *Proceedings of the SIGCHI Conference on Human Factors in Computing Systems*, ACM Press, New York (2013), 573–582.

McGrenere, Joanna, Baecker, Ronald M., and Booth, Kellogg S., A field evaluation of an adaptable two-interface design for feature-rich software, *ACM Transactions on Computer-Human Interaction 14,* 1 (2007), 3.

Medhi, I., Toyama, K., Joshi, A., Athavankar, U., and Cutrell, E., A comparison of list vs. hierarchical UIs on mobile phones for non-literate users interface layout and data entry, *Proceedings of IFIP INTERACT'13: Human-Computer Interaction 2* (2013), 497–504.

Oh, U., and Findlater, L., The challenges and potential of end-user gesture customization, *Proceedings of SIGCHI Conference on Human Factors in Computing Systems*, ACM Press, New York (2013), 1129–1138.

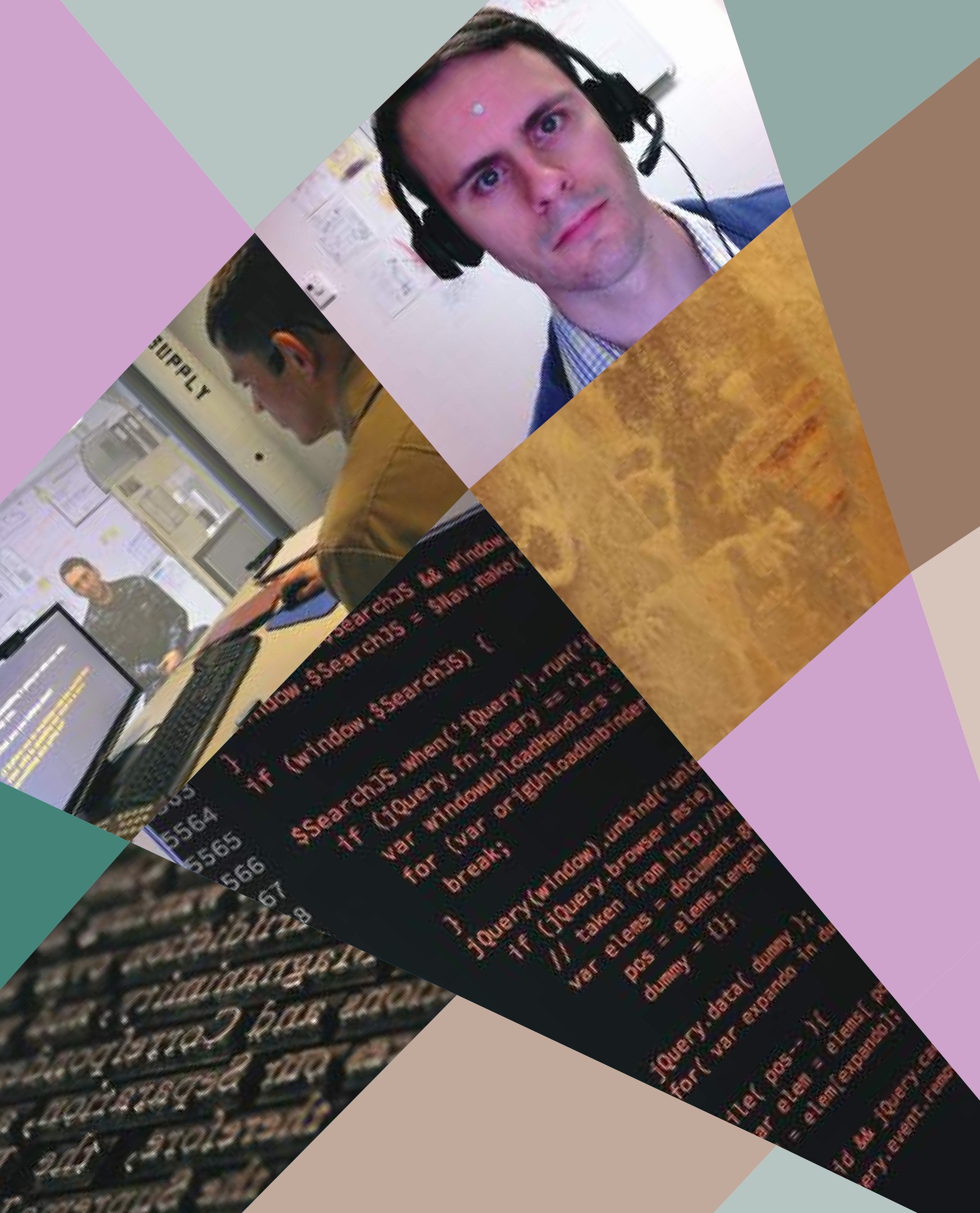
Shay, R., Bauer, L., Christin, N., Cranor, L. G., Forget, A., Komanduri, S., Mazurek, M. L., Melicher, W., Segreti, S., and Ur, B., A spoonful of sugar? The impact of guidance and feedback on password-creation behavior, *Proceedings SIGCHI Conference on Human Factors in Computing Systems,* ACM Press, New York (2015), 2903–2912.

Wigdor, Daniel, and Wixon, Dennis, *Brave NUI World: Designing Natural User Interfaces for Touch and Gesture*, Morgan Kaufmann, San Francisco, CA (2011).

Wroblewski, L., *Mobile First*, A Book Apart (2011).

Zhai, S., Kristensson, P. O., Appert, C., Andersen, T. H., and Cao, X., Foundational issues in touch-screen stroke gesture design: An integrative review, *Foundations and Trends in Human-Computer Interaction*, The essence of knowledge, *5,* 2 (2012), 97–205.

This page intentionally left blank


SUPPLY

CHAPTER 9

Expressive Human and Command Languages

> I soon felt that the forms of ordinary language were far too diffuse. . . . I was not long in deciding that the most favorable path to pursue was to have recourse to the language of signs. It then became necessary to contrive a notation which ought, if possible, to be at once simple and expressive, easily understood at the commencement, and capable of being readily retained in the memory.
>
> **Charles Babbage**
> *"On a Method of Expressing by Signs the Action of Machinery,"* 1826

CHAPTER OUTLINE

9.1 Introduction

The dream of speaking to computers and having computers speak has long lured researchers and visionaries. Arthur C. Clarke's 1968 fantasy of the HAL 9000 computer in the book and movie *2001: A Space Odyssey* has set the standard for performance of computers in science fiction and for developers of natural language systems. The reality is more complex and sometimes more frustrating than the dream, but much-improved speech recognizers have now joined the well-established speech telephone-based menu applications to reach a wide array of applications. Errors remain a significant challenge, and not all situations benefit enough from speech input to balance the cost of errors and the frustration of error correction. Once the commands, questions, or statements have been recognized, human language technologies may be needed to execute the appropriate action, initiate a clarifying dialogue, or provide translations.

Some applications simulate natural language interaction. They require users to speak a restricted set of the spoken commands that users have to learn and remember. Similarly, some textual interaction systems rely on the availability of vast text repositories that can be searched using standard search algorithms to find answers to questions written in full sentences. Repositories of translated text, such as the multiple language translations from the United Nations, can also help make good-quality translations of words, snippets, or full sentences.

See also:

Chapter 14, Documentation and User Support (a.k.a. Help)
Chapter 15, Information Search

The use of command languages in the early days of computing (e.g., DOS or Unix) receded with the advent of graphical user interfaces. However, command languages are still widely used by expert users of specialized applications from computer programmers to the millions of engineers and scientists using tools like MATLAB®, which combine a command language and graphical environment. In fact, one could argue that the spread of speech interfaces is re-invigorating the development of command languages as designers choose which combinations of words will be recognized as commands in speech interfaces.

While understanding natural language remains an unattainable dream, there are many applications that can successfully make use of the words people say, type, or listen to (Box 9.1).

This chapter starts with the rapidly growing speech interfaces (from speech recognition in Section 9.2 to speech production in Section 9.3) and then discusses

BOX 9.1

Speech technologies.

- Store and replay (museum guides)
- Dictation (document preparation, web search)
- Close captioning, transcription
- Transactions over the phone
- Personal "assistant" (common tasks on mobile devices)
- Hands-free interaction with a device
- Adaptive technology for users with disabilities
- Translation
- Alerts
- Speaker identification

human language technologies (Section 9.4) including translation educational applications. Finally, Section 9.5 reviews the traditional, yet expressive, command language interfaces.

9.2 Speech Recognition

Speech recognition has made significant progress in recent years (Huang et al., 2014) and is now being used in a number of well-targeted knowledge domains such as airline information, lost luggage, medical-record data entry, and personal digital assistants (Cohen 2004; Karat et al., 2012; Pieraccini, 2012; Bouzid and Ma, 2013; Neustein and Markowitz, 2013; Mariani et al., 2014). Driven by the difficulty of typing while using mobile devices (phones or touch tablets), spoken input has gained acceptability. More users learn to use spoken commands such as "Where is the closest coffee shop?" or "Tell John I will be late." Discoverability and learnability are often an issue, but commands can be spoken without looking at the screen, while driving a car (equipped with a hands-free phone), or while hiking on a bumpy trail. However, commands such as "Make space in my drive" are still a great challenge and would require extensive dialog design (see Section 9.4 on human language technology). Improved recognition rates are making dictation and transcription possible, but error correction remains a challenge, and most applications require users to learn and remember complex sets of commands to accomplish their tasks. Background noise and variations in user speech performance make the challenge of speech recognition still greater.

9.2.1 The place for spoken interaction

While speech recognition is used successfully in a growing number of applications, the vision of computers chatting leisurely with users about varied open-ended topics remains more of a fantasy than a reality. While HAL 9000 of *2001: A Space Odyssey* communicated with the ship crew mostly by voice, newer science-fiction writers have shifted their scenarios, with reduced use of spoken interaction in favor of larger visual displays and gestures, from *Star Trek: Voyager* to *Minority Report* and *Avatar* or *Mission Impossible 4*. Voice interaction with emotion-evoking robots remains a theme in movies such as *Her* and *Ex Machina*.

While early applications of speech recognition were mostly limited to discrete-word recognition (with extensive training for the system to learn a particular user's voice), the major breakthrough in the past decade has been the improvement of continuous-speech recognition algorithms and the availability of very large repositories of voice data on the web, which can be analyzed to train algorithms. The other significant advance that made speech recognition possible on mobile devices is the ability to process the spoken input remotely and quickly enough for rapid interaction. Reduced training (or its elimination with speaker-independent systems) has greatly expanded the scope of commercial applications. Quiet environments, head-mounted high-quality microphones, and careful choice of vocabularies improve recognition rates in all cases. Low-cost speech chips and compact microphones and speakers enable designers to include speech systems in high-volume products, such as dolls and other toys.

Applications are successful when certain conditions exist (see Box 9.2) and when they serve users' needs to work rapidly with low cognitive load and low error rates. Even as technical problems are being solved and the recognition rates are improving, spoken commands are more demanding of users' working memory than is hand/eye coordination and thus may be more disruptive to users while they are carrying out tasks. Speech requires use of limited resources, while hand/eye coordination is processed elsewhere in the brain, enabling a higher level of parallel processing. Planning and problem solving can proceed in parallel with hand/eye coordination, but they are more difficult to accomplish while speaking (Radvansky and Ashcraft, 2013). In short, speaking is more demanding than many advocates of speech recognition report.

Early applications include systems for aircraft-engine inspectors, who wear wireless microphones as they walk around the engine, their hands busy opening cover plates or adjusting components. They can issue orders, read serial numbers, or retrieve previous maintenance records by using limited vocabulary. As in all speech input systems, they can be disruptive to others who find the noise a serious distraction.

The benefits of speech recognition to people with physical or visual disabilities, even temporary ones, are rewarding to see (Fig. 9.1). Its value during mobile

BOX 9.2

Speech recognition and production: Opportunities and obstacles.

Opportunities

- When users have physical impairments
- When the speaker's hands are busy
- When mobility is required
- When the speaker's eyes are occupied
- When harsh or cramped conditions preclude use of a keyboard
- When application domain vocabulary and tasks are limited
- When the user is unable to read or write (e.g., children)

Obstacles to speech recognition

- Interference from noisy environments and poor-quality microphones
- Commands need to be learned and remembered
- Recognition may be challenged by strong accents or unusual vocabulary
- Talking is not always acceptable (e.g., in shared office, during meetings)
- Error correction can be time-consuming
- Increased cognitive load compared with typing or pointing
- Math or programming difficult without extreme customization

Obstacles to speech production

- Slow pace of speech output when compared with visual displays
- Ephemeral nature of speech
- Not socially acceptable in public spaces (also privacy issues)
- Difficulty in scanning/searching spoken messages

use can be significant for users who take the time to learn and remember what can be accomplished with spoken commands, but general users of office or personal computers are not rushing yet to adopt speech input and output devices.

9.2.2 Speech recognition applications

For designers of human-computer interaction systems, speech recognition technologies have many variations, which can also be combined productively with speech production (Li et al., 2015).

The goal of speech recognition is primarily to *produce text based on spoken input* (Lewis, 2011), the most straightforward application being *dictation*. Dictation systems have now reached recognition rates that are acceptable in many situations

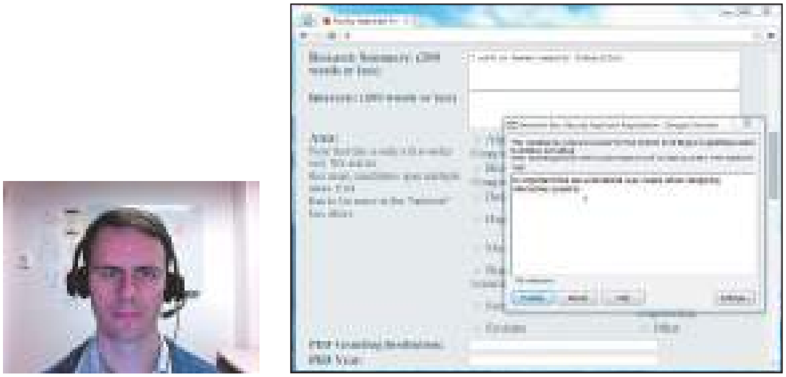


FIGURE 9.1

Using Dragon™ speech dictation and a head mouse (as made visible by the little silver dot on his forehead), a computer scientist is able to overcome a temporary hand disability.

(e.g., Google Docs' Voice Typing). They allow users to compose a document or speak search terms such as "movie theater in college park" and then correct mistakes with the keyboard instead of typing all the text. It can be a big time-saver with mobile devices, but keyboards, function keys, and pointing devices with direct manipulation often remain more rapid, depending on the quality of the recognition and the context of use (mobile or not), user's typing abilities, vocabulary complexity, nonnative speaker, and so on.

Ironically, technical fields with a lot of jargon are good candidates for speech recognition because of the distinctive nature of the terminology and the often-constrained documentation needs. For example, specialized systems for medical workers have become commercial successes (Nuance's Dragon® Medical, also embedded in electronic health record systems such as Cerner PowerChart Touch™). Dictation may be able to handle large vocabularies but inevitably requires specialized terminology to be useful to medical practitioners or lawyers.

While dictation is becoming practical, the cognitive burdens of dictation interfere with planning and sentence formation. In dictation, users may experience more interference between outputting their initial thought and elaborating on it. Spoken language may also be too informal compared with carefully typed sentences.

Speech recognition can also be used to transcribe recorded audio materials, either in real time or in a delayed fashion. It can facilitate *closed captioning* of radio or television programs or *transcription* of court proceedings or lectures.

Some applications may be beneficial even when there are errors. For example, errors will be irritating but acceptable for most television or YouTube viewers, but the payoff is that searching becomes possible. Where exact spelling is required, such as with person or place names, careful checking and error correction must be provided.

The other large category of speech recognition use is to allow users to *speak commands* that the user interface is trained to recognize effectively. This includes completing transactions over the phone, interacting with a device when direct manipulation is not convenient or possible, and using specialized voice services or "assistants." Dictation without using a keyboard will also require the use of commands to correct errors, start a new paragraph, or request the possibility to spell a name.

Specialized voice services or *"personal assistants"* like Siri, Google Now, Cortana, and Hound have become the more visible use of speech recognition. Because mobile interaction makes the use of keyboards impractical, speech becomes attractive to allow users to speak commands that execute the most common tasks performed on those devices, such as finding a location of interest, setting reminders, calendaring, communicating with others, or launching apps. Since the arrival of Apple's Siri in 2011, the competition has been fierce as companies compete to provide the most flexible and reliable services. The aim is to allow natural language, but users are often left wondering what they can say to get reliable results. This habitability problem remains a key problem, but logs of failed recognition facilitate efforts to broaden the acceptable inputs. Speaking assistants are now widely available, but many users never use them; others use only the few commands that they have learned and can remember; a smaller number of users can impress friends with apparent magic by having learned all the tricks. Companies do not report how much the assistants are being used, and while the demonstrations are impressive, the comparison tests often reveal problems (Ezzohari, 2015). Heavy use of spoken commands might be compared with the heavy use of keyboard shortcuts in traditional desktop users: not for everyone, but experts who have mastered them cannot live without them.

Speech is now widely used to complete *transactions* or access a service over the phone, for example, to report an electrical outage, trade stocks, or track lost luggage. These phone services, also called *information voice response* (IVR) systems, enable large financial savings for companies and provide 24/7 services for consumers (Lewis, 2011). Voice prompts welcome users and indicate what choices are possible. Users respond by pressing a number or speaking the word or short phrase that match their choice. Simple IVRs can be seen as an audio menu (see further discussion in Section 8.5).

A particularly challenging application is the *translation* of speech to facilitate human communication, such as foreign travelers or soldiers who must communicate in a language that they do not know well. Other emerging uses of speech recognition include the rapid *spotting of specific words or topics* in videos or telephone calls or *speaker verification* (also called *voice biometrics*). While users answers questions, the system verifies that they are who they claim to be. However, ensuring robust performance, coping with users with colds, and dealing with noisy environments are still challenges.

9.2.3 Designing spoken interaction

After designers have established that using voice is appropriate, they must decide whether the interaction will be conducted entirely via the audio channel (using speech recognition and production; e.g., on the phone or when users are driving or have visual impairments). Alternatively, they may integrate voice and visual channels to provide informative feedback or display results on the screen of a mobile device or a computer (Oviatt and Cohen, 2015). In general, combining input by voice with visual output is much preferable, as reading on the screen is much faster than listening to long prompts and allows rapid selection. Having access to a keyboard to correct errors is also of great help.

Initiation The first step in using spoken interaction is for users to indicate that they wish to start the spoken interaction. In phone systems, a welcome prompt is sufficient to get started, but on the screen, a start button is needed (usually in the shape of a microphone), or an option is available to use a voice command to turn on the listening (e.g., "Hey Siri" or "Wake up"). This spoken command has to be very carefully chosen so that it is not misrecognized, but false positives will inevitably occur, causing frustration and possible chaos if further commands are recognized without users noticing it. The initiation may be done for each command, or a separate spoken command may be needed to stop the recognition process. For example, the Nuance Dragon™ system uses "Wake up" and "Go to sleep" and allows users to chat with others—or just relax—in the middle of a spoken interaction session. An on-screen reminder of the stop command is helpful to novice and intermittent users.

Knowing what to say Next, users need to know what can be said and reliably recognized. Learnability is one of the main issues of human language technologies that attempt to mimic natural language. In IVR phone systems, spoken prompts guide users and invite them to press keys or speak one of the proposed menu choices. Because they are typically used by novices or intermittent users, the possible transactions remain simple and the dialogue entirely directed (e.g., users are instructed to please say "account balance," "bill pay," or "fund transfer"). Some IVR systems use more open-ended prompts (e.g., "What service do you need?") and rely on a series of dialogues to clarify and confirm choices. The use of speech recognition allows users to shortcut through menu trees, which can be successful when users know the names of what they seek, such as a city, person, or stock name. They may even be able to speak while the instructional prompt is being read. This *barge-in* technique works well when most users are repeat users who can immediately speak the options they have learned from previous experience. In all cases, the challenge is to identify novice users who attempt to use commands that are not recognized and switch them to a more directed mode that lists the possible commands. Users will become frustrated when they have to navigate a complex and deep menu structure (Section 8.4), when they are not allowed to "barge-in," when long spoken information segments contain irrelevant information, or when the menu of choices does not address their information need.

Users of mobile digital "assistants" are left with the burden of learning and remembering what the effective commands are. They may quickly become frustrated and quit if none of their attempts leads to success. Help can be provided with examples of commands (Fig. 9.2), or users are left to search blog postings to find lists of effective prompts (Cross, 2015), but those lists may be very long and commands still have to be remembered.

Recognition errors Slips produced by speech-recognition programs make for entertaining sections in product reviews in the trade press. Common errors occur when the vocabulary includes similar terms ("dime/time" or "Houston/Austin"). Challenges include dealing with regional or foreign accents and background noise. Users might also stutter, misspeak, or use the wrong terms. Dealing with unknown new words (and even failing to recognize that a word is unknown) can lead to confidently misrecognizing a similar-sounding word. Of course, the most difficult problem is matching the semantic interpretation and contextual understanding that humans apply easily to predict and disambiguate what was said. This problem was nicely highlighted in one of the few humorous titles of IBM technical reports: "How to Wreck a Nice Beach" (a play on "How to Recognize Speech"). To quote a summary of speech recognition's accomplishments, Huang et al. (2014) humbly report that "despite the impressive progress over the

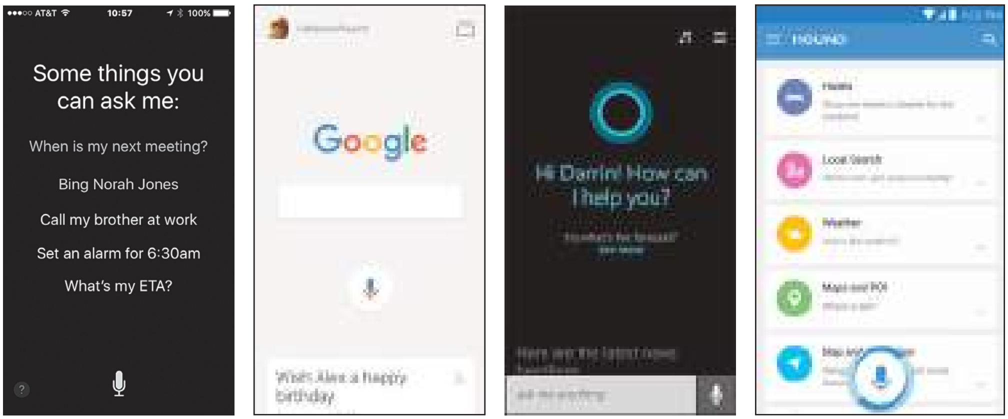


FIGURE 9.2

Mobile device assistants (from left to right: Siri, GoogleNow, Cortana, and Hound) all have similar microphone buttons but different ways of presenting suggestions.

past decades, today's speech recognition systems still degrade catastrophically even when the deviations are small in the sense (that) the human listener exhibits little or no difficulty. Robustness of speech recognition remains a major research challenge" (101). Finally, only a small portion of the myriad of world languages have adequate recognizers, and the mixing of two or more languages in the same sentence—which is common for multilingual speakers—also causes problems.

Early speech recognition systems were *speaker dependent*, and users were required to train the system to recognize their voice or deal with a particular microphone. This is not the case anymore for mobile phone use but is encouraged for professional applications that incorporate some level of personalization to increase the recognition rate. Changing microphones also required recalibration. In all cases, limiting the world of possible commands and carefully selecting easily differentiated terms dramatically improve recognition.

Correcting errors Correcting errors can be very taxing, especially when users do not have access to a keyboard or pointing device so all corrections have to be done using speech, possibly compounding errors with new ones. Even when a keyboard and pointer is available, having to correct errors is a significant distraction from the main task. A pause is generally required to separate dictation from editing commands, but providing correction commands that are very distinct will also facilitate their identification. Facilitating the erasing of last spoken text (e.g., saying "scratch that") allows repeating or rephrasing. Once a correction command has been identified (Fig. 9.3), alternative text can be proposed, or

FIGURE 9.3

Correcting a word during dictation using Nuance Dragon™. After saying "Correct finnish," the word is selected and possible corrections are displayed in a menu along with additional commands such as "spell that." Users can use the cursor, arrow keys, or voice to specify their choice.

users can add and record new terms (e.g., "IEEE" pronounced as "I triple E") or spell out words (e.g., for new names or cities).

Mapping to possible actions The secret of most successful speech recognition applications today is that they are limited to narrow application domains—so the world of actions is limited, and they use commands carefully chosen to increase recognition (e.g., using "scratch that" to delete text). Banking IVRs only know about banking terms and have a small set of possible actions. Users of personal assistants on mobile devices may impress friends with the variety of possible commands, but each app has a limited set of possibilities. This stems from two main causes: First, mobile applications designers by nature focus on a limited number of often-used functions that are used constantly. Second, because speech is a highly variable signal, large corpora of recorded speech matching the application domain are needed to achieve good recognition results, so speech recognition achieves much better results in application domains that have been studied and modeled extensively. Even if the speech recognition made no error, there are many levels of possible errors mapping the corresponding text to the expected action, as illustrated in Fig. 9.4. Companies continuously collect data from users as they speak and correct errors to improve both the speech recognition and the mapping to appropriate actions. Comparisons of today's assistants such as Siri, Google Now, Cortana, and Hound seem to suggest that mapping the recognized text to the most appropriate action is the most challenging task (Ezzohari, 2015).

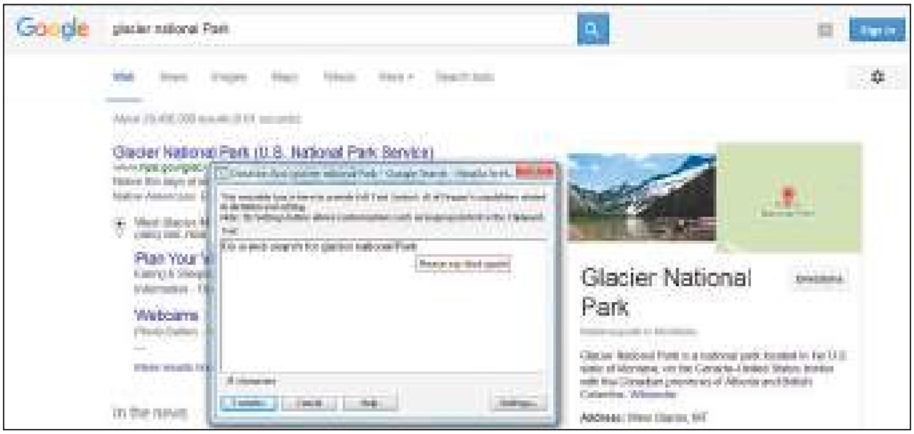


FIGURE 9.4

It can be difficult to remember what exact command will accomplish the task. In this example, when the user said, "Search the web for glacier national park," a Google search was launched and a search executed as expected, but when the user said, "Do a web search for glacier national park," all the words were accurately recognized but not as a command, so the text was placed in the Nuance Dragon™ dictation box.

Being able to rely on contextual information such as location or text from previous commands gives the impression of a more conversational interaction. For example, it might be possible to say "show me close by restaurants," "how about in Baltimore instead," "with 3 or more star reviews." Those chains of commands are significantly more difficult to interpret correctly and are today only achieved in constrained applications and by trained users who have learned what will be successful.

Feedback and dialogues During dictation or transcription, the recognized text is shown in the document being composed or in a dictation buffer, usually after a short delay (one to two seconds). Users can continue speaking or start correcting errors with the keyboard or by speaking navigation or editing commands. After correction, the text can also be transferred to a search box, the body of an e-mail message, or a field in a form. Applications tightly integrated with the speech recognition (opposed to relying on a dictation buffer) are more likely to be attractive and can generate spoken feedback as well.

Commands are usually executed directly, unless confirmation is preferable (e.g., "I am ready to e-mail this to Ben Shneiderman, should I go ahead?") or additional information or disambiguation is needed (e.g., "There are 2 John Smiths in your address book, which one should the e-mail be sent to?"). When context information has been used, feedback indicates how it was used. Specific questions may be asked to fill the holes in the task model and its attributes; for example, saying "Set an alarm" triggers a response asking "Set an alarm for when today?" (i.e., the date and time are missing from the alarm-setting task model, today was selected as the default date attribute, and a time attribute is still needed).

The availability of a display can greatly speed up interaction by presenting the proposed action in detail and only asking users to confirm or cancel, but it precludes eye-free operation (e.g., potentially endangering drivers). On the other hand, entirely spoken dialogues can be lengthy and even reveal information the user didn't want to be heard.

9.2.4 Spoken prompts and commands

When human language technology has been identified as appropriate for an application, prompts and commands resembling natural languages have to be designed. A language may have a simple or complex syntax and may have a few operations or hundreds, but the key issue—and the main usability determinant—is to adequately design clear prompts and a set of commands users can speak comfortably and remember easily and the system can recognize reliably.

The choice to use speech instead of keyboard entry is primarily a matter of user choice or possibility, but even with speech designers are the ones who decide what features to support, what commands will be used, how users will discover what is possible, and what feedback or error messages will be provided.

The designer's first step is to study the users' task domain. The outcome is a list of task actions and objects, which is then abstracted into a set of interface actions and objects. These items, in turn, can be represented with the low-level interface syntax. Observing users speaking aloud is critical to discover commands that users might speak "naturally." Both commands and prompts may include terms that are rarely used in direct manipulation or menu systems; for example, users are likely to say "set an appointment for tomorrow" even though no specific menu for "tomorrow" exists in the menus of the graphical calendar interfaces.

A typical form is a verb followed by a noun object with qualifiers or arguments for the verb or noun—for example, users might say "launch Facebook" or "set an alarm for 7 a.m." Human learning is greatly facilitated by meaningful structure. If a set of commands is well-designed, users will recognize its structure and easily encode it in their semantic-knowledge storage. For example, if users can uniformly edit words, sentences, and documents, this meaningful pattern is easy for them to learn, apply, and recall. On the other hand, if they must use different terms to change a word, revise a sentence, or alter a document, the challenge and potential for error grow substantially, no matter how elegant the syntax is. The "naturalness" will result from careful design and inclusion of synonyms (Fig. 9.5).

An effective way to test early versions of a spoken language interaction is to conduct a *Wizard of Oz evaluation* in which a hidden person is transcribing the spoken commands into text to simulate perfect recognition and typing dialog prompts that are shown to the unsuspecting participant on a screen (for an example, see Dyke et al., 2013).

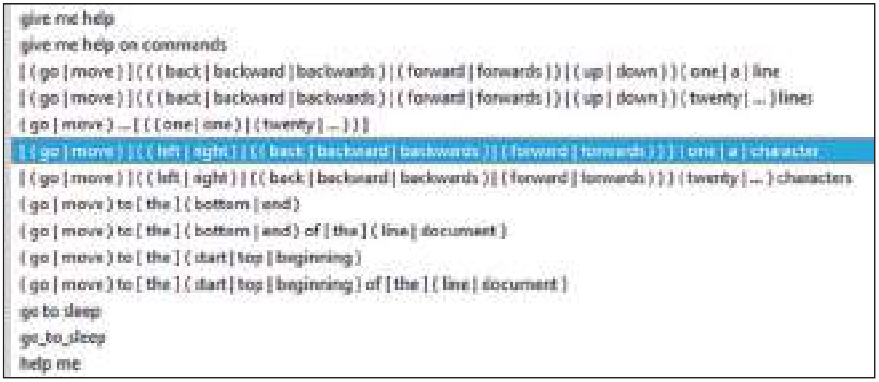


FIGURE 9.5

A small subset of the rich set of commands used in the Nuance Dragon™ speech recognition system. Synonyms are included and used consistently.

9.3 Speech Production

Speech production is usually successful when the messages are simple and short and users' visual channels are overloaded; when they must be free to move around or on the phone; or when the environment is too brightly lit, too poorly lit, subject to severe vibration, or otherwise unsuitable for visual displays. However, designers must cope with the four obstacles to speech output: the slow pace of speech output when compared with visual displays, the ephemeral nature of speech, acceptability and privacy issues in public spaces, and the difficulty in scanning/searching (Box 9.2).

There are three general methods to produce speech. A common type of speech generation available commercially is *formant synthesis*, which produces entirely machine-generated speech using a set of algorithms to product sounds based on the phonetic representation of the text. The speech sounds somewhat artificial and robot-like. *Concatenated synthesis* instead combines tiny recorded human speech segments into phonemes, words, and phrases into full sentences. The voice is more natural but requires significantly more storage and computing power to assemble sentences on the fly. Formant synthesis and concatenated synthesis can generate any sentence as needed. Finally, *canned speech* consists of a fixed set of digitized speech segments which can be assembled together to create longer segments (e.g., "The next bus will arrive in" followed by "11" then "minutes"), but the number of possible complete sentences is limited and the seams between segments may sound awkward.

The quality of generated speech can be evaluated in terms of understandability, naturalness and acceptability. For some applications, a computer-like sound may be preferred. For example, the robot-like sounds used in the Atlanta airport subway drew more attention than did a tape recording of a human giving directions. Interactive voice response systems (IVRs) typically mix canned speech segments and speech synthesis to allow appropriate emotional tone and current information presentation.

Audio books or audio tours in museums and tourist sites also use canned speech. They are successful because they allow users to control the pace while conveying the curator's enthusiasm or author's emotion. Educational psychologists conjecture that if several senses (sight, touch, hearing) are engaged, learning can be facilitated. Adding a spoken component to an *instructional system* or an *online help system* (Section 14.3.2) may also improve the learning process.

Alerts and warnings can be presented using speech. They have been used in automobile navigation systems ("Turn right onto route M1"), internet services ("You've got mail"), or utility-control rooms ("Danger, temperature rising"), but in most cases, the novelty wears off quickly. Talking supermarket checkout machines that read out products and prices were found to violate shoppers' sense of privacy about purchases. Only generic instructions are spoken now, but many consumers still find them too noisy. Similarly, annoying warnings from cameras (“Too dark—use flash”) and automobiles (“Your door is ajar”) were removed and replaced with gentler tones or visual indicators. Spoken warnings in cockpits and control rooms are still used because they are omnidirectional and elicit rapid responses. However, even in these environments, spoken warnings can be missed, especially when in competition with human-human communication, and multiple methods are used simultaneously (e.g., a visual alert or a dialog box).

Applications for the visually impaired are an important success story. Utilities like the built-in Microsoft Windows Narrator or Apple VoiceOver can be used to read passages of text or hear descriptions of items on the screen. Screen readers like Freedom Scientific’s JAWS, NV Access’s NonVisual Desktop Access (NVDA), or Apple VoiceOver allow users with visual impairments to productively navigate between windows, select applications, browse graphical interfaces, and of course read text. Such tools rely on textual descriptions being made available for visual elements (labels for icons and image descriptions for graphics). Reading speed is adjustable, which allows interaction to be speeded up as well when needed. Book readers are also widely used in libraries. Patrons can place a book on a copier-like device that scans the text and does an acceptable job of reading it.

The slow pace of normal spoken output, the ephemeral nature of speech, and the difficulty in scanning/searching remain challenges, but speech production is widely used because it enables services that would otherwise be too expensive; hiring well-trained customer-service representatives available 24 hours a day is not practical for many organizations.

9.4 Human Language Technology

Even before there were computers, people dreamed about creating machines that would be able to understand *natural language*—that is, be able to take the appropriate action in various contexts without users having to learn any command syntax or select from menus. It is a wonderful fantasy, but language is subtle; there are many special cases, contexts are complex, and emotional relationships have a powerful and pervasive effect in human-human communication. Although true comprehension and generation of open-ended language seem an inaccessible goal, there has been extensive research on human language technology; widespread use is slow in developing, primarily because the alternatives are more appealing. Contrary to common belief, human-human interaction is not necessarily an appropriate model for human operation of computers. Since computers can display information 1,000 times faster than people can enter commands, it is advantageous to use the computer to display large amounts of information and to allow users simply to choose among the items. Selection helps to guide users by making clear what objects and actions are available. For knowledgeable and frequent users who are thoroughly aware of the available functions, a precise, concise language (typed or spoken) is usually preferred (Section 9.5).

Natural language interaction (NLI) in the form of a series of exchanges resembling a dialogue is difficult to design and build for even a single topic. The key impediment is the *habitability* of the user interface—that is, how easy it is for users to determine what objects and actions are appropriate. Visual interfaces provide the cues for the semantics of interaction, but NLI interfaces typically depend on assumed user models. Users who are knowledgeable about their tasks—for example, stock-market brokers who know the stock codes (objects) and buy/sell actions—can place orders in natural language, but these users prefer compact command languages because they are more rapid and reliable.

While early conceptions of human language technology assumed that computers would parse natural language expressions in text or spoken forms and derive some level of "understanding" and description of users' "intent," the current successes rely instead on *statistical methods* based on the analysis of vast textual or spoken corpora and usage data of millions of users.

For example, question-answering strategies are successful in situations where there are relevant corpora and designers have crafted effective user interfaces that expand queries, search databases, show users alternatives, and present final results in ways that are most likely to be useful. Their success comes not from the understanding of the natural language but from the fact that the question at hand has already been asked before—using the same terminology—and has already been answered by others (Hearst, 2011). Another method is to analyze web search usage logs to find what results users seek often. For example, when users type "Leddo restaurant," human language technology extracts relevant queries from the dataset and identifies that "Leddo" does not exist but "Ledo" is a frequent entry. Then the word "restaurant" has been repeatedly identified as a term that leads users to look for an address, hours of operation, or a map, so that information can be presented by default. This can be done on the basis of frequency of past queries and on the log of previous users' actions.

Other applications include *extraction and tagging*. Extraction refers to the process of analyzing human language to create a more structured format, such as a relational database. The advantage is that the parsing can be done once in advance to structure the entire database and to speed searches when users pose relational queries. Legal (Supreme Court decisions or state laws), medical (scientific journal articles or patient histories), and journalistic (Associated Press news stories or *Wall Street Journal* reports) texts have been used. A variant is to tag documents based on content. For example, it is useful to have an automated analysis of business news stories to classify them as covering mergers, bankruptcies, or initial public offerings for companies in various industries such as electronics, pharmaceutical, or petroleum. Extracting and tagging applications are promising because users appreciate even a modest increase in suitable retrievals, and imperfect retrievals are more acceptable than errors in natural language interaction. On the other hand, errors can become quite problematic when the extracted information is used to make decisions or inform policies. One example is the use of human language technology in medicine. A large amount of information about medical conditions, treatments, and outcomes is buried in textual notes written by physicians in electronic health records. Automatically extracting diagnoses or test results out of the text notes can be very useful to identify possible candidates for a clinical trial, as all records will be reviewed by a clinician. On the other hand, the use of automatic tags for clinical decision making can be problematic. The rare cases of success are limited to situations with specific users, document types, and decision support goals (Demner-Fushman et al., 2009). *Sentiment analysis* is a specialized tagging, which can be applied to groups of news articles, reviews, or social media to monitor global changes in opinion, but tagging of individual documents remains error-prone.

Human language text generation is used for simple tasks, such as the preparation of structured weather reports ("80% chance of light rain in northern suburbs by late Sunday afternoon") in which generated reports from structured databases can be sent out automatically. Automatically generated text can be used to supplement standard data charts such as bar charts or scatterplots in order to make them more accessible to users with visual impairments (e.g., Google Sheet's Explore or iweave.com). More elaborate applications of text generation include preparation of reports of medical laboratory or psychological tests. The computer generates not only readable reports ("White-blood-cell count is 12,000") but also warnings ("This value exceeds the normal range of 3,000 to 8,000 by 50%") or recommendations ("Further examination for systemic infection is recommended"). Still more involved scenarios for text generation involve the creation of legal contracts, wills, or business proposals. Text summarization remains a much greater challenge with limited success, as summaries must capture the essence of the content and convey it accurately in a compact manner (Liu et al., 2012).

Human language technologies are used in *instructional systems*. Successful examples are in grammatical error detection and proofreading. Also widely used—but more controversial—is the automated scoring of short-answer responses or essays during student assessment. Human language technology has been introduced into a variety of educational contexts such as reading support. Tutorials with materials and pedagogy that have been carefully tested can provide feedback in natural language, which encourages students to stay engaged in the educational process. Simulations can also be used to practice communication skills learned in other settings (Fig. 9.6).

A remaining question is whether learning differs when students speak their responses or type them. A Wizard of Oz experiment (where a human transcribed the learner's speech before submitting it to the tutor) suggests that learning gains and preferences are similar with both modalities, but highly motivated students reported lower cognitive load and demonstrated increased learning when typing compared with speaking (D'Mello et al., 2011).

Translation between human languages has long been a goal (Green et al., 2015), but older strategies of word replacement with some grammatical parsing have given way to statistical methods based on having large databases with correct human translations, such as United Nations documents that appear in five required languages. Then well-designed user interfaces clarify what users can input in text windows, present translation options, show the translation, and guide subsequent user actions (Fig. 9.7). This design effort gets more complex with input errors and languages that may have unfamiliar characters, differ from English left-to-right formatting, and invoke words that do not exist in the target languages.

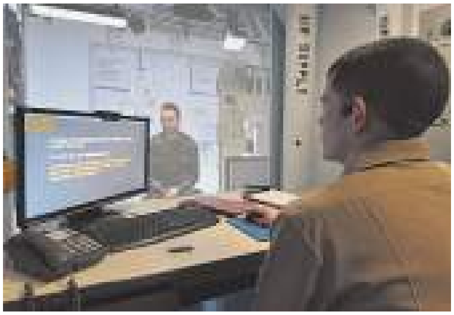


FIGURE 9.6

Using the Immersive Naval Officer Training System (INOTS), new Navy officers can practice their counseling skills in a virtual reality environment. Officers listen to an avatar and respond using spoken language, loosely following suggestions from multi-choice prompts presented on the screen and designed to match the learning objectives. The interaction is constrained, but assessment is facilitated (Dyke et al., 2013; http://www.netc.navy.mil/nstc/news_page_2012_02_24_2.asp).

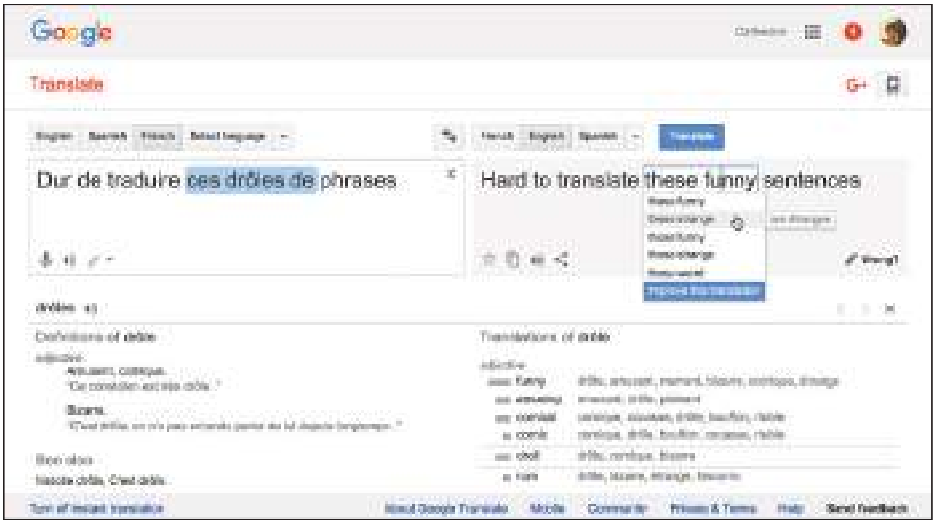


FIGURE 9.7

Google Translate, showing a French sentence translated in English. A click on the word “drôle” displayed its definition. Selecting “funny” highlighted “these funny” as well as the matching French words, and an alternative translation can be selected.

9.5 Traditional Command Languages

Early tally marks and pictographs on cave walls existed for millennia before precise notations for numbers or other concepts appeared. Eventually, languages with a small alphabet and rules of word and sentence formation dominated. Computers were quickly found to be effective to manipulate logical expressions, operate on the real world, or search vast libraries. These applications encouraged designers to find convenient notations to direct the computer, leading to command languages. With a command-language interface, users type a command and watch what happens. If the result is correct, the next command is issued; if not, some other strategy is adopted. By contrast, menu-selection users view or hear the limited set of menu items and they respond more than initiate. With command-language interfaces, users must recall the exact words to be used and the correct syntax, although prompted input often supplies a visual list of correct completions. For example, in the early days of command languages, users who needed to print a document may have been instructed to type:

```
CP TAG DEV E VTSO LOCAL 2 OPTCD=J F=3871 X=GB12
```

Another example is the Unix command used to delete blank lines from a file:

```
grep –v ^$ filea > fileb
```

Command-line interfaces are often preferred when the application is used in an advanced way (e.g., professionals using an application for hours every day). Casual users favor graphical user interfaces, but both styles of interface can be made available successfully because they do not always provide the same functionality. For example, in MATLAB, the command language can handle all the calculations, and a large subset of calculations is also available via the graphical user interface, which makes it easier for novice users to get started. Being able to type complex Boolean expressions using AND, OR, or NOT as well as regular expressions remains a key motivation for experienced users who can accomplish remarkable feats at amazing speed (Fig. 9.8).

Web addresses or URLs can be seen as a form of command language. Users come to memorize the structure of their favorite site addresses, even though the typical usage is to click on a link to select an address from a webpage or a search result page. The address field of browsers can also be used as a command line. For example, typing "(1024*768)/25" in the URL field in a Chrome browser will calculate the result, and typing "100 feet to meters" will launch the conversion tool and show the result: 30.48 meters.

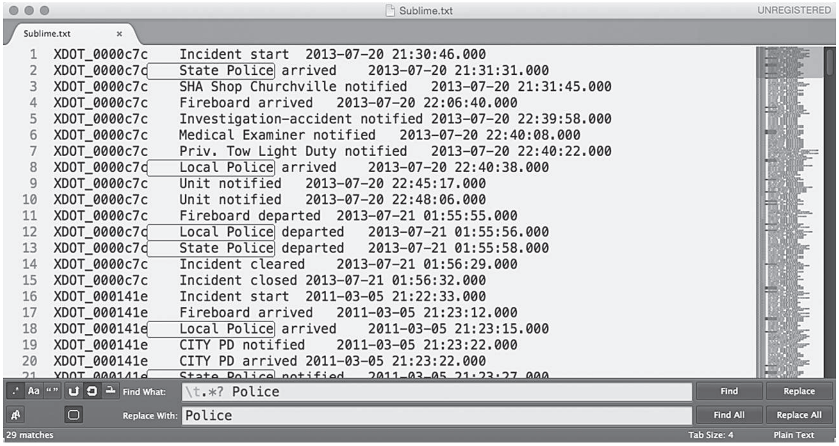


FIGURE 9.8

Using the Sublime text editor, a user is doing a search and replace in a data table using regular expressions. Typing " \t.*? Police" in the search box searches for a tab followed by zero or more characters, a space, and then the word "Police." The patterns found in the document are highlighted with a thin black line, showing that both "Local Police" and "State Police" have been found and selected. An overview of the entire document is visible on the right, revealing the presence of many other matches that can now be replaced all at once.

Twitter tags (#hcil, $TWTR, or @benbendc) can also be considered an example of new command language that needs to be learned and remembered, along with the proliferation of acronyms and abbreviations used by clever text-message writers (e.g., *LOL* for "laugh out loud" or *2G2BT* for "too good to be true"). In the traditional desktop environment, shortcut keys also remain heavily used by users who take the time to learn them (e.g., typing Ctrl-Q for Quit or Ctrl-P for Print; see Section 8.2.2). Programmers or professionals who use a single application all day long (e.g., a computer-aided design or publishing application) can memorize hundreds of commands and shortcuts, helping them gain mastery of their application (Cockburn et al., 2014).

One important opportunity linked to command languages is that histories can easily be kept and macros or scripts created to automate actions, but the essence of command languages is that they have an ephemeral nature and they produce an immediate result on some object of interest. Feedback is generated for correct commands, and error messages (Section 12.7) result from unacceptable forms or typos. Auto-completion is critical to help prevent errors. Command-language systems may offer brief prompts with choices, becoming closer to menu-selection systems. Command languages typically do not require a pointing device and therefore can become a lifesaver for users with visual impairments, which make the use of mice and touchscreens impractical.

Database-query languages for relational databases were developed in the middle to late 1970s; they led to the still widely used Structured Query Language, or SQL™, which emphasized short segments of code (2 to 20 lines) that could be typed and executed immediately. For example:

```
SELECT * FROM Products
WHERE Price BETWEEN 10 AND 20;
```

Here the goal of the user is to create a result rather than a program. A key part of database-query languages and information-retrieval languages is the specification of Boolean operations—AND, OR, and NOT—which can be very challenging to specify. See Chapter 15 for more on advances regarding searching.

Major considerations for expert users are the possibilities of tailoring the language to suit personal work styles and of creating named macros to permit several operations to be carried out with a single command. Macro facilities allow extensions that the designers did not foresee or that are beneficial to only a small fragment of the user community. A macro facility can become a full programming language that might include specification of arguments, conditionals, iteration, integers, strings, and screen-manipulation primitives plus library and editing tools—resembling a full-blown programming language.

In summary, while error rates remain high, the complexity and power of command languages have a certain attraction for a portion of the computer user community. Users gain satisfaction in overcoming the difficulties and becoming one of the inner circle "gurus" of their favorite command language.

Practitioner's Summary

The dream of natural language interaction has been mostly replaced by the effective use of statistical methods based on very large spoken and text corpora and logs of user interactions. Speech recognition for personal digital assistants and dictation has become increasingly successful, but errors and error correction remain issues. Speech-based approaches for guided interactions over telephones are also proving to be useful.

Speech generation, when well-designed, can support effective applications with phone, mobile devices, or book readers. Well-designed user interfaces enable integration of visual displays and touchscreens with speech. Text analysis, generation, and translation are useful human language technologies based on large training databases and appropriate user interfaces to prompt users and handle interactions.

Command languages continue to be attractive for expert users who learn the semantics and syntax because they can rapidly specify actions involving multiple options. Command languages allow sequences of commands to be stored for future use as a macro or script.

For command languages as well as spoken command languages, designers begin with a careful task analysis to determine what functions should be provided. Meaningful specific names aid learning and retention.

Researcher's Agenda

Speech recognition and generation user interfaces are maturing rapidly as effective designs have a growing user community. Improved user interfaces that integrate speech with visual displays and touchscreen controls may attract still larger communities; however, research on error reduction and methods to facilitate error correction is still needed.

Natural language interaction success stories are still elusive, but human language technology has become an important part of the success of search technology (Chapter 15). Spoken and text generation has shown value, so further research is warranted. For those who continue to explore specific applications, empirical tests and long-term case studies offer successful strategies to identify the appropriate niches and designs.

WORLD WIDE WEB RESOURCES

www.pearsonglobaleditions.com/shneiderman

- Designers will find many demonstrations of spoken interaction on YouTube. For example, the different styles of feedback and dialogue used by personal "assistants" can be seen at Hound Beta vs. Siri vs. Google Now vs. Cortana: https://www.youtube.com/watch?t=134&v=9zNh8kQLhfo.
- Experimenting with common search engines and personal digital assistants such as Siri or Google Now provides hints about the current human language technology strategies used for question answering.
- Translation: http://translate.google.com or http://www.babelfish.com
- Speech recognition commercial systems: http://www.nuance.com/dragon
- IVR dialog system: http://www.ibm.com/smarterplanet/us/en/ibmwatson/developercloud/dialog.html

Discussion Questions

1. Consider voice-activated digital assistants such as Siri, Cortana, or Google Talk. Identify a situation or scenario where you chose to use this personal assistant, and identify a scenario where you chose to avoid it.
2. As a follow-on to the previous question, produce a thoughtful argument about what role spoken interaction should have in user interfaces. Be sure to list at least three benefits and limitations of spoken interaction.
3. Briefly describe the applications of speech recognition.
4. What are the obstacles to speech recognition and production?
5. There exist applications of human language understanding technology. Name some examples.
6. List several situations when command languages can be attractive for users.

References

Bouzid, Ahmed, and Ma, Weiye, *Don't Make Me Tap! A Common Sense Approach to Voice Usability*, Dakota Press (2013).

Cockburn, A., Gutwin, C., Scarr, J., and Malacria, S., Supporting novice to expert transitions in user interfaces, *ACM Computing Surveys 47*, 2 (2014), Article 2.

Cohen, M. H., Giangola, J. P., and Balogh, J., *Voice User Interface Design*, Addison Wesley (2004).

Cross, J., A list of all the Google Now voice commands, Greenbot blog http://www.greenbot.com/article/2359684/system-software/a-list-of-all-the-ok-google-voice-commands.htm (2015).

Demner-Fushman, D., Chapman, W. W., and McDonald, C. J., What can natural language processing do for clinical decision support? *Journal of Biomedical Informatics 42*, 5 (2009), 760–772.

D'Mello, S. K., Dowell, N. N., and Graesser, A., Does it really matter whether students' contributions are spoken versus typed in an intelligent tutoring system with natural language? *Journal of Experimental Psychology 17*, 1 (2011), 1–17.

Dyke, G., Adamson, A., Howley, I., and Rosé, C. P., Enhancing scientific reasoning and discussion with conversational agents, *IEEE Transactions on Learning Technologies 6*, 3 (2013), 240–247.

Ezzohari, H., [ULTIMATE] personal assistant review: Hound vs Siri vs Google Now vs Cortana http://www.typhone.nl/blog/ultimate-voice-assistant-review/ (2015).

Green, S., Heer, J., and Manning, C. D., Natural language translation at the intersection of AI and HCI, *Communications of the ACM 58*, 9 (2015), 46–53.

Hearst, M. A., "Natural" search user interfaces, *Communications of the ACM 54*, 11 (2011), 60–67.

Huang, X., Baker, J., and Reddy, R., A historical perspective of speech recognition, *Communications of the ACM 57*, 1 (2014), 94–103.

Karat, M-C., Lau, J., Steward, O., and Yankelovich, N., Speech and language interfaces, applications and technologies, in Jacko, J. (Editor), *The Human-Computer Interaction Handbook,* CRC Press (2012), 367–386.

Lewis, J. R., *Practical Speech User Interface Design*, CRC Press (2011).

Li, Jinyu, Deng, Li, Haeb-Umbach, Reinhold, and Gong, Yifang, *Robust Speech Recognition: A Bridge to Practical Applications*, Academic Press (2015).

Liu, S., Zhou, M. X., Pan, S., Song, Y., Qian, W., Cai, W., and Lian, X., TIARA: Interactive, topic-dased visual text summarization and analysis, *ACM Transactions on Intelligent Systems Technology 3*, 2 (2012), 28 pages.

Mariani, Joseph, Rosset, Sophie, Garnier-Rizet, and Devillers, Laurence (Editors), *Natural Interaction with Robots, Knowbots and Smartphones: Putting Spoken Dialog Systems into Practice,* Springer (2014).

Neustein, A., and Markowitz, J. A. (Editors), *Mobile Speech and Advanced Natural Language Solutions,* Springer (2013).

Oviatt, Sharon, and Cohen, Philip, *The Paradigm Shift to Multimodality in Contemporary Computer Interfaces*, Morgan & Claypool (2015).

Pieraccini, Roberto, *The Voice in the Machine: Building Computers that Understand Speech*, MIT Press (2012).

Radvansky, G., and Ashcraft, M., *Cognition,* 6th Edition, Pearson (2013).

This page intentionally left blank

CHAPTER 10

Devices

> The wheel is an extension of the foot, the book is an extension of the eye, clothing, an extension of the skin, electric circuitry an extension of the central nervous system.
>
> **Marshall McLuhan and Quentin Fiore**
> *The Medium Is the Message*, 1967

CHAPTER OUTLINE

10.1 Introduction

Input and output devices represent the physical medium through which users operate computers. Along with improvements in computer processor speeds and storage capabilities in the past 50 years, their physical form factor and basic functionality have also changed dramatically. Only two decades ago, the standard computer platform was the desktop or laptop personal computer equipped with a screen, a mouse, and a keyboard, but mobile devices have revolutionized the face of computing to the point that many people do not realize that their ever-present smartphones, tablets, or portable MP3 players are, indeed, powerful computers. Computing has reached a point where it is deeply woven into the very fabric of our everyday existence (Dourish and Bell, 2011). With easily more than 5 billion mobile devices in existence, more than 25% of them "smart" and able to access the internet, compared to some 800 million personal computers, it is clear that mobile computing is the universal computing platform for the world (Baudisch and Holz, 2010). What's more, unlike the previous generation of personal computers, this new pervasiveness of mobile computing is no longer restricted to industrialized parts of the world but is quickly becoming an integral and integrated aspect of life even in rural, poor, and underdeveloped regions (Pew Research Center, 2014). In fact, many regional efforts, such as DataWind's $35 Aakash tablet designed for the Indian market (Fig. 10.1) as well as the Uhuru Tab by Rlg Communications Limited in Ghana, are making significant inroads toward making advanced computing available to everyone.

The veritable explosion of new and exciting computing technology has increased the importance of interaction design so as to accommodate such a wide diversity of input and output modalities. To keep up with this rapid pace of change, successful designers are increasingly employing micro-HCI and macro-HCI theories to transcend the specific capabilities and characteristics of individual devices. Some of these theories involve micro-scale ideas on consistency, responsiveness, discoverability, hierarchies, information architecture, and feedback as well as macro-scale ones on context, social settings, emotions, learning, and personalization. Refer to Chapter 3 for a discussion on micro-HCI and macro-HCI.

Despite increased complexity, devices also represent the part of computer technology that perhaps has the largest capacity for game-changing innovation for user interfaces. Indeed, the hype of such innovations often concerns the user-interface aspects of a new device. The Apple iPhone and iPad changed smartphones and tablet computing overnight when they were introduced, mainly on account of their

See also:

Chapter 7, Direct Manipulation and Immersive Environments

Chapter 8, Fluid Navigation

FIGURE 10.1

Indian IT minister Kapil Sibal announcing the Aakash, a $35 tablet for the Indian market, in 2011.

smooth and natural user experience. The Nintendo Wiimote and the Xbox Kinect introduced gestural and full-body interaction, respectively, to the living rooms of millions of people around the world. Finally, the Oculus Rift and the Microsoft HoloLens are bringing virtual and augmented reality to life for everyone (see Chapter 7).

Given such diversity and scope, this chapter gives a brief introduction to the most important families of input and output devices. The chapter first reviews text entry (Section 10.2), including keyboards and keypads as well as their layout, physical design, and accessibility adaptations. It also discusses text entry techniques for mobile devices. Pointing (Section 10.3) is another common interaction for interfaces, and the section below reviews ideas for how to make pointing more efficient, more accurate, and more accessible. Section 10.4 presents both traditional as well as novel display technologies, focusing on particularities of large and small displays as well as wearable computing

FIGURE 10.2

Many baby monitors use video cameras to provide a real-time feed of the baby's activities on a remote display device. There are also some wearable baby monitors that incorporate advanced features such as a baby's heart rate, respiration patterns, and skin temperature, which parents can track using their smartphone.

and shape-changing displays. Examples of possible solutions for users with disabilities are distributed throughout the chapter.

10.2 Keyboards and Keypads

Text entry is one of the most common input tasks, and the primary mode of text entry is still the keyboard (Fig. 10.3). Despite having received much criticism over the years, the keyboard is very successful and still represents the most efficient text-entry mechanism. Billions of people use keyboards; although the rate for beginners is generally less than one keystroke per second and the rate for average office workers is five keystrokes per second (approximately 50 words per minute), some users achieve speeds of up to 15 keystrokes per second (approximately 150 words per minute). Contemporary keyboards generally permit only one keypress at a time, although dual keypresses are used for capitals (Shift plus a letter) and special functions (Ctrl or Alt plus a letter).

More rapid data entry can be accomplished if several keys can be pressed simultaneously (i.e., *chording*). An inspiration might be the piano keyboard, an impressive data-entry device that permits several finger presses at once and is responsive to different pressures and durations. Similarly, chord keyboards use multiple keypresses that represent several characters or entire words. While this requires training and continued use, such chorded keyboards can allow for text entry outside the standard office environment, such as one-handed or eyes-free typing on small mobile devices or for wearable computing. In the courtroom,

FIGURE 10.3

An Apple MacBook Air laptop with a QWERTY keyboard (left) showing the inverted T movement keys at the bottom right and function keys across the top. A multi-touch trackpad supports pointing. On the right, a detail photograph of a Lenovo laptop keyboard shows a pointing stick (also called a trackpoint) mounted between the G and H keys on the keyboard.

such devices are called stenotypes and allow court reporters to rapidly enter the full text of spoken arguments at rates of up to 300 words per minute. However, this feat requires months of training and frequent use to retain the complex patterns of chord presses.

10.2.1 Keyboard layouts

The Smithsonian Institution's National Museum of American History in Washington, DC, has a remarkable exhibit on the development of the typewriter. During the middle of the nineteenth century, hundreds of attempts were made to build typewriters, with a stunning variety of positions for the paper, mechanisms for producing characters, and layouts for the keys. By the 1870s, Christopher Latham Sholes's design was becoming dominant—it had a good mechanical design and a clever placement of the letters that slowed down the users enough that key jamming was infrequent. This so-called *QWERTY layout* puts frequently used letter pairs far apart, thereby increasing finger travel distances.

Sholes's success led to such widespread standardization that, more than a century later, almost all keyboards use the QWERTY layout or one of its variations developed for other languages. The development of electronic keyboards eliminated the mechanical problems of typewriters and led many twentieth-century inventors to propose alternative layouts to reduce finger travel distances. The *Dvorak layout* could increase the typing rate of expert typists from about 150 words per minute to more than 200 words per minute and even reduce errors. Its failure to gain acceptance is an interesting example of how even documented improvements can be impossible to disseminate because the perceived benefit of change does not outweigh the effort required to learn a new, nonstandard interface.

A third keyboard layout of some interest is the *ABCDE style*, which has the 26 letters of the English alphabet laid out in alphabetical order. The rationale here is that non-typists will find it easier to locate the keys. A few data-entry terminals for numeric and alphabetic codes still use this style, though studies have shown no advantage for the ABCDE style; users with little QWERTY experience are eager to acquire this expertise and often resent having to use the ABCDE layout.

Number pads are a further source of controversy. Telephones have the 1–2–3 keys on the top row, but calculators place the 7–8–9 keys on the top row. Studies have shown a slight advantage for the telephone layout, but most computer keyboards use the calculator layout.

Some researchers have recognized that the wrist and hand placement required for standard keyboards is awkward and have proposed more ergonomic keyboards. Various geometries have been tried with split and tilted keyboards, but empirical verification of benefits in typing speed, accuracy, or reduced repetitive strain injury is elusive.

10.2.2 Accessible text entry

While people with motor impairments often can still use regular keyboards, albeit very slowly, several approaches to aid such users exist. Early solutions were based on large menus of fixed choices, but methods currently used in practice include adaptive keyboards, where keys are lowered instead of raised to aid acquisition, as well as on-screen keyboards accessed using alternative input devices like head pointers or oversized trackballs. All such text-entry methods can be improved significantly by incorporating dictionary-based auto-completion as well as automatic error correction (Kane et al., 2008). In contrast, visually impaired users represent a particular challenge for text entry. PerkInput (Azenkot et al., 2012) and BrailleTouch (Southern et al., 2012) both provide nonvisual input methods for one-handed or two-handed Braille typing on multi-touch smartphone displays.

Some techniques go beyond the traditional keyboard. Dasher predicts probable characters and words as users make their selections in a continuous 2-D stream of choices and has been adapted to brain-computer interfaces (BCI), where people use their brain alone to input text (Wills and MacKay, 2006). Also, orbiTouch's Keyless Keyboard replaces the keys with two inverted bowls, on top of which the user's hands rest comfortably (Fig. 10.4). A combination of small hand movements and small finger presses on the two bowls selects letters or controls the cursor. No finger or wrist movement is needed, which might be helpful to users with carpal tunnel syndrome or arthritis.

Finally, yet another approach reconsiders the use of a keyboard entirely. One idea is to rely on pointing devices such as mice, touchpads, or eye-trackers for data entry. Another builds on wearable devices, such as a wristband or ring form factor, to enter text (Ye et al., 2014). Common among many accessible text-entry methods, particularly for mobile settings, is the increasing use of speech input for this purpose (Chapter 9).

FIGURE 10.4

orbiTouch Keyless Keyboard with integrated mouse functionality (http://orbitouch.org/). The orbiTouch requires small finger presses and no actual hand motion to operate yet supports high-performance typing and pointing.

10.2.3 Keys

Keyboards keys have been refined carefully and tested thoroughly in research laboratories and the marketplace. The keys tend to have slightly concave surfaces for good contact with fingertips and a matte finish to reduce both reflective glare and the chance of finger slips. Keypresses require a 40- to 125-gram force and a displacement of 1 to 4 millimeters, which enables rapid typing with low error rates while providing suitable feedback to users. An important element in key design is the profile of force displacement. When the key has been depressed far enough to send a signal, the key gives way and emits a very light click. This tactile and audible feedback is extremely important in touch typing; hence, membrane keyboards that use a nonmoving surface are difficult to use for extensive touch typing. However, such keyboards are durable and therefore acceptable for challenging environments such as fast-food restaurants, factory floors, or amusement parks.

Certain keys, such as the space bar, Enter key, Shift key, or Ctrl key, should be larger than others to allow easy, reliable access. Other keys, such as Caps Lock and Num Lock, should have a clear indication of their state, such as by physical locking in a lowered position or by an embedded light. Large-print keyboards are available for vision-impaired users. The placement of the cursor-movement keys (up, down, left, and right) is important in facilitating rapid and error-free use. The popular and compact inverted-T arrangement of arrow keys (Fig. 10.3) allows users to place their middle three fingers in a way that reduces hand and finger movement. The cross arrangement is a good choice for novice users. Some large keyboards reuse the peripheral eight keys on the numerical keypad (all keys except the central 5 key) to simplify diagonal movements. For such keyboards, the Num Lock key is used to toggle between keypad and arrow mode. In some applications, such as games, where users spend hours using the movement keys, designers reassign letter keys as cursor-movement keys to minimize finger motion between the movement keys and other action keys. The WASD keys are often used for this purpose. Finally, the auto-repeat feature, where repetition occurs automatically with continued depression, may improve performance, but control of the repetition rate must be provided to accommodate user preferences (this is particularly important for very young users, older adult users, and users with motor impairments).

10.2.4 Mobile text entry

As computers morph into new form factors—such as tables, tablets, and phones—as well as become universally usable for a broader population of users from different backgrounds, nationalities, and capabilities, text entry is also changing beyond the traditional keyboard. Most older or low-cost mobile devices provide only a numeric keypad. Entering text using keypads requires multiple taps, where users hit a number key multiple times to cycle through several letters assigned to that key. Using the same key for consecutive letters requires the user to pause between letters. Predictive techniques, such as T9® by Tegic Communications, use dictionary-based disambiguation to speed up text entry and are often preferred for writing longer texts. Similarly, LetterWise uses the probabilities of prefixes and facilitates the entry of non-dictionary words, such as a proper nouns, abbreviations, or slang. After training, users were able to type 20 words per minute with LetterWise compared with 15 words per minute with multi-tap (MacKenzie et al., 2001).

While the current generation of smartphones, heralded by the initial release of the Apple iPhone (Fig. 10.6) in 2007, tends to eschew physical keyboards in favor of soft keyboards, some still use a traditional QWERTY keyboard. Fig. 10.5 shows two such mobile devices. Physical keyboards are still preferred by many mobile users who need to enter large amounts of text using their phones, such as to manage e-mail on the go. With practice, users can reach speeds of 60 words per minute when using both thumbs with those mechanical keyboards or more when the device auto-corrects "off-by-one" errors where the user accidentally presses a key adjacent to the one intended (Clawson et al., 2008).

Nevertheless, the proliferation of touchscreen technology means that physical mobile keyboards are increasingly being replaced by virtual, or so-called "soft," keyboards, where the keyboard is merely a visual representation on the touchscreen (Dunlop and Masters, 2008). Projection keyboards, where the physical world is appropriated (Harrison, 2010) to display an image of the keyboard, are based on the same principle. The benefit of soft keyboards is that they can be dynamically relabeled, such as for a new character set or layout, as well as rescaled and rotated to fit the physical display and device orientation

FIGURE 10.5

A Blackberry Q10 (http://www.blackberry.com) shown here on the left with a small physical QWERTY keyboard; users typically type with one finger or with both thumbs. On the right, a larger keyboard uses the longer dimension of a LG Cosmos 2 device and can be slid back into the device when not needed (http://www.LG.com/).

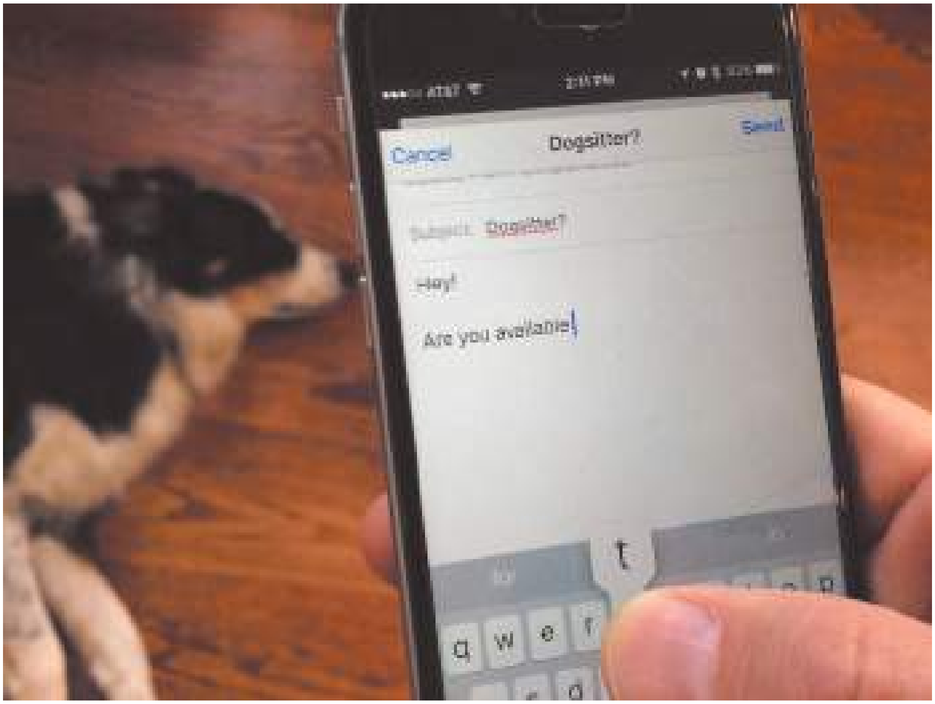


FIGURE 10.6

Soft keyboard on an Apple iPhone with the Shift technique (Vogel and Baudisch, 2007). Shift displaces a pressed or hovered key to a position above the user's finger to reduce the so-called "fat finger" problem, where the finger occludes the touch target. The iPhone keyboard also increases precision by allowing repositioning and then activates on lift-off.

(Fig. 10.6). However, because soft keyboards lack the tangible and tactile feedback of a physical keyboard, they are difficult to use for eyes-free operation and typically yield only modest performance, around 20 to 30 words per minute. Still, one study demonstrated that providing tactile feedback using the phone's vibration motor could improve typing speed (Hoggan et al., 2008), and another study found that expert typists can reach an average of 59 wpm and 90% key accuracy without even seeing a visual representation of the keyboard (Findlater et al., 2011).

Several methods exist to improve text entry on touchscreens. Just like keypad-based text-entry methods can make use of dictionary-based or predictive text-entry algorithms, current touchscreen text-entry methods commonly suggest possible word completions given an input string. More advanced techniques use language models to predict the word the user is trying to write based on the current sentence; such language models are incorporated in new Apple iOS and Android mobile operating systems. Similarly, Swype and ShapeWriter (Zhai and Kristensson, 2003) enable typing by tracing letters using a single touch gesture without the need to lift the user's finger, resolving conflicts using a language model. Finally, while originally designed for pen-based interfaces, the Shift technique has been adopted to mitigate the "fat finger" problem in text entry, for example, by the Apple iPhone, where the user's own finger occludes the key being pressed (Fig. 10.6).

Another text entry method is simply to write by hand on a touch-sensitive surface, typically with a stylus, but character recognition remains error-prone. Contextual clues and stroke speed plus direction can enhance recognition rates, but successful gestural data-entry methods are based on simplified and more easily recognizable character sets, such as the unistrokes used by Graffiti® for Palm OS devices. Another promising method is to allow shorthand gesturing on a keyboard instead of tapping on a touchscreen keyboard, using shapes that match the tapping patterns. Long-term studies confirm that it is possible to achieve good text-entry performance with this technique (Kristensson and Denby, 2009).

For some languages, such as Japanese or Chinese, handwriting recognition has the potential to dramatically increase the number of potential users. On the other hand, users with disabilities, older adults, and young children may not have the necessary fine-motor control to use such interfaces on tiny touch-sensitive surfaces. For them, innovations such as EdgeWrite (Wobbrock et al., 2003) might be helpful. EdgeWrite relies on the use of a physical border to frame the drawing area and uses a modified character set that can be recognized by identifying the series of corners being hit instead of the pattern of the pen stroke, resulting in higher accuracy for all users compared with Graffiti. The EdgeWrite character set has also been used successfully with trackballs or eye-trackers to address the needs of users with disabilities (Wobbrock et al., 2008).

Finally, the proliferation of so-called smartwatches (Fig. 10.23) has led to new research efforts on making text entry practical even on such devices. Of course, the challenge with smartwatches is that their effective display and input area is approximately on the order of 1 inch (about 2 to 3 cm). ZoomBoard (Oney et al., 2013) is one approach that uses iterative zooming to make impossibly tiny keys usable on a small display; other methods exist or are likely forthcoming.

10.3 Pointing Devices

The new generation of touch displays invites users to tap, drag, and pinch the images on the screen directly. Furthermore, with complex information displays such as those found in computer-assisted design tools, drawing tools, or air-traffic-control systems, it is often convenient for the user to point at and select items. This direct-manipulation approach (see Chapter 7) is attractive because the users can avoid having to learn commands, reduce the chance of typographic errors on a keyboard, and keep their attention on the display. The results are often faster performance, fewer errors, easier learning, and higher satisfaction. Pointing devices are also important for small devices and large wall displays that make keyboard interaction less practical.

The diversity of tasks, the variety of devices, and the strategies for using them create a rich design space (Hinckley and Wigdor, 2011). There are many ways to categorize pointing devices, such as physical device attributes (rotation or linear movement), number degrees of freedom (horizontal, vertical, yaw, pitch, etc.), and positioning (relative or absolute). The description below focuses on tasks and degree of directness as organizing dimensions.

10.3.1 Pointing tasks and control modes

Pointing devices are useful for seven types of interaction tasks:

1. *Select.* Choosing from a set of items. This technique is used for traditional menu selection, the identification of objects of interest, or marking an object in a slide deck.
2. *Position.* Choosing a point in a one-, two-, three-, or higher-dimensional space. Positioning may be used to place shapes in a drawing, to place a new window, or to relocate a block of text in a figure.
3. *Orient.* Choose a direction in a two-, three-, or higher-dimensional space. The direction may rotate a symbol on the screen, indicate a direction of motion, or control the operation of a device, such as a robot arm.
4. *Path.* Define a series of positioning and orientation operations. The path may be realized as a curving line in a drawing program, a character to be recognized, or the instructions for a cloth-cutting or other type of machine.
5. *Quantify.* Specify a numeric value. The quantify task is usually a one-dimensional selection of integer or real values to set parameters, such as the page number in a document, the velocity of a vehicle, or the music playback volume.
6. *Gesture.* Perform an action by executing a pre-defined motion. Examples of gestures include dwelling on an object to bring up a context menu, swiping to the left (or right) to turn a page forward (or backward), and pinching (or separating) your fingers to zoom out (or in).
7. *Text.* Enter, move, and edit text in 2-D space. The pointing device indicates the location of an insertion, deletion, or change; see Section 10.2 for details on text-entry devices. More elaborate text tasks include centering, setting margins and font sizes, highlighting (boldface or underscore), and page layout.

While all of these tasks can be performed using a keyboard by specifying directions, coordinates, and distances using a command language, this is an indirect and inefficient method that requires training. In the past, the keyboard was used for all of these purposes, but now most users employ pointing devices to perform the tasks more rapidly and with fewer errors; expert users can further improve performance by using keyboard shortcuts for tasks that are invoked frequently (e.g., Ctrl-C followed by Ctrl-V to copy and paste).

Pointing devices can be grouped into those that offer *direct control* on the screen surface, such as the touchscreen or stylus, and those that offer *indirect control* away from the screen surface, such as the mouse, trackball, joystick, graphics tablet, or touchpad. Within each category are many variations, and novel designs emerge frequently (Box 10.1).

BOX 10.1

Pointing devices.

Direct control devices (easy to learn and use, but hand may obscure display)

- Touchscreen (single- and multi-touch)
- Stylus (passive and active)

Indirect control devices (take time to learn)

- Mouse
- Trackball
- Joystick
- Pointing stick (trackpoint)
- Touchpad
- Graphics tablet

Novel devices and strategies (for special purposes)

- Bimanual input
- Eye-trackers
- Sensors (accelerometer, gyroscopes, depth cameras)
- 3-D trackers
- Data gloves
- Haptic feedback
- Foot controls
- Tangible user interfaces
- Digital paper

Criteria for success

- Speed and accuracy
- Efficacy for task
- Learning time
- Cost and reliability
- Size and weight

Another way to think about pointing devices is whether they use absolute or relative input. Touchscreens, graphics tablets, and eye-trackers also use an input model where the input (motor) space is directly mapped to the output (visual) space. This is called *absolute input*, since one point in motor space corresponds to one point in visual space. *Relative input*, on the other hand, deals with translations (and rotations) from a current position and includes devices such as the mouse, joystick, and trackball. While this distinction is not used in the below discussion, the reader may want to think about each device in absolute versus relative terms as well.

10.3.2 Direct-control pointing devices

Touchscreens are the canonical direct control pointing devices and allow users to interact directly with the visual content of the screen by touching it with their fingers. Because of their natural *affordance*, i.e. their form inviting appropriate action, touch-enabled screens are often integrated into applications directed at novice users in which the keyboard can be eliminated and touch is the main interface mechanism.

Early touchscreen implementations had problems with imprecise pointing, as the software accepted the touch immediately (the land-on strategy), denying users the opportunity to verify the correctness of the selected spot. These early designs were based on physical pressure, impact, or interruption of a grid of infrared beams. High-precision designs dramatically improved touchscreens. The resistive, capacitive, or surface-acoustic-wave hardware often provides up to 1600 × 1600 pixel resolution, and the so-called lift-off strategy enabled users to point at a single pixel. This lift-off strategy has three steps: Users touch the surface and then see a cursor that they can drag to adjust its position; when they are satisfied, they lift their finger off the display to activate.

High-precision touchscreens have transformed mobile devices (Section 10.3.6), such as tablets and phones, to the point that it has become natural for users to be able to directly point to objects on the mobile display. Combined with device miniaturization, where users are perpetually asking for mobile devices to become smaller, lighter, and more powerful, touch computing has led to current mobile devices consisting almost entirely of a touchscreen. In fact, researchers are investigating ways to increase the input and output surface of mobile devices, either by using the back of the device (Baudisch and Chu, 2009) or by appropriating the surrounding physical world using projectors and input sensors (Harrison, 2010). However, pointing using the user's own fingers is prone to the aforementioned "fat finger" problem (Section 10.2.4), where the user's hand and fingers occlude on-screen content. New techniques such as Shift (Vogel and Baudisch, 2007) and occlusion-aware interfaces (Vogel and Balakrishnan, 2010) try to remedy this by displacing the screen content based on the user's touch interaction.

Another way to avoid the fat finger problem is to use a *stylus*, which has a familiar and comfortable feel for most users while simultaneously minimizing hand-screen occlusion. These advantages, however, must be balanced against the need to pick up and put down the stylus. Most stylus interfaces (also called “pen-based interfaces”) are based on touchscreen technology; users can write with a stylus for more natural handwriting and increased motion control but can also use a finger for quick selection (Vogel and Baudisch, 2007). In fact, common capacitive touchscreens, which form the majority of today’s tablets and smartphones, can be interacted with using a low-cost blunt-tipped stylus with a capacitive tip. Results show that even such inexpensive stop-gap measures improve accuracy and performance for drawing and sketching tasks on standard touchscreens (Badam et al., 2014). However, using a stylus on a standard touch display may result in unintentional touches if users rest their hands on the display; such a situation calls for *palm rejection* techniques that discard interaction resulting from the hand based on shape or on the timing of finger and stylus input. There is also risk of losing the stylus.

Beyond mobile devices, the availability of high-precision touchscreens has opened the door to many professional applications in banking, medical, or military systems. Furthermore, because touchscreens can be made to be very robust, they are particularly appropriate for public-access kiosks and mobile applications. Designers of public-access systems value touchscreens because there are no moving parts and durability in high-use environments is good (the touchscreen is the only input device that has survived at Walt Disney World® theme parks). Strategies have been described to provide access to touchscreen systems, such as for information kiosks or voting systems for people who are vision-impaired or blind, are hard of hearing or deaf, have trouble reading or are unable to read at all, or have physical disabilities (Vanderheiden et al., 2004). For kiosk designs, arm fatigue can be a problem, which can be addressed by tilting the screen and providing a surface on which to rest the arm. On the other hand, kiosks are generally not used for extensive interactive sessions. In general, arm fatigue for mid-air or unsupported interaction can be measured using the Consumed Endurance metric, which is based on a biomechanical model of the arm (Hincapié-Ramos et al., 2014).

10.3.3 Indirect-control pointing devices

Indirect pointing devices separate the input (motor) space from the output (display) space, thus minimizing hand fatigue, by providing a surface for the hand to rest as well as eliminating hand-screen occlusion, by keeping the spaces apart. However, they require the hand to locate the device and also demand more cognitive processing and hand/eye coordination to bring the on-screen cursor to the desired target.

The *mouse* is the most common indirect pointing device and is appealing because of its low cost and wide availability. While using a mouse, the hand rests in a comfortable position, buttons on the mouse are easy to press, long motions can be done rapidly by moving the forearm, and positioning can be done precisely with small finger movements. However, users must grab the mouse to begin work, desk space is consumed to operate it, and users must separate their attention between the motor and display space. Other problems are that pick-up and replace (also called *clutching*) actions are necessary for long motions and some practice is required to develop skills (usually from 5 to 50 minutes, but sometimes much more for older adults or users with disabilities). The variety in terms of mouse technologies (physical, optical, or acoustic), number of buttons, placement of the sensor, weight, and size indicates that designers and users have yet to settle on one preferred design. Personal preferences and the variety of tasks to be done leave room for lively competition. The mouse may be simple or may incorporate a wheel and additional buttons to facilitate scrolling, web browsing, or specific applications (Fig. 10.7). Such additional mouse features can sometimes be programmed to perform common tasks of special-purpose applications, such as adjusting the focus of a microscope and switching its magnification level.

The *trackball* is controlled by spinning a ball along two axes and has sometimes been described as an upside-down mechanical mouse. It is usually implemented

FIGURE 10.7

The Apple Magic Mouse 2 wireless mouse on the left has only one button activated by pressing down on the whole mouse (http://www.apple.com). The Razer Ouroboros® gaming mouse on the right has two standard buttons, a center mouse wheel that can also be clicked, and an additional set of nine buttons that can be programmed for specific gaming settings (http://www.razerzone.com/gaming-mice/razer-ouroboros).

FIGURE 10.8
The Logitech Trackman Wheel Optical is a popular trackball device (http://www.logitech.com/).

as a rotating ball, 1 to 15 centimeters in diameter, that moves a cursor on the screen as it is moved (Fig. 10.8). The trackball is wear-resistant and can be firmly mounted in a desk to allow users to hit the ball vigorously and to make it spin. Trackballs have also been embedded in control panels for air-traffic-control or museum information systems, and they are commonly used in video game controllers.

The *joystick*, whose long history began in aircraft-control devices and early computer games, has dozens of versions with varying stick lengths and thicknesses, displacement forces and distances, anchoring strategies for bases, and placement relative to the keyboard and screen. Joysticks are appealing for tracking purposes (to follow or guide an object on a screen), partly because of the relatively small displacements needed to move a cursor, the ease of direction changes, and the opportunity to combine the joystick with additional buttons, wheels, and triggers (Fig. 10.9).

The *directional pad* (or D-pad) originated in game consoles and consists of four directional arrows arranged in a cross with a trigger button in the center. An example is the Wii remote control (left part of Fig. 10.10). This system is also used in mobile devices for navigation in menus. Beyond the Wii, current "eighth-generation" video game consoles also use similar controllers that integrate both joysticks and D-pads; Fig. 10.10 (right) shows the Sony PlayStation 4 DualShock controller.

The *pointing stick* (or *trackpoint*) is a small isometric joystick embedded in keyboards between the letters G, B, and H (Fig. 10.3). It is sensitive to pressure and does not move. It has a rubber tip to facilitate finger contact, and with modest practice, users can quickly and accurately use it to control the cursor while keeping their fingers on the keyboard home position. The pointing stick

FIGURE 10.9

A flight simulator control, combining a joystick (right) and a throttle (left) for two-handed operation.

FIGURE 10.10

The left image shows a Nintendo Wii remote controller (Wiimote, right hand) with attached Nunchuck (left hand). The Wiimote includes a three-axis accelerometer that detects movement in three dimensions. The right image shows a Sony DualShock4 wireless controller for the PlayStation 4 video game console.

is particularly effective for applications such as word processors that require constant switches between the keyboard and the pointing device. Because of their small size, pointing sticks can easily be combined with other devices such as keyboards or even mice to facilitate 2-D scrolling.

Touchpads offer the convenience and precision of a touchscreen while keeping the user's hand off the display surface. Users can make quick movements for long-distance traversals and can gently rock their fingers for precise positioning before lifting off. Often embedded below the keyboard, the touchpad can be used with the thumbs while keeping the hands in typing position. Their lack of moving parts and thin profile make touchpads appealing for laptops. Furthermore, current touchpads often have multi-touch capability, allowing for up to five simultaneous touches; windowing systems such as Apple OS X and Microsoft Windows use this capability for gestures, for example, for scrolling, panning, and zooming a document or graphical view.

The *graphics tablet* is a touch-sensitive surface separate from the screen, usually laid flat on the desk/table or in the user's lap. This separation again allows for comfortable hand positioning and keeps the users' hands off the screen. The

FIGURE 10.11

A digital artist using a Wacom® 13HD Touch graphical tablet with a wireless stylus (http://www.wacom.com/). The Wacom pressure-sensitive stylus and graphics tablet allow the precise pointing and accurate control that artists need.

graphics tablet is appealing when users' hands can remain with the device for long periods without switching to a keyboard. For this reason, graphics tablets are often popular with digital artists who engage in drawing and sketching operations. Furthermore, the graphics tablet permits adding application options, such as palettes, tools, and brushes, beyond the screen itself to its surface, thereby preserving valuable screen space and providing both guidance to novice users as well as easy access to experts. Graphics tablets are typically operated using a finger, pencil, puck, or stylus through acoustic, electronic, or contact position sensing. Artists tend to prefer wireless pens for high precision and freedom (Fig. 10.11).

Among the above indirect pointing devices, the mouse has been the greatest success story. Given its rapid, high-precision pointing abilities and comfortable hand position, the modest training period is only a small impediment to its use. Most desktop computer systems offer a mouse, but the battle for the laptop continues, with many vendors offering multiple pointing devices on a single machine.

10.3.4 Comparison of pointing devices

Early studies found that direct pointing devices such as a touchscreen were often the fastest but the least accurate devices. Decades of studies have consistently shown the merits of the mouse over alternative devices for speed and accuracy. The pointing stick has been found to be slower than the mouse due to tremors during fine finger movements (Mithal and Douglas, 1996). Trackballs and touchpads fall somewhere in between. Users' tasks matter when comparing devices. For example, when browsing the web, users are constantly involved in both scrolling and pointing. One study showed that a mouse with a finger wheel did not improve users' performance over a standard mouse. Future research might provide a better understanding of the benefits and limitations of each device.

Common wisdom states that pointing devices are faster than cursor-movement keys for selecting objects, but this assertion depends on the task. When a few (2 to 10) targets are on the screen and the cursor can be made to jump from one target to the next, the cursor keys can be faster than using pointing devices. For short distances and for tasks that mix typing and pointing, cursor keys have also been shown to be faster than and preferred to the mouse. However, many users never learn keyboard shortcuts (e.g., Ctrl-Z to undo), despite the fact that menu selections can be performed much faster using those shortcuts than by using a pointing device (Grossman et al., 2007).

Users with motor disabilities often prefer joysticks and trackballs over mice, as the location of such devices remains fixed, they have a small footprint (allowing them to be mounted on wheelchairs), and they can be operated by small residual movements. On the other hand, touch-sensitive devices are useful when applying force is difficult—for instance, for users with motor disabilities—but designers should attempt to detect inadvertent or uncontrolled movements and smooth out trajectories. Using active target areas that are larger than the button or icon to be selected is effective to shorten selection time and reduce frustration for every user and, in some cases, might be all that is needed to render an application usable by a much wider audience.

Pointing devices are extremely challenging for users who have vision impairments. Well-designed cursors of adjustable size and shape may help users with limited vision impairments, but indirect-control devices such as the mouse are simply not practical for users with severe vision impairments who have to rely on the keyboard. Alternative keyboard or keypad navigation options should be provided whenever possible. Touchscreen interfaces can more easily be explored and memorized when speech synthesis or sonification is available to describe the display, read menu options, and confirm selections. For example, in a touchscreen voting kiosk, users can use arrow keys to navigate through lists of candidates whose names are read aloud via headphones (Fig. 10.12). Successful examples demonstrate that it is possible to design powerful systems that are truly accessible to the general public, including users with a wide range of disabilities (Vanderheiden et al., 2004). Finally, *tactile graphics* can be produced by using thermal paper expansion machines and placed on top of touchscreens for use by blind users.

In summary, individual differences and the user tasks are critical when selecting a pointing device. The touchscreen and trackball are durable in public-access, shop-floor, and laboratory applications. The mouse, trackball, trackpoint, graphics tablet, and touchpad are effective for pixel-level pointing. Pens and styli are appreciated for drawing and handwriting, and simple gestures can be used to specify actions and quantify their parameters. Cursor jump keys remain attractive when there are a small number of targets. Joysticks are appealing for games or specialized navigation applications.

10.3.5 Fitts's Law

One of the few scientific models of human-computer interaction is Paul Fitts's (1954) law of human hand movement. Often referred to simply as Fitts's Law (or even Fitts' Law), this micro-scale HCI theory allows designers to decide on the optimal locations and sizes of buttons and other elements when laying out screens as well as indicates which pointing devices are best suited to performing common tasks. Fitts noticed that the time required to complete hand movements was dependent on the distance users had to move, *D*, and the target size, *W*. Doubling the distance (say, from 10 cm to 20 cm) resulted in longer completion times, but not twice as long. Increasing the target's size (say, from 1 cm^2 to 2 cm^2) enabled users to point at it more rapidly.

FIGURE 10.12
Users of this touchscreen voting tablet need only touch any text on the screen to have it read aloud, with the sound communicated to them via headphones. Touching the check box marks the vote, with verbal confirmation if headphones are used. Users who are completely blind or have severe physical disabilities that prevent them from using the touchscreen (even with voice) can use a detachable keypad—with or without voice. The keypad also allows connection of custom switches voters bring with them (http://www.trace.wisc.edu/).

Since the time to start and stop moving is constant, an effective equation for the movement time (*MT*) for a given device, such as a mouse, turns out to be

$$MT = a + b \log_2(D/W + 1)$$

where a approximates the start/stop time in seconds for a given device and b measures the inherent speed of the device. Both a and b need to be determined experimentally for each device. For example, if a were 300 milliseconds, b were

FIGURE 10.13

A blind student uses a Touch Graphics tactile map mounted on a touchscreen. On the right, a different embossed overlay is used to learn the location of states and capitals, with a pen providing audio descriptions (http://www.touchgraphics.com/).

200 msec/bit, D were 14 cm, and W were 2 cm, then the movement time MT would be $300 + 200 \log_2(14/2 + 1)$, which equals 900 milliseconds.

Several versions of Fitts's Law are used, but this equation has been demonstrated to provide accurate predictions in a wide range of situations. The variations are due to differences such as the direction of motion (horizontal or vertical), device weight (heavier devices are harder to move), device grasp, shape of targets, and arm position (on a table or in the air). MacKenzie (2013) lucidly describes what Fitts's Law is, how it has been applied, and refinements for cases such as 2-D pointing. Studies of high-precision touchscreens have shown that in addition to the gross arm movement predicted by Fitts, there was also a fine-tuning motion of the fingers to move in on small targets such as single pixels. A three-component equation was thus more suited for the precision-pointing movement time (*PPMT*):

$$PPMT = a + b \log_2 (D/W + 1) + c \log_2(D/W).$$

The third term, time for fine tuning, increases as the target width, W, decreases. This extension to Fitts's Law is quite understandable; it suggests that the precision-pointing movement time consists of the start/stop time (a), a time for gross movement, and a time for fine adjustment. Other studies deal with a greater range of arm motion with pointing in 3-D space or with two-thumb text entry. Alternative extensions (e.g., Chapuis and Dragicevic, 2011) focus on other aspects of the model.

Fitts's Law is well-established for adult users, but it may need refinements for special populations such as young children or older adults. In one study, 13 four-year-olds, 13 five-year-olds, and 13 young adults performed point-and-click selection tasks (Hourcade et al., 2004). As expected, age had a significant effect

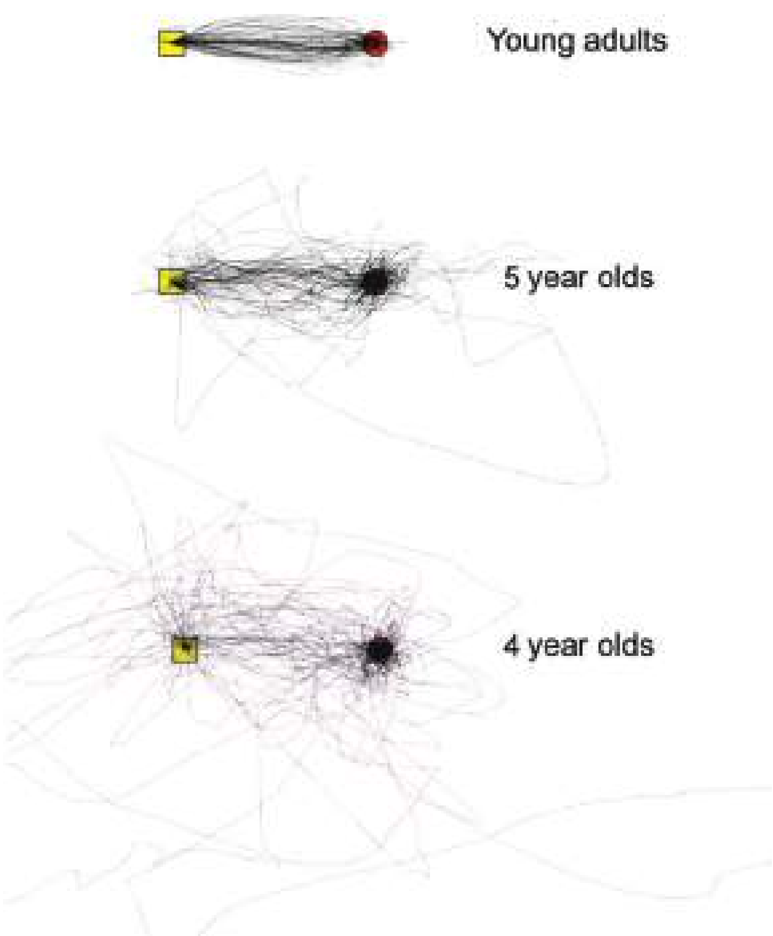


FIGURE 10.14

Tracing the trajectory of the pointer during a repeated target-selection task illustrates the dramatic difference between children's and young adults' use of the mouse (Hourcade et al., 2004).

on speed and accuracy (and of course trajectories, as shown in Fig. 10.14). A detailed analysis showed that Fitts's Law models children well for the first time they enter the target, but not for the time of final selection.

The open problem remains: how to design devices that produce smaller constants for the predictive equation, in effect "beating" Fitts's Law (Balakrishnan, 2004). One study has shown that multi-scale pointing with zooming works best with two-handed input and a constant zoom speed (Guiard et al., 2001). Another study looked at crossing-based interfaces in which targets are merely crossed (e.g., as in how a finish line is crossed) instead of pointed at. The target-crossing completion time was found to be shorter than or equal to pointing performance under the same index of difficulty and depended on the type of task performed (Accot and Zhai, 2002). The quest for faster selection times continues.

10.3.6 Novel pointing devices

The popularity of pointing devices and the quest for new ways to engage diverse users for diverse tasks have led to provocative innovations. Improving the match between the task and the device and refining the input plus feedback strategies are common themes (Kortum, 2008).

Bimanual input—input using two hands simultaneously—can facilitate multitasking or compound tasks. In such settings, the nondominant hand sets a frame of reference in which the dominant hand operates in a more precise fashion. A natural application of bimanual operation for desktop applications is that the nondominant hand selects actions (for example, the Fill command of a paint program) while the dominant hand precisely selects the objects of the operation (see Chapter 8 for more on supporting navigation).

Since users' hands might be busy on the keyboard, designers have explored alternative methods for selection and pointing. *Foot controls* are popular with rock-music performers, organists, dentists, and car drivers, so maybe computer users could benefit from them as well. A foot mouse was tested and was found to take about twice as much time to use as a hand-operated mouse, but benefits in special applications may exist—for example, switches and pedals activated by foot might be effective to specify modes.

Eye-trackers are gaze-detecting controllers that use video-camera image recognition of the pupil position to give one- or two-degree accuracy (Figs. 5.3 and 5.4). Fixations of 200 to 600 milliseconds are used to make selections. Unfortunately, eye-tracking easily results in the "Midas touch" problem, since every gaze has the potential to activate an unintended command. Combining eye-tracking with manual input is one way to address this problem (Stellmach and Dachselt, 2013), but for now, eye-tracking remains mostly a research and evaluation tool (Section 5.3.1) and a possible aid for users with motor disabilities (Wobbrock et al., 2008).

Multiple-degree-of-freedom devices can sense multiple dimensions of spatial position and orientation. Control over 3-D objects seems a natural application, but comparisons with other strategies reveal low precision and slow responses. Support for virtual reality (Chapter 7) is one motivation, but many design, medical, and other tasks may require 3-D input or even six degrees of freedom to indicate a position and an orientation. Commercial tracking devices include the Logitech company's 3Dconnexion®, Ascension®, Intersense®, and Polhemus™.

Ubiquitous computing and *tangible user interfaces* (Dourish and Bell, 2011) depend on embedding sensing technologies into the environment. For example, active badges with Radio Frequency Identification (RFID) tags can trigger the preloading of personal files into a room's computer when users enter a room. The positioning of physical objects can specify modes or trigger actions. Ambient light, sound, or airflow can be modified to present a small amount of information to users. Entertainment and artistic applications use video cameras or body sensors to track body positions and create enticing user experiences. Early explorations by performance artist Vincent John Vincent led to 3-D environments for theatrical exploration such as Mandala, in which performers or amateur users touch images of harps, bells, drums, or cymbals, and the instruments respond. Myron Krueger's artificial realities contain friendly video-projected cartoon-like creatures that playfully crawl on your arm or approach your outstretched hand. Such environments invite participation, and the serious research aspects fade as joyful exploration takes over and you step inside the computer's world (Section 7.6). StoryRoom is another such application that enables children to actively construct their own interactive environments, using props and magic wands to create stories that other children are invited to experience (Montemayor et al., 2004).

Even *paper* can be used as an input device. Early applications demonstrated the benefits of capturing annotations on large documents such as sketchbooks, blueprints, or lab notebooks. Pens such as the Livescribe™ 3 stylus (Fig. 10.15) with Anoto® functionality facilitate interaction, particularly in mobile situations. The pen has a small camera in its tip, and it records pen strokes drawn on a special paper printed with a unique pattern that identifies the location of each stroke. The handwriting can then be transferred to a computer or a mobile phone. The

FIGURE 10.15

The Livescribe 3 smartpen with Anoto technology records the strokes of ink written on augmented paper (paper with a unique dot pattern), and the data are transferred wirelessly to the tablet. A microphone allows users to record audio at the time an annotation is being made to be replayed later via the embedded speaker by tapping on the annotation (http://www.livescribe.com/).

ease of learning might help novice users: Paper Augmented Digital Documents (PADDs) can be edited both in digital and paper form (Liao et al., 2008), or translations can be requested by writing words on paper (Fig. 10.15).

Mobile devices can also be used as input devices. For example, Carnegie Mellon University's Pebbles project (Myers, 2005) and University of Maryland's PolyChrome project (Badam and Elmqvist, 2014) explored how mobile devices can be used to communicate with personal computers, other mobile devices, large displays, home appliances, automobiles, or factory equipment. Mobile devices can act as intelligent universal remote controls, potentially empowering all users by reading aloud product information or menu options, translating instructions written in a foreign language, or offering speech recognition when needed. Finally, the images captured by cameras of mobile phones can be used as input to augmented-reality applications (Rohs and Oulasvirta, 2008) or even to control general apps (Hansen et al., 2006).

Sensors mounted in the environment or added to handheld devices can enrich the interaction with the devices themselves. For example, *accelerometers* allow the Apple iPhone to detect changes in the device's orientation, causing the display to switch dynamically between portrait and landscape orientations. Similar functionality exists in most digital cameras, allowing them to automatically determine whether a picture should be shown in landscape or portrait orientation. As users become familiar with gesture interaction, designers may be able to find additional natural uses of movement information. For example, users may be able to zoom and pan on a map by adjusting the proximity or lateral position of mobile devices in front of them. Tilting the device could scroll through a list of names, and bringing the device near the ear could answer an incoming call. The remote control of the Nintendo Wii video game console includes a three-axis accelerometer that can detect movement in three dimensions and respond to gestures. For example, to hit a tennis ball, users swing the controller like a racket with a realistic arm motion. The Wii has inspired many applications that require users to be more active and has successfully attracted more female and older adult users to video games (Fig. 10.10). Finally, commodity *depth cameras*, such as the Xbox Kinect, allow for capturing the motion of the user's entire body using a combination of standard and infrared cameras. Recent work has explored how to use this capability to reconstruct the user's hand pose for more refined touch computing (Murugappan et al., 2012).

Specialized *hand sensors* may be able to determine the exact posture and position of each finger joint in a user's hand. The Leap Motion controller uses infrared cameras to track a user's hand at 200 frames per second in a volume above the sensor without the need for specialized markers or even touching the sensor (Fig. 10.16). This enables detecting the hand position and posture at high accuracy. Similarly, the VPL *DataGlove* appeared in 1987 and attracted researchers, game developers, cyberspace adventurers, and virtual reality devotees. Descendants

FIGURE 10.16

The Leap Motion controller tracks the user's hand in three dimensions using infrared cameras, precisely reconstructing the posture of each finger (http://www.leapmotion.com/).

of the original data glove have attached fiber-optic sensors to measure angles of finger joints (Fig. 7.13). The displayed feedback can show the relative placement of each finger; thus, commands such as a closed fist, open hand, index-finger pointing, and thumbs-up gesture can be recognized. Combined with a hand-tracker, complete 3-D placement and orientation can be recorded. Devotees claim that the naturalness of gestures will enable use by many keyboard-averse or mouse-phobic users. But while the simple gestures of the Nintendo Wii games are easily learned because their number is small and they mimic the actions being simulated (e.g., the swing of a golf club or the rotation of a wheel), data glove and Leap Motion users require substantial training to master more than half a dozen gestures, especially if they do not map to naturally occurring gestures. Still, gestural input with the glove can make special applications possible, such as the recognition of American Sign Language or virtual musical performances.

Pointing devices with *haptic feedback* are an intriguing research direction (Kortum, 2008). Several technologies have been employed to allow users to push a mouse or other device and to feel resistance (for example, as they cross a window boundary) or a hard wall (e.g., when navigating a maze). 3-D versions, such as SensAble Technology's PHANTOM, are still more intriguing, but commercial applications are slow to emerge. Because sound and vibrations are often a good substitute for haptic feedback, the use of advanced haptic devices remains limited to special-purpose applications (such as training surgeons for heart surgery), while devices using simple vibrations have become mainstream in game controllers and even the common mouse.

Finally, a recent development in pointing has been the introduction of *hybrid pointing techniques* that combine several of the above techniques. This development goes hand in hand with the increasing prevalence of large as well as small displays, since pointing on such devices breaks from the standard; it is simply not practical to use a mouse or even your own finger touches on a large display, but this may require distant pointing. At the same time, such distant pointing using laser pointers or even the user's own gaze or finger is error-prone. New work that successfully integrates both small and large displays achieves this by combining direct and indirect pointing (Forlines et al., 2006; McCallum and Irani, 2009). Further advanced pointing techniques that use novel sensors will likely appear in the future, such as head or shoulder movements, a light blow of air in a tube, the blink of an eye, and even faint myoelectric currents generated when muscles are being tensed. Such reality-based (Jacob et al., 2008) and "natural" (Wigdor and Wixon, 2011) interaction mechanisms are indicative of a broader trend of going beyond the traditional mouse and keyboard for the new generation of computing devices.

10.4 Displays

The display is the primary source of visual feedback to users from the computer. It has many important characteristics, such as:

- Physical dimensions (usually the diagonal dimension and depth)
- Resolution (the number of pixels available)
- Number of available colors and color correctness
- Luminance, contrast, and glare
- Power consumption
- Refresh rates (sufficient to allow animation and video)
- Cost
- Reliability

Usage characteristics also distinguish display devices. Portability, privacy, saliency (need to attract attention), ubiquity (likelihood of being able to locate and use the display), and simultaneity (number of simultaneous users) can be used to describe displays (Raghunath et al., 2003). For example, mobile phones are perceived as personal devices and provide portable and private displays, whereas large displays allow social interactions between, for example, multiple users controlling characters in video games. Similarly, salient information displays found in malls or museums might offer store location information to a single user or an emotional theatrical experience to dozens of impressed visitors. Whiteboard displays allow collaborators to share information, brainstorm, and make decisions. Finally, immersive displays can transport a user into an imaginary world for recreation or to learn a new skill.

10.4.1 Display technology

The classic *raster-scan cathode-ray tubes* (CRTs) have now mostly vanished, replaced by *liquid-crystal displays* (LCDs) with their thin form, light weight, and low electricity consumption. Like LCDs, *plasma displays* have a flat profile, but they consume more electricity. They are very bright and readable even from the side, making them valuable for mounted wall displays in control rooms, public displays, or conference rooms. *Light-emitting diodes* (LEDs) are now available in many colors and are being used in large public displays. Matrices of miniature LEDs are also used in some head-mounted displays. Manufacturers are actively developing new displays using organic light-emitting diodes (OLED). These durable organic displays are energy-efficient and can be laid on flexible plastic or metallic foil, leading to new opportunities for wearable or rollable displays.

New products attain paper-like resolution using *electronic ink* technology. They contain tiny capsules in which negatively charged black particles and positively charged white particles can be selectively made visible. Because electronic ink displays use power only when the display content changes, they have an extended battery life over other types of displays and are well-suited for ebooks (e.g., Amazon's Kindle™, shown in Fig. 10.17, the Barnes & Noble Nook, or Bookeen's Cybook). Slow display rates allow some animation but no video displays.

Tiny projectors, so-called pico projectors (Dachselt et al., 2012), are able to project color images on the wall from mobile devices and make collaboration using those devices more practical. *Braille displays* for blind users provide up to 80 cells, each displaying a character. A couple of cells can be mounted on a mouse, and small displays can fit above the keyboard. Prototypes of refreshable graphic displays with up to several thousand pins are being developed. Manufacturers and government agencies are addressing health concerns relating to the different types of visual displays, such as visual fatigue, stress, and radiation exposure. Adverse effects seem for the most part attributable to the overall work environment more than the visual display units themselves.

FIGURE 10.17

An Amazon Kindle (http://www.amazon.com/) book reader being used to browse *Bleak House* by Charles Dickens. The Kindle uses E-Ink® technology (http://www.eink.com/), providing a bright display that uses power only when the display changes, and can be read in direct sunlight and at varying angles, which can improve reading comfort (see Section 14.4 for a discussion of reading on paper versus on a display).

10.4.2 Large wall displays

The ubiquity of computer displays, from desktops to mobile devices, projectors, and large televisions, lets us envision how integrating all those displays could provide more productive work and play environments (Ardito et al., 2015). The differentiation might fade in the future, but there are currently three types of large wall displays. *Informational wall displays* provide shared views to users standing far away from the display, while *interactive wall displays* allow users to walk up to the display and interleave interaction and discussion among participants. Finally, users sitting at their desks can connect *multiple-desktop displays* to their computers to have a larger number of windows and documents visible at the same time and within reach of the mouse. Of course, hybrid combinations are possible (see, for example, Badam and Elmqvist, 2014).

Large informational wall displays are effective in control rooms to provide overviews of the system being monitored (Fig. 10.18); details can be retrieved on individual consoles. Military command and control operations, utility management, and emergency response are common applications, as large displays

FIGURE 10.18

Multiple high-resolution displays tiled together to present weather, traffic, message sign status, and road-condition information to the operators in the State of Maryland's Highway Administration control room (http://www.chart.state.md.us/).

help to build situation awareness through a common understanding of the information presented and to facilitate coordination. Wall displays also allow teams of collaborating scientists or decision makers to look at applications that may be running on different computers, locally or remotely, but are presented on a single display.

Originally built with matrices of CRTs and made popular in commercial or entertainment settings, wall displays now often use rear-projection techniques or tiled LCD displays. Rear projection has become popular because improved calibration and alignment techniques allow for seamless displays. When seen from a distance, informational wall displays require bright projectors, but the resolution does not need to be very high—35 dots per inch is sufficient. However, when users want to view or interact with the display at close range, the low resolution of current projectors is often insufficient. Furthermore, rear projection requires a large amount of space for the throw distance of each projector. In such situations, tiled displays consisting of a grid of LCD monitors may be the most cost-effective solution (extremely large LCD displays are still prohibitively expensive to build) even if the display bezels yield visible seams between each tile. The Stony Brook University Reality Deck

FIGURE 10.19
Users discussing and pointing at details on the Stony Brook University Reality Deck (Papadopoulos et al., 2015), an immersive gigapixel display consisting of 416 thin-bezel LCD displays and powered by 18 graphics workstations connected using a high-speed network (https://labs.cs.sunysb.edu/labs/vislab/reality-deck-home/).

(Fig 10.19) is an example of an extreme-scale tiled LCD display consisting of more than 400 individual high-resolution screens (Papadopoulos et al., 2015). Regardless of the technical solution, these multi-display environments typically need specialized software frameworks to distribute the rendering and merge input across multiple computers; PolyChrome is an example of such a framework (Badam and Elmqvist, 2014).

For interactive wall displays (Figs. 10.18, 10.19, and 10.20), the traditional desktop interaction techniques, such as indirect-control pointing devices and pull-down menus, become impractical. New techniques are being devised to maintain fluid interaction with freehand sketching or novel menu techniques (Chapter 8). Even on large interactive group displays, space is limited and designers are exploring new ways to dynamically scale, summarize, and manage the information presented or generated by users on the display.

Digital whiteboard systems, such as the SMART Board® from SMART® Technologies, Inc., provide a large touch-sensitive screen on which a computer image is projected. Their functionality is identical to that of the desktop machine, using

FIGURE 10.20

Two users collaboratively control a lens on a gigapixel image of Paris, France, using a tablet touchscreen as well as an interactive cursor (Chapuis et al., 2014). Photo © Inria - H. Raguet, with permission.

users' fingers as pointing devices. Colored pens and a digital eraser simulate a traditional whiteboard, augmented with annotation recording and a software keyboard.

Facilitating collaboration between local or remote users (Chapter 11), managing the recording and reuse of brainstorming information, providing new creative tools for artists and performers, and designing new interaction methods using mobile devices are examples of challenges and opportunities created by interactive wall displays.

Multiple-desktop displays usually employ traditional flat panels, which introduce discontinuities in the overall display surface (Fig. 10.21). Those displays can also be of different size or resolution, adding the possibility of misalignments. On the other hand, users can continue interacting with applications in the familiar way, eliminating training as users simply spread windows across displays. Another concern is that multiple-desktop displays may require users to stand, or at least rotate their heads or bodies, to attend to all displays, and even attentive users might not notice warnings or alarms that are far from their foci of attention. Organized users might assign displays to particular functions (for example, the left-hand display may always show e-mail and calendar applications and the

FIGURE 10.21
Analyst interacting with capital markets data on a six-monitor Bloomberg Terminal. One of approximately 320,000 units, these terminals are leased to clients on a subscription basis at approximately $20,000 per user (http://www.bloomberg.com/professional/products-solutions/). Note the non-overlapping user interface and the specialized keyboard.

front display always a word processor), but this strategy can be detrimental when the current task would benefit from using the entire display space.

Multiple-desktop displays are particularly useful for personal creative applications. For example, creating an interactive web application in JavaScript might require a timeline, a stage, graphic-component editors, a scripting-language editor, a directory browser, and a preview window, all open at the same time. Multiple-desktop displays might also facilitate side-by-side comparisons of documents, software debugging, or reasoning based on a large number of information sources. They are usually greatly appreciated by users, and empirical evidence of their benefits is emerging (Andrews et al., 2011).

Of course, there is a danger that cluttered displays will become more cluttered as their physical size grows. Also, direct manipulation on large displays can become a challenge because of the distance between objects. On the other hand, results show that having access to large displays even for individual use makes reasoning easier because the user can intelligently organize information in space (Andrews et al., 2011). Innovations can mitigate many of these challenges. Refinements should be made so that the mouse cursor can easily be found and tracked across displays. Rapid focus switching between windows might be facilitated by clicking on small overviews placed at strategic locations on the display. Furthermore, strategies for automatic window layout and coordination among windows will become critical (refer to Section 12.3). Nevertheless, as the cost of displays continues to drop, it seems clear that multiple-desktop displays or simply larger single displays will become prevalent.

10.4.3 Tabletop (horizontal) displays

While wall displays promote coordination and consensus, horizontal surfaces have been shown to invite collaboration and discussion (Rogers and Lindley, 2004). For this reason, *tabletop displays* have become an interesting platform for deeply collaborative settings, such as for creative design, problem solving, or real-time resource management and planning. Such digital tabletops are generally equipped with *multi-touch touchscreens*, which allow a single user to use both hands or multiple fingers at once or allow multiple users to work together on a shared surface. The Microsoft Surface, Surface 2, Perceptive Pixel, and SurfaceHub are examples of such devices. Circle Twelve's DiamondTouch™ display allows the application to tell which user touched the screen, allowing better identification of personal versus collaborative interactions. With horizontal displays, users can be positioned anywhere around the table, so applications that can be used in any orientation are desirable (Fig. 10.22). Physical objects might also be used to mark positions and facilitate design sessions. Using stereoscopic displays, volumetric displays, or head-mounted displays, it may become possible to design effective 3-D tabletop interactions (Grossman and Wigdor, 2007). Furthermore, combining shared tabletops with user's personal mobile devices is also particularly powerful (McGrath et al., 2012).

FIGURE 10.22

Two people collaborating on a real estate task using a tabletop display and mobile tablet (McGrath et al., 2012). The tabletop serves as a shared and public display where changes affect all collaborators, whereas the tablet is perceived as a private display that allows users to work independently.

Multi-touch displays—both in terms of tabletops as well as mobile devices—have ushered in a vocabulary of simple gestures for complex direct manipulation. Such gestures include *pinching*, grasping or spreading apart two fingers (e.g., to zoom out or in on an image), and *dwelling*, holding a finger touch for an extended amount of time (e.g., to bring up a context menu on the selected object).

10.4.4 Heads-up and head-mounted displays

Personal-display technology involves small portable monitors, often made with LCDs in monochrome or color. A *heads-up display* projects information on the partially silvered windscreen of an airplane or car, for example, so that the pilots or drivers can keep their attention focused on the surroundings while receiving computer-generated information.

An alternative, the *head-mounted display* (HMD) used in virtual reality or augmented reality applications (Section 7.6), lets users see information even while turning their heads. In fact, tracking the user's head allows for dynamically changing the information that is seen as a function of the direction in which the user is looking. Different models provide varying levels of visual-field obstruction, audio capabilities, and resolution. Google's Glass project was an attempt at blending HMDs with mobile devices, yielding a wearable computer with a display in the corner of the user's vision. The Oculus Rift and Microsoft HoloLens devices (Chapter 7) represent reimaginings of the HMD concept, again mostly targeted at recreational and non-professional users.

10.4.5 Mobile device displays

The use of mobile devices is becoming widespread in personal and business applications and has the potential to improve medical care, facilitate learning in schools, and contribute to more fulfilling sightseeing experiences. Medical monitors can alert doctors when a patient's life signs reach a critical level, schoolchildren may gather data or solve problems collaboratively using handheld devices, and emergency rescue personnel can evaluate their situation in dangerous environments by using small devices fixed on their suits. Small displays are also finding ways into our homes, with reprogrammable picture frames and other devices, and even onto our bodies, with ever more powerful wristwatches with customization features to fit the needs of the moment. The new generation of so-called "smartwatches," such as the Apple Watch and Fitbit Surge (Fig. 10.23), integrate step counters, heartbeat, and GPS as well as additional advanced functionality such as text, e-mail, calendar, voice recognition, and even electronic payment options into the wristwatch form factor.

FIGURE 10.23

The Apple Watch on the left supports both fitness as well as personal information management applications, such as e-mail, calendar, and electronic payment. The Fitbit Surge smartwatch on the right is designed mainly for personal fitness applications and contains a step counter, heart rate monitor, and GPS.

Guidelines are emerging from experience with mobile devices (Ballard, 2007). Industry has been leading the way by providing useful design case studies and detailed guidelines such as those developed for Android devices or the iPhone. Such guidelines can be seen as micro-HCI theories.

Ballard (2007) sees mobile devices being developed in four classes, depending on their intended usage: (1) general-purpose work (similar to the Blackberry or Pocket PC), (2) general-purpose entertainment (which focus on multimedia features like the Apple iPod), (3) general-purpose communication and control (extensions of today's phones), and (4) targeted devices that do only a few tasks (e.g., the United Parcel Service drivers' DIAD IV). Mobile devices are often used for brief but routine tasks. Therefore, it is critical to optimize the designs for those repetitive tasks while hiding or eliminating less important functions. Whenever possible, data entry should be reduced and complex tasks offloaded to the desktop.

While researchers and developers are steadily increasing the scope of applications for mobile devices, a framework for thinking about the range of actions may be helpful. Whether the application is financial-, medical-, or travel-related, the following five pairs of actions should be considered: (1) *monitor* dynamic information sources and *alert* when appropriate, (2) *gather* information from many sources and *spread out* information to many destinations, (3) *participate* in a group and *relate* to individuals, (4) *locate* services or items that are not visible (for example, the nearest gas station) and *identify* objects that are seen (for example, the name of a person or flower), and (5) *capture* information from local resources and *share* your information with future users.

Poor readability will be an issue in low light or for users with poor eyesight, and users will appreciate the ability to adjust the font size. Reading on small screens might also be improved with *rapid serial visual presentation* (RSVP), which presents text dynamically at a constant speed or at a speed adapted to the content. Using RSVP, although no differences were found for long texts, a 33% improvement was measured in speed of reading for short texts (Öquist and Goldstein, 2003).

Successful interaction design for mobile devices requires adapting content to the display, regardless of its size and capabilities (color depth, resolution, update frequency). This is particularly important for web designers, as people are increasingly using their mobile devices to access the web. *Responsive web design* (RWD) is an approach to creating webpages and web applications so that the result is optimized for whatever device is being used to view a website. Responsive web design can be summarized in a small set of basic principles: (1) *mobile-first design*, rather than desktop first, so that the constraints of limited resources become more apparent to the designer; (2) *unobtrusive dynamic behavior*, so that a website is not wholly dependent on JavaScript; and (3) *progressive enhancement*, where websites are layered with increasingly advanced functionality so that backward compatibility is retained for basic browsers while advanced browsers can take full advantage. Fig. 10.24 shows responsive web design in practice on three different display sizes.

Mobile users often have only one hand available and rely on their thumbs to interact with the devices. Guidelines for mobile interfaces that support one-handed interaction include placing targets close to one another to minimize grip adjustment, allowing users to configure tasks for either left- or right-handed operation, and placing targets toward the center of the device (Karlson et al., 2008).

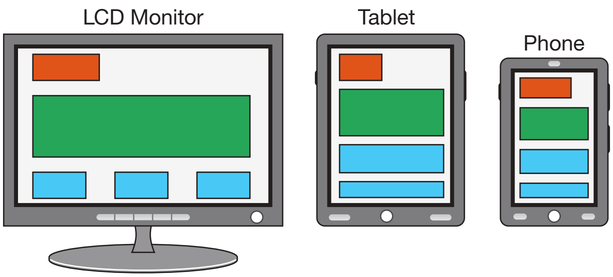


FIGURE 10.24

Examples of responsive web design. The monitor layout on the left is automatically adapted to the smaller display space of a tablet (middle) and a smartphone (right). Also see Chapter 8.

Another challenge facing designers of mobile-device applications is the growing diversity of devices, which may require finding interaction styles that adapt to multiple screen sizes and can be activated by multiple input mechanisms (QWERTY keyboards as well as touchscreens, keypads, or directional pads).

As mobile devices become information appliances, they may contribute to the goal of bridging the digital divide. These low-cost devices, which are easier to master than desktop computers, may enable a wider range of people to benefit from information and communication technologies. Developing countries are seeing a rapid spread of mobile technology, which requires less local infrastructure than providing stable electricity. For users with disabilities, mobile telecommunication devices offer a unique opportunity to design *modality translation services*, as described by a project of the Trace Center at the University of Wisconsin. Remote services can provide instant translation from one presentation mode to another, anywhere and at any time, via mobile devices. This permits text-to-speech, sign language, international language, and language-level translation as well as print recognition and image/video description services. Modality translation could benefit people with disabilities and people who have no disabilities but experience functional limitations when driving their cars, visiting a foreign country, or touring a museum without reading glasses.

10.4.6 Deformable and shape-changing displays

Current displays are flat due to reasons of both manufacturing and tradition, but this will likely not remain the case in the near future. In terms of manufacturing, new technological advances in shape displays, digital fabrication, and programmable matter will soon allow hardware companies to build displays that are of virtually any shape. In terms of tradition, while computer interfaces have long been displayed on planar surfaces, human history, culture, and technology are full of examples of non-planar manifestations of data and content, such as sculptures, statues, tokens, souvenirs, paintings, medals, and mementos. In other words, future displays will not only go beyond the flat plane, but they will also be shape-changing in that they bend, move, and respond to not only virtual but also physical interactions.

Some examples of future non-flat and shape-changing displays are given in Fig. 10.25. Physical visualizations are 3-D embodiments of data graphics rendered on the screen (Jansen and Dragicevic, 2013). They represent examples of static physicalizations (http://dataphys.org/) of data and can often be fabricated using a 3-D printer (which also can be seen as a form of display or output device). Tilt displays challenge the flat LCD surface by mounting multiple small displays on actuators (Alexander et al., 2012). The displays move in response to the data and can also be manipulated themselves, making them also input devices. The PaperPhone is an exercise in flexible displays and is a prototype of a smartphone based on bendable paper (Lahey et al., 2011). Just like the tilt display, the PaperPhone is also an input device with several gestures defined for how to make and

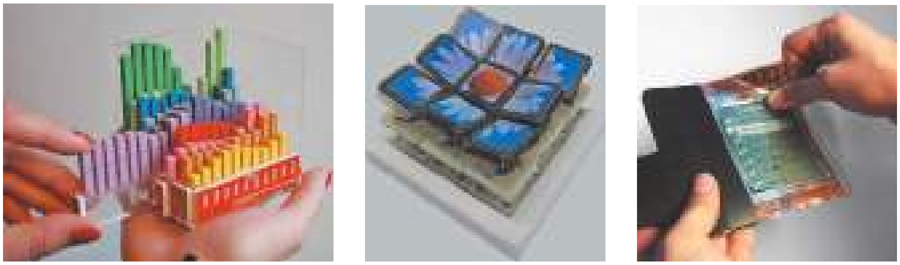


FIGURE 10.25
The left image shows a physical bar chart visualization displaying complex data (Jansen and Dragicevic, 2013). The middle shows the tilt display that consists of multiple small displays mounted on actuators (Alexander et al., 2012). On the right is the PaperPhone, a flexible smartphone prototype that supports bending interaction (Lahey et al., 2011).

take calls, navigate in the address book, and so on. Finally, Fig. 10.26 shows the inFORM dynamic shape display that serves as both output as well as input display for data physicalization (Follmer et al., 2013).

FIGURE 10.26
A telepresence application being used on the inFORM dynamic shape display, an actuated and touch-responsive shape-changing display (Follmer et al., 2013).

Practitioner's Summary

Choosing hardware always involves making a compromise between the ideal and the practical. The designer's vision of what an input or output device should be must be tempered by the realities of what is commercially available within the project budget. Devices should be tested in the application domain to verify the manufacturer's claims, and testimonials or suggestions from users should be obtained.

New devices and refinements to old devices appear regularly; device-independent architecture and software permit easy integration of novel devices. Avoid being locked into one device, as the hardware is often the softest part of the system. Instead, aim to transcend device hardware by focusing on the task rather than the mechanics of performing it. Also, remember that a successful software idea can become even more successful if reimplementation on other devices is easy to achieve and if cross-modality permits users with disabilities to access the system. Remember Fitts's Law to optimize speed of performance, and consider two-handed operations.

Keyboard entry is here to stay, but consider other forms of input when text entry is limited. Selecting rather than typing has many benefits for both novice and frequent users. Direct-pointing devices are faster and more convenient for novices than are indirect-pointing devices, and accurate pointing is possible, but remember that users on the go are likely to use the devices with a single hand. Simple gestures can trigger a few actions and are showing promising applications. Mobile devices have become the universal computing platform of the world, and many people nowadays exclusively use their smartphone for both work and play. For this reason, designers may want to design mobile-first interfaces that are responsive to varying devices.

Display technology is moving rapidly, and user expectations are increasing. Multi-touch displays are now expected by most users, and these come with a standardized vocabulary of touch gestures. Beyond mobile devices, large high-resolution displays are becoming prevalent. The current next big thing in displays are those that go beyond the plane and become volumetric, deformable, and shape-changing, in effect blurring the border between input and output.

Researcher's Agenda

Novel text-entry keyboards to speed input and to reduce error rates will have to provide significant benefits to displace the well-entrenched QWERTY design. For the numerous applications that do require extensive text entry or run on mobile device applications, many opportunities remain to create special-purpose devices or to redesign the tasks to permit direct-manipulation selection instead of keyboarding. Increasingly, input can be accomplished via conversion or extraction of data from online sources. Another input source is optical character recognition of printed text or bar codes (for example, in magazines or bank statements) or RFID tags attached to objects, clothing, or even personalized dolls. Finally, while it is not possible to truly "beat" Fitts's Law for pointing situations, there exist many ways to assist with or minimize tasks that require pointing.

The range of display sizes available has widened enormously, and users need applications that can operate on mobile devices, desktops, and large wall or tabletop displays. Researchers need to understand how to design plastic or multi-modal interfaces that allow users to adapt their interfaces depending on the environment, their preferences, and their abilities. What are the strategies for increasing productivity with multiple screens? Sensors embedded in the environment and in many mobile devices can provide information about users' locations or activities to enable development of context-aware applications. The benefits may be large, but inconsistent behavior and privacy concerns will have to be addressed before adoption becomes widespread. Finally, the new breed of shape-changing displays requires considerable further study in order to realize its full potential.

WORLD WIDE WEB RESOURCES

www.pearsonglobaleditions.com/shneiderman

Input and output computing hardware is a major industry, and each manufacturer have its own websites and resources for more information. The following are the most dominant manufacturers for keyboards and pointing devices.

- Logitech:
  http://www.logitech.com/
- Microsoft: http://www.microsoft.com
- Dell: http://www.dell.com

Touchscreens are now commonplace and have no centralized resource, but Wacom has long been the leader in graphical tablets.

- Wacon: http://www.wacom.com

Common brands for large-scale touchscreens, such as tabletops and wall-mounted screens, include:

- SMART Technologies:
  http://www.smarttech.com
- Microsoft: http://www.microsoft.com

Finally, the Trace Center has excellent resources for accessible and inclusive input and output devices:

- Trace Center:
  http://trace.wisc.edu/

Discussion Questions

1. A company is designing a kiosk that can display weather information in public locations. The kiosk will feature a touch screen so users can select a city by pointing on a map. Give three reasons why a touch screen is an effective device for this application.
2. Explain the difference between direct-control and indirect-control pointing devices. Name a task when the one type is a more appropriate device than the other.
3. What are the different interaction tasks for which pointing devices are useful? How can the challenges faced by visually impaired people while using pointing devices be addressed?
4. Define responsive design. What characteristics of a display would make an individual state that the design they are viewing seems responsive?
5. What are the advantages of large wall displays? What are the limitations?
6. Give a definition of context-aware computing. Provide an example of one application of context-aware computing that would meet the user needs of a tourist.

References

Accot, Johnny, and Zhai, Shumin, More than dotting the i's: Foundations for crossing-based interfaces, *Proceedings of the ACM Conference on Human Factors in Computing Systems*, ACM Press, New York (2002), 73–80.

Alexander, Jason, Lucero, Andrés, and Subramanian, Sriram, Tilt displays: Designing display surfaces with multi-axis tilting and actuation, *Proceedings of the ACM Conference on Human Computer Interaction with Mobile Devices and Services*, ACM Press, New York (2012), 161–170.

Andrews, Christopher, Endert, Alex, Yost, Beth, and North, Chris, Information visualization on large, high-resolution displays: Issues, challenges, and opportunities, *Information Visualization 10*, 4 (2011), 341–355.

Ardito, Carmelo, Buono, Paolo, Costabile, Maria Francesca, and Desolda, Giuseppe, Interaction with Large Displays: A Survey, *ACM Computing Surveys 47*, 3 (2015), 46.

Azenkot, Shiri, Wobbrock, Jacob O., Prasain, Sanjana, and Ladner, Richard E., Input finger detection for nonvisual touch screen text entry in PerkInput, *Proceedings of Graphics Interface*, Canadian Information Processing Society, Toronto, ON, Canada (2012), 121–129.

Badam, Sriram Karthik, and Elmqvist, Niklas, PolyChrome: A cross-device framework for collaborative web visualization, *Proceedings of the ACM Conference on Interactive Tabletops and Surfaces*, ACM Press, New York (2014), 109–118.

Badam, Sriram Karthik, Chandrasegaran, Senthil, Elmqvist, Niklas, and Ramani, Karthik, Tracing and sketching performance using blunt-tipped styli on direct-touch tablets, *Proceedings of the ACM Conference on Advanced Visual Interfaces*, ACM Press, New York (2014), 193–200.

Balakrishnan, Ravin, "Beating" Fitts' law: Virtual enhancements for pointing facilitation, *International Journal of Human-Computer Studies 61*, 6 (2004), 857–874.

Ballard, Barbara, *Designing the Mobile User Experience*, John Wiley & Sons, New York (2007).

Baudisch, Patrick, and Chu, Gerry, Back-of-device interaction allows creating very small touch devices, *Proceedings of the ACM Conference on Human Factors in Computing Systems*, ACM Press, New York (2009), 1923–1932.

Baudisch, Patrick, and Holz, Christian, My new PC is a mobile phone, *XRDS: Crossroads, The ACM Magazine for Students 16,* 4 (2010), 36–41.

Chapuis, Olivier, and Dragicevic, Pierre, Effects of motor scale, visual scale and quantization on small target acquisition difficulty, *ACM Transactions on Computer-Human Interaction 18,* 3 (2011).

Chapuis, Olivier, Bezerianos, Anastasia, and Frantzeskakis, Stelios, Smarties: An input system for wall display development, *Proceedings of the ACM Conference on Human Factors in Computing Systems*, ACM Press, New York (2014), 2763–2772.

Clawson, James, Lyons, Kent, Rudnicky, Alex, Iannucci, Jr., Robert A., and Starner, Thad, Automatic whiteout++: Correcting mini-QWERTY typing errors using keypress timing, *Proceedings of the ACM Conference on Human Factors in Computing Systems*, ACM Press, New York (2008), 573–582.

Dachselt, Raimund, Häkkilä, Jonna, Jones, Matt, Löchtefeld, Markus, Rohs, Michael, and Rukzio, Enrico, Pico projectors: Firefly or bright future? *ACM Interactions 19*, 2 (2012), 24–29.

Dourish, Paul, and Bell, Genevieve, *Divining a Digital Future: Mess and Mythology in Ubiquitous Computing*, MIT Press (2011).

Dunlop, Mark, and Masters, Michelle, Pickup usability dominates: A brief history of mobile text entry research and adoption, *International Journal of Mobile Human Computer Interaction 1,* 1 (2008), 42–59.

Findlater, Leah, Wobbrock, Jacob O., and Wigdor, Daniel, Typing on flat glass: Examining ten-finger expert typing patterns on touch surfaces, *Proceedings of the ACM Conference on Human Factors in Computing Systems*, ACM Press, New York (2011), 2453–2462.

Fitts, P. M., The information capacity of the human motor system in controlling amplitude of movement, *Journal of Experimental Psychology* 47 (1954), 381–391.

Follmer, Sean, Leithinger, Daniel, Olwal, Alex, Hogge, Akitmitsu, and Ishii, Hiroshi, inFORM: Dynamic physical affordances and constraints through shape and object actuation, *Proceedings of the ACM Symposium on User Interface Software & Technology*, ACM Press, New York (2013), 417–426.

Forlines, Clifton, Vogel, Daniel, and Balakrishnan, Ravin, HybridPointing: Fluid switching between absolute and relative pointing with a direct input device, *Proceedings of the ACM Symposium on User Interface Software & Technology*, ACM Press, New York (2006), 211–220.

Grossman, Tovi, and Wigdor, Daniel, Going deeper: A taxonomy of 3D on the tabletop, *Proceedings of the IEEE International Workshop on Horizontal Interactive Human-Computer Systems*, IEEE Press, Los Alamitos, CA (2007), 137–144.

Grossman, Tovi, Dragicevic, Pierre, and Balakrishnan, Ravin, Strategies for accelerating on-line learning of hotkeys, *Proceedings of the ACM Conference on Human Factors in Computing Systems*, ACM Press New York (2007), 1591–1600.

Guiard, Yves, Bourgeois, Frédéric, Mottet, Denis, and Beaudouin-Lafon, Michel, Beyond the 10-bit barrier: Fitts' law in multi-scale electronic worlds, in *People and Computers XV—Interaction Without Frontiers (Joint Proceedings of HCI 2001 and IHM 2001)*, Springer-Verlag, London, UK (2001), 573–587.

Hansen, Thomas Riisgaard, Eriksson, Eva and Lykke-Olesen, Andreas, Use your head: Exploring face tracking for mobile interaction, *Extended Abstracts of the ACM Conference on Human Factors in Computing Systems*, ACM Press, New York (2006), 845–850.

Harrison, Chris, Appropriated interaction surfaces, *IEEE Computer 43*, 6 (2010), 86–89.

Hincapié-Ramos, Juan David, Guo, Xiang, Moghadasian, Paymahn, and Irani, Pourang, Consumed endurance: A metric to quantify arm fatigue of mid-air interactions, *Proceedings of the ACM Conference on Human Factors in Computing Systems*, ACM Press, New York (2014), 1063–1072.

Hinckley, Ken, and Wigdor, Daniel, Input technologies and techniques, in Jacko, Julie, and Sears, Andrews (Editors), *The Human-Computer Interaction Handbook*, Laurence Erlbaum Associates, Mahwah, NJ (2011), 95–132.

Hoggan, Eve, Brewster, Stephen A., and Johnston, Jody, Investigating the effectiveness of tactile feedback for mobile touchscreens, *Proceedings of the ACM Conference on Human Factors in Computing Systems*, ACM Press, New York (2008), 1573–1582.

Hourcade, Juan-Pablo, Bederson, Benjamin B., Druin, Allison, and Guimbretière, Francois, Differences in pointing task performance between preschool children and adults using mice, *ACM Transactions on Computer-Human Interaction 11,* 4 (December 2004), 357–386.

Jacob, Robert J. K., Girouard, Audrey, Hirshfield, Leanne M., Horn, Michael S., Shaer, Orit, Solovey, Erin Treacey, and Zigelbaum, Jamie, Reality-based interaction: A framework for post-WIMP interfaces, *Proceedings of the ACM Conference on Human Factors in Computing Systems*, ACM Press, New York (2008), 201–210.

Jansen, Yvonne, and Dragicevic, Pierre, and Fekete, Jean-Daniel, Evaluating the efficiency of physical visualizations, *Proceedings of the ACM Conference on Human Factors in Computing Systems*, ACM Press, New York (2013), 2593–2602.

Kane, Shaun, Wobbrock, Jacob O., Harniss, Mark, and Johnson, Kurt L., TrueKeys: Identifying and correcting typing errors for people with motor impairments, *Proceedings of the ACM Conference on Intelligent User Interfaces*, ACM Press, New York (2008), New York, 349–352.

Karlson, Amy, Bederson, Benjamin B., and Contreras-Vidal, Jose L., Understanding one handed use of mobile devices, in Lumsden, Johanna (Editor), *Handbook of Research on User Interface Design and Evaluation for Mobile Technology*, Information Science Reference/IGI Global, Hershey, PA (2008), 86–101.

Kortum, Philip (Editor), *HCI Beyond the GUI: Design for Haptic, Speech, Olfactory and Other Nontraditional Interfaces*, Elsevier/Morgan Kaufmann, Amsterdam, Netherlands (2008).

Kristensson, Per Ola, and Denby, Leif C., Text entry performance of state of the art unconstrained handwriting recognition: A longitudinal user study, *Proceedings of the ACM Conference on Human Factors in Computing Systems*, ACM Press, New York (2009), 567–570.

Lahey, Byron, Girouard, Audrey, Burleson, Winslow, and Vertegaal, Roel, PaperPhone: Understanding the use of bend gestures in mobile devices with flexible electronic paper displays, *Proceedings of the ACM Conference on Human Factors in Computing Systems*, ACM Press, New York (2011), 1303–1312.

Liao, Chunyuan, Guimbretière, François, Hinckley, Ken, and Hollan, Jim, Papier-Craft: A gesture-based command system for interactive paper, *ACM Transactions on Computer-Human Interaction 14,* 4 (2008), 14.

MacKenzie, I. Scott, Kober, Hedy, Smith, Derek, Jones, Terry, and Skepner, Eugene, LetterWise: Prefix-based disambiguation for mobile text input, *Proceedings of the ACM Symposium on User Interface Software and Technology*, ACM Press, New York (2001), 111–120.

MacKenzie, Ian Scott, *Human-Computer Interaction: An Empirical Research Perspective*, Morgan Kaufmann, San Francisco, CA (2013).

McCallum, David, and Irani, Pourang, ARC-Pad: Absolute + relative cursor positioning for large displays with a mobile touchscreen, *Proceedings of the ACM Symposium on User Interface Software & Technology*, ACM Press, New York (2009), 153–156.

McGrath, Will, Bowman, Brian, McCallum, David, Hincapié-Ramos, Juan David, Elmqvist, Niklas, and Irani, Pourang, Branch-Explore-Merge: Facilitating real-time revision control in collaborative visual exploration, *Proceedings of the ACM Conference on Interactive Tabletops and Surfaces*, ACM Press, New York (2012), 235–244.

Mithal, Anant Kartik and Douglas, Sarah A., Differences in movement microstructure of the mouse and the finger-controlled isometric joystick, *Proceedings of the ACM Conference on Human Factors in Computing Systems*, ACM Press, New York (1996), 300–307.

Montemayor, Jaime, Druin, Allison, Chipman, Gene, Farber, Allison, and Guha, Mona Leigh, Tools for children to create physical interactive storyrooms, *ACM Computers in Entertainment 1,* 2 (2004), 12.

Murugappan, Sundar, Vinayak, Elmqvist, Niklas, and Ramani, Karthik, Extended multitouch: Recovering touch posture and differentiating users using a depth camera, *Proceedings of the ACM Symposium on User Interface Software and Technology*, ACM Press, New York (2012), 487–496.

Myers, Brad, Using handhelds for wireless remote control of PCs and appliances, *Interacting with Computers 17*, 3 (2005), 251–264.

Öquist, Gustav, and Goldstein, Mikael, Towards an improved readability on mobile devices: Evaluating adaptive rapid serial visual presentation, *Interacting with Computers 15,* 4 (2003), 539–558.

Oney, Stephen, Harrison, Chris, Ogan, Amy, and Wiese, Jason, ZoomBoard: A diminutive QWERTY soft keyboard using iterative zooming for ultra-small devices, *Proceedings of the ACM Conference on Human Factors in Computing Systems*, ACM Press, New York (2013), 2799–2802.

Papadopoulos, Chariloas, Petkov, Kaloian, Kaufman, Arie E., and Mueller, Klaus, The Reality Deck – An immersive gigapixel display, *IEEE Computers & Graphics 35,* 1 (2015), 33–45.

Pew Research Center, *Emerging Nations Embrace Internet*, *Mobile Technology*, February 2014.

Raghunath, Mandayam, Narayanaswami, Chandra, and Pinhanez, Claudio, Fostering a symbiotic handheld environment, *IEEE Computer 36,* 9 (2003), 57–65.

Rogers, Yvonne, and Lindley, Siân E., Collaborating around vertical and horizontal large interactive displays: Which way is best? *Interacting with Computers 16,* 6 (2004), 1133–1152.

Rohs, Michael, and Oulasvirta, Antti, Target acquisition with camera phones when used as magic lenses, *Proceedings of the ACM Conference on Human Factors in Computing Systems*, ACM Press, New York (2008), 1409–1410.

Southern, Caleb, Clawson, James, Frey, Brian, Abowd, Gregory D., and Romero, Mario, An evaluation of BrailleTouch: Mobile touchscreen text entry for the visually impaired, *Proceedings of the ACM Conference on Human-Computer Interaction with Mobile Devices and Services*, ACM Press, New York (2012), 317–326.

Stellmach, Sophie, and Dachselt, Raimund, Still looking: Investigating seamless gaze-supported selection, positioning, and manipulation of distant targets, *Proceedings of the ACM Conference on Human Factors in Computing Systems*, ACM Press, New York (2013), 285–294.

Vanderheiden, G., Kelso, D., and Krueger, M., Extended usability versus accessibility in voting systems, *Proceedings of the 27 RESNA International Annual Conference*, RESNA Press, Arlington, VA (2004).

Vogel, Daniel, and Balakrishnan, Ravin, Occlusion-aware interfaces, *Proceedings of the Conference on Human Factors in Computing Systems*, ACM Press, New York (2010), 263–272.

Vogel, Daniel, and Baudisch, Patrick, Shift: A technique for operating pen-based interfaces using touch, *Proceedings of the ACM Conference on Human Factors in Computing Systems*, ACM Press, New York (2007), 657–666.

Wills, Sebastian A., and MacKay, David J. C., Dasher - An efficient writing system for brain-computer interfaces? *IEEE Transactions on Neural Systems and Rehabilitation Engineering 14,* 2 (2006), 244–246.

Wigdor, Daniel, and Wixon, Dennis, *Brave NUI World: Designing Natural User Interfaces for Touch and Gesture*, Morgan Kaufmann (2011).

Wobbrock, J., Myers, B., and Kembel, J., EdgeWrite: A stylus-based text entry method designed for high accuracy and stability of motion, *Proceedings of the ACM Symposium on User Interface Software and Technology*, ACM Press, New York (2003), 61–70.

Wobbrock, J. O., Rubinstein, J., Sawyer, M. W., and Duchowski, A. T., Longitudinal evaluation of discrete consecutive gaze gestures for text entry, *Proceedings of the ACM Symposium on Eye Tracking Research and Applications*, ACM Press, New York (2008), 11–18.

Ye, Hanlu, Malu, Meethu, Oh, Uran, and Findlater, Leah, Current and future mobile and wearable device use by people with visual impairments, *Proceedings of the ACM Conference on Human Factors in Computing Systems*, ACM Press, New York (2014), 3123–3132.

Zhai, Shumin, and Kristensson, Per Ola, Shorthand writing on stylus keyboard, *Proceedings of the ACM Conference on Human Factors in Computing Systems*, ACM Press, New York (2003), 97–104.


Cancel
Subject: Dogsitter?
Hey!
Are you available

CHAPTER 11

Communication and Collaboration

written in collaboration with Nicholas Diakopoulos

> Coming together is a beginning, staying together is progress, and working together is success.
>
> **Henry Ford**

> Drop the 'The.' Just 'Facebook.' It's cleaner.
>
> **Justin Timberlake as Sean Parker**
> *The Social Network*, 2010

CHAPTER OUTLINE

11.1 Introduction

Constant and immediate communication and interaction with family, friends, collaborators, colleagues, coworkers, and even pets are now commonplace in the increasingly networked world. Communication and collaboration systems are re-making entire swaths of how lives are lived, including how people do work, find romance, exchange and deliberate policy, engage in civic participation, produce software and other creative wares, play and entertain themselves, shop for and select products and services, find support when they are in ill health, gain an education, and receive and produce information (Rainie and Wellman, 2012). The intrinsically motivating role of interpersonal connectedness (Deci and Flaste, 1995) drives human beings to want to communicate and interact with others across the full range of experiences.

Originally launched as a microblogging platform in 2006 for people to post short, 140-character text messages about what they are doing, Twitter allows users to subscribe to each other's feeds, forming a "publish and subscribe" social network. In the 10 years since its launch, the platform has become an important way for people to share information, find experts, surface breaking news photos and videos, coordinate efforts during natural disaster responses, and maintain an awareness of the pulse and reactions of those on the network (Vieweg et al., 2010; Diakopoulos et al., 2012). It's reshaping how news and information are disseminated, moving away from a traditional gatekeeping model where professional editors had ultimate control over what was published or promoted to a system in which social activity, personal connections, and algorithmic ranking are just as important. Facebook is another major social network site (boyd and Ellison, 2007) and in 2015 had close to 1.5 billion global users who log on at least once a month. Facebook is similarly changing the information production and consumption landscape and has also been shown to have substantial benefits for individuals' relationships. People often join Facebook in order to keep up with friends or to solidify relationships with acquaintances like dormmates, classmates, and work colleagues. Early research has shown that Facebook helps people build social capital—the resources available to an individual as a result of having a durable network of relationships (Ellison et al., 2007). While Facebook and Twitter are dominant in the United States, other social network sites and chat platforms are flourishing elsewhere around the world, such as Weibo and Renren in China, VKontakte in Russia, and Kakao Talk in South Korea. Such platforms allow users to easily and cheaply maintain connections and crystallize relationships. The positive outcomes from such platforms are wide-ranging, from job seekers hoping for support or a lead on a new job (Burke and Kraut, 2013) to citizens seeking to organize a political protest to fight oppression. Social network sites can even enable positive public health outcomes such as contributing to smoking cessation programs by providing social support (Phua, 2013).

But it's also important for designers to consider and account for the downsides and negative exigencies of such systems. Criminals, terrorist recruiters, and oppressive political leaders can use social network sites for negative purposes. A miscreant can bully or deceive others online, and hate groups can spew their propaganda. Some may become addicted to communication tools or waste time that could be more productive, while others will inadvertently share photos or other information that damages their reputation and is difficult to remove from their online profile. Social media can make behaviors such as stalking and public shaming easier (Ronson, 2015). Trusting or naïve children and adolescents may become targets of predatory adults seeking underage sexual relationships. Ideas can become polarized when liberal and conservative thinkers cluster together based on homophily and pay less attention across the philosophical divide. Sherry Turkle has been critical of the role that mobile communication technologies play in distracting people from fully participating in real-life conversation (Turkle, 2015). Important questions for society to consider are how new forms of communication change the way people think, build relationships and community, and practice political organization. Designers must be aware of such behaviors and possible outcomes and consider design options that circumvent or mitigate the worst possibilities. Good design, effective community leadership, and thoughtful governance policies and strategies can lead to more positive social outcomes.

Despite their huge popularity, social network sites are just one particular form of online communication. A Pew survey of Americans in 2014 found that four of the five top uses of smartphones were communication applications with modalities including text messaging, voice or video calls, e-mail, and social networking (Smith and Page, 2015). Different communication channels and tools are more or less suited for different tasks and human needs, whether they be chatting with a friend or coworker, writing a collaborative document with someone, posting to a discussion forum or Q/A site, participating in group project management, coordinating a real-world community gathering like a meetup, sharing files, or teleconferencing, among others. The academic field that emerged in the 1980s to study technology used by two or more people is called computer-supported cooperative work (CSCW), though 30 years later the field has also adopted *social computing* as an umbrella term that implies less of a strict devotion to collaboration and work per se and includes cooperation, collaboration, and competition as well as non-work activities like gaming and romance.

The communication and collaboration tools that designers create shape the ability to work and accomplish shared goals with one another. The degree of interactivity, the social cues present in the interface, and the mobility of communication technologies are but a few of the design dimensions that affect use in different contexts (Baym, 2015). Design is just a starting point for behavior, though, and often people will quickly repurpose communication technologies to accommodate their specific needs. On Twitter, the need to re-share information while attributing it to the original source (a social norm) led to linguistic innovations like *via*, *retweeting*, and *R/T* before ultimately catching on as *RT* a compressed version of *retweet* that took up minimal space within the 140 characters allowed in a message (Kooti et al., 2012). Years later, Twitter formalized this convention by building the ability to retweet a message directly into the platform. The social shaping of technology perspective reflects the idea that the tools that designers craft do not precisely determine how people will use them but rather interact with human goals as the technology co-evolves: "The consequences of technologies arise from a mix of 'affordances'—the social capabilities technological qualities enable—and the unexpected and emergent ways that people make use of those affordances" (Baym, 2015). Interface and experience design must be an iterative and constantly evolving endeavor as people and technology co-evolve.

Research in collaboration and communication interfaces is often more complex than in single-user interfaces. The multiplicity of users makes it difficult to conduct experiments that control for group variability. Differences in physical distribution of participants can make the application of some research methods considerably more arduous. Studies of small-group psychology, industrial and organizational behavior, sociology, and anthropology can provide useful research paradigms (Lofland and Lofland, 2005). Content analysis methods can be used to analyze and create typologies of the types of messages that individuals post, leading to insights not only about content but also about the relationships between individuals (Riffe et al., 2013). Also, as questions of macro-HCI are considered (see Chapter 3), studying communication platforms at scale requires that methods in data science be adapted. Communication texts, including chat logs, tweets, Facebook posts, and online comments can be analyzed using natural language processing (NLP) algorithms. These methods are useful for identifying, counting, or scoring texts (Diakopoulos, 2015), such as according to positive or negative sentiments expressed. Text analysis can also be combined with structural understanding using methods from social network analysis (Hansen et al., 2011; Leetaru, 2011). For example, topical network maps have elucidated structures among online Twitter groups like polarized crowds, tight crowds, brand clusters, community clusters, and broadcast or support networks (see Fig. 11.1). Such methods allow better understanding of how individuals organize and communicate online, elucidating structures and strategic locations or roles within the network (Smith et al., 2014).

Questions of ethics become paramount when studying open communication networks as people may share sensitive personal information without realizing

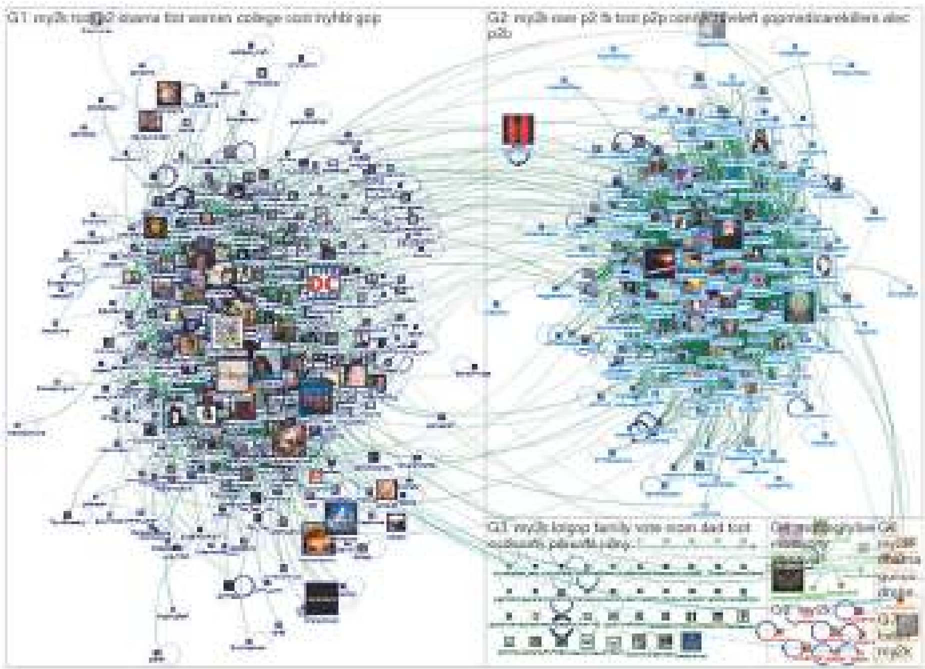


FIGURE 11.1

A network map made with NodeXL software shows the polarized nature of the conversation on Twitter around the #My2k hashtag that emerged over U.S. budget struggles in 2012.

that a researcher might scoop it up and analyze it for other purposes. Researchers must carefully consider whether they should anonymize names and remove other identifying information from subsequent published analyses (Bruckman et al., 2015) as well as more generally consider the risks and benefits of such research. A Facebook study indicating that users' emotional state could be modulated based on the positive or negative sentiments they were exposed to in their newsfeed (Kramer et al., 2014) resulted in widespread questioning of the ethics of such experimental manipulations. A survey of researchers who use online data found consensus around several ethical guidelines, including notifying participants about why data is being collected, sharing research results with participants, removing individuals from datasets upon their request, and being cautious when sharing results with identifiable outliers (Vitak et al., 2016). Studying others' platforms leads to additional challenges like a lack of control of the interface and an inability to know how the platform may be shifting or dynamic due to A/B tests (see Chapter 5.3.4). One way to overcome some of these issues is to build proprietary social software and develop a user base that is genuinely motivated to inhabit that online space. This approach is difficult but has been successful for projects like GroupLens at the University of Minnesota, which has been able to run large-scale studies of recommender systems and other online communities in this fashion.

The following section (Section 11.2) presents a model to help orient readers within the design space. In Section 11.3, different collaboration and communication goals and contexts are presented to illustrate how design can adapt to support different user needs. And finally, in Section 11.4, several design considerations and challenges related to communication and collaboration technology are articulated.

11.2 Models of Collaboration

Consider a typical day for a digital native: Wake up and check social networking accounts to get the latest news, go to work and collaboratively edit a report, chat with an office colleague about the new intern, post a question to a Q/A site about a statistical test needed to complete the report, then on the way out of the office text message a significant other to coordinate dinner plans, and after dinner receive a crowd-based recommendation for a movie to watch. Each of these activities hinges on communication—a process in which information is exchanged between individuals—if not active collaboration involving achieving or doing something with those individuals. Yet the wide variety of types of communication and collaboration raises the question of how to make sense of the design space. Which of these daily activities is more similar and more different from a design point of view? A descriptive model or framework for design can help start putting this into perspective and provide the ability to recognize, compare, and discuss the features and demands of various design contexts.

The traditional way to decompose collaborative interfaces is by using the time/space matrix, which has four quadrants: same time, same place (e.g., shared table display, wall display); same time, different place (e.g., teleconferencing); different time, same place (e.g., public display); and different time, different place (e.g., e-mail, discussion forums, version control). The terms *synchronous* (same time), *asynchronous* (different time), *co-located* (same place), and *remote* (different place) are often used. Certainly time and space are both important dimensions to consider when designing such collaboration tools, but the binary nature of the matrix is somewhat of an oversimplification. In terms of time, for instance, modern communication tools like Slack or Facebook blur the line between asynchronous messaging and synchronous chat and are not distinctly asynchronous or synchronous.

A more contemporary framework that operates at the mezzo and macro level is the Model of Coordinated Action (MoCA), which incorporates the traditional model but expands it into a set of seven dimensions and shifts toward a deeper understanding of "coordinated action" in order to encompass goal-directed activities that are not traditionally considered to be work (Lee and Paine, 2015). Lee and Paine define coordinated action as the "interdependence of two or more actors who, in their individual activities, are *working towards a particular goal through one or more overlapping fields of action*." This definition accounts for situations where participants may be collaborating in a diffuse or even indirect way, such as crowdsourcing or collaborative recommendation engines. A "field of action" need not be the same for all collaborators as they take on different tasks in order to accomplish some greater goal. The seven dimensions, described in detail next and shown in Fig. 11.2 are synchronicity, physical distribution, scale, number of communities of practice, nascence, planned permanence, and turnover. In some cases, these dimensions will reflect on the design of the communication tool or platform, but just as often, the nature, qualities, and intents of the people involved in the coordinated actions play just as large a role in achieving a successful outcome. Universal designs for collaboration and communication systems fluidly accommodate users across the spectrum of these dimensions.

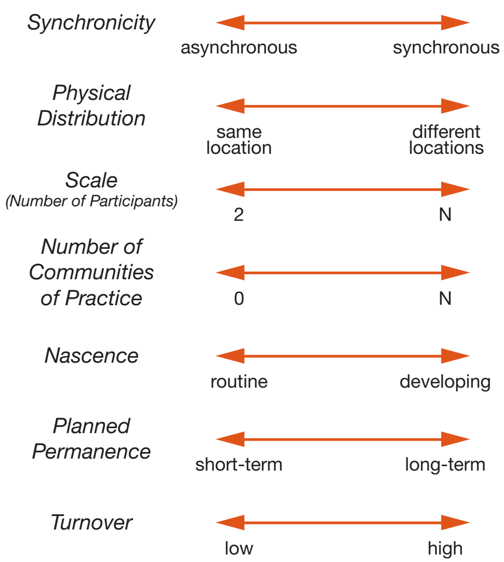


FIGURE 11.2

The seven dimensions of the MoCA model (Lee and Paine, 2015).

11.2.1 Synchronicity

Coordinated actions can take place along a spectrum of synchronicity, ranging from actions that are entirely asynchronous to those that are entirely synchronous. Importantly, this dimension allows for actions to be a mixture of synchronous and asynchronous rather than enforcing a distinct boundary between extremes. In ongoing work processes, a larger context of asynchronous interaction often embeds episodic synchronous activity (Olson and Olson, 2000). Examples of more synchronous communication channels include voice or video conferencing (e.g., Skype), whereas more asynchronous channels include things like messaging systems or Q/A forums (e.g., iMessage and Stack Overflow, respectively). The degree of synchronicity of a channel (i.e., the delay between turns) can also be a function of its context of use or social expectations. Imagine users in a chat session on their phones, happily exchanging messages in a near-synchronous fashion when one chat partner introduces a large delay into the next response. Maybe the partner got distracted, or his or her attention otherwise shifted and was reprioritized, but the result is that the chat went from being more synchronous to more asynchronous.

Newer project management tools reflect the idea that collaborations often require a mixture of asynchronous and synchronous communication and allow users options within that range. Google Docs is a writing program that allows collaborators to co-write texts. It allows different users to write asynchronously across different time zones or work shifts as well as allowing for real-time editing by multiple users. Moreover, for situations where there is a need to work concurrently and resolve questions, there is chat functionality built in so that collaborators can exchange messages and discuss any edits they may need to make (see Fig. 11.3).

11.2.2 Physical distribution

Teams working together can exist along a continuum ranging from being at the same shared desk to the same room, building, campus, city, country, continent, or planet. A collaboration can thus be more or less physically distributed. Despite all of the internet communication channels available to users, the actual physical distribution of collaborators still matters (Olson and Olson, 2000). Physical presence can afford unplanned interactions and rapport building that are unavailable through other channels—sometimes the trust built over such "watercooler" talk is essential to project success or resilience.

Physical location can be a proxy for cultural differences that might include expectations for things like pauses in conversation or who has permission to speak when and to whom (Olson and Olson, 2013). Different countries may have different holiday schedules or work hours. For instance, in parts of the Middle East, Sunday is a work day, whereas in the United States it is not. Time zones can make

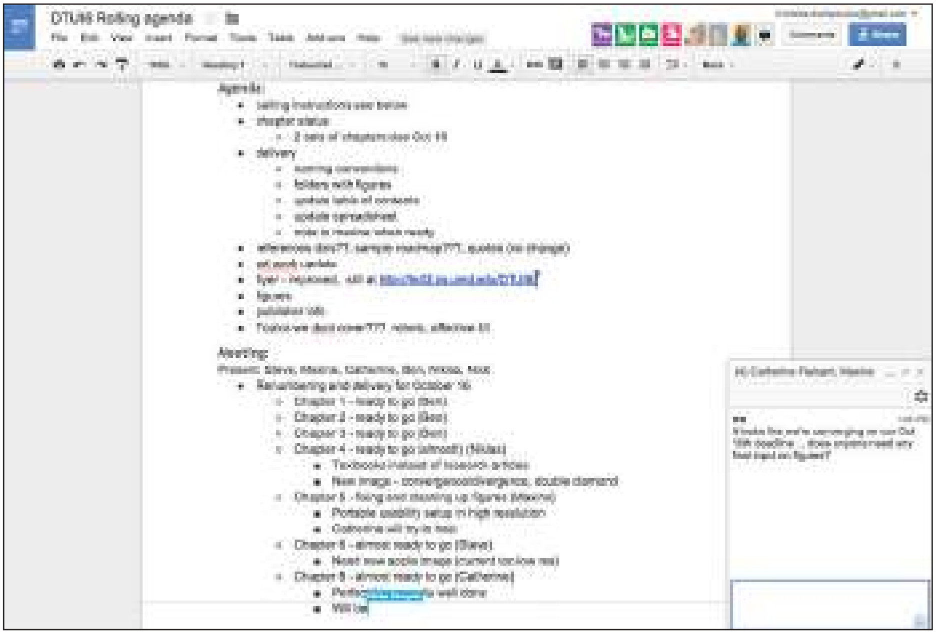


FIGURE 11.3

The Google Docs interface showing how multiple users can simultaneously edit a document. Note that colored flags for different users offer concrete feedback that signals who is editing a particular part of the document. In the lower right corner, a chat box invites users to synchronously converse with each other around the document.

it difficult to schedule synchronous communications that are convenient to all parties: a 5 p.m. call in Norway is an 8 a.m. call in California, at the extremes of the work day and probably not terribly convenient for either party. Who on the team deserves to be *more* inconvenienced in case a common time cannot be found?

11.2.3 Scale

The number of participants involved, or the scale of the collaboration, is an important dimension that affects the nature and type of interactions that emerge. The difference is substantial between co-writing a paper with one other person and contributing to a Wikipedia article with 10 or 100 others. In smaller collaborations, each person might know everyone else by name, whereas in much larger-scale collective actions, there may in fact be little direct contact between individuals. Many users may remain in lightweight contact as occasional lurkers or provide fine-grained contributions like voting or tagging. Wikipedia manages what is often a large scale of contributions from different users by allowing for a range of granularities of contributions, anywhere from fixing spelling or

FIGURE 11.4

A Wikipedia page after editing. In the UI tabs at the top of the page, users can quickly access the "Edit" tab where they can directly edit and then save a new version of the page. For more substantial editing decisions, they might visit the "Talk" tab to discuss with other editors first.

adding a missing comma to conceptually restructuring the article (see Fig. 11.4). In traditional organizations, the typical way to deal with larger-scale tasks is to introduce a hierarchy that decomposes tasks and clarifies authority and accountability. Hierarchical task decomposition, integration of work, and quality oversight have been explored in large-scale collaborations like crowdwork (Kittur et al., 2013). As efforts scale, the role of leadership and expertise becomes apparent in ensuring a successful outcome (Luther et al., 2010).

11.2.4 Number of communities of practice

A community of practice refers to the idea that over time individuals form a group as they teach and learn from one another and develop coherent values, norms, and practices (Wenger, 1998). A group or team may reflect many different communities of practice that must come together to coordinate and work toward a common goal. The notion of interdisciplinarity is key here: When scientists want to start a company around their brilliant new discovery, they need disciplines like engineering, law, accounting, communications, and business to develop that idea into a viable product for the market. Yet the ways of thinking and the vocabulary in these different communities of practice may differ substantially—to an engineer “code” is something that a computer interprets, whereas for a legal professional it may correspond to a statute that needs to be adhered to. Different communities may have different goals or notions of impact, different standards, or different toolsets that reflect disciplinary educational patterns. At one extreme of the spectrum, a group may consist of a single community of practice, whereas at the other end, there is a huge diversity of people in the world that may come together from various disciplines, ways of working, language, and culture to collaborate. Working in diverse teams means that different communities of practice may need to be bridged.

11.2.5 Nascence

New groups of people are constantly popping into existence to coordinate their actions, while others have been around for a long time. Some of these groups only last for a short while, such as for coordinating a response to a local natural disaster, whereas other groups may be around for many years or even generations. Nascence refers to the degree to which coordinated actions are already established and routine or if they are un-established, new, and developing. For instance, early on, different (and potentially diverse) team members may need to align goals and develop common ground. The organization of collaborative work is often in flux on the birthing of a new group, team, or community, but that is not to say that communities that have been around much longer can’t still continue to develop. Periods of routine work may need to adapt all of a sudden to new demands, contexts, or norms. Research has shown that the characteristics and behaviors of founders early in a group’s lifespan predict how long it will survive (Kraut and Fiore, 2014). Actions like visiting the group frequently, having multiple group administrators, and articulating a group description and logo during the group’s most nascent stage (i.e., the first week) predicted group survival.

11.2.6 Planned permanence

Some coordinated actions are shorter-term, whereas others are longer-term. For instance, responding to a crisis event may take place over the course of hours, days, weeks, months, or even years, and it may be apparent at the outset based on the magnitude of the response needed that the timescale would fit into any of those buckets. Regardless of whether collaborations are temporary or permanent, the participants will need to develop shared vocabulary and coordinate work practices and output. When participants know that a collaboration should and will endure for a longer time frame, they may also begin to develop their own standards that coalesce ideas from different communities of practice. Planning for longer-term collaboration can oftentimes entail a higher overhead in terms developing and agreeing on administrative and work frameworks or policies.

FIGURE 11.5

Three example badges used in the DUST alternate reality game (http://fallingdust.com) to signal different achievements during collaborative play.

11.2.7 Turnover

Turnover refers to the stability of the people involved in a collaboration in terms of how frequently new participants enter and leave the group. On one end of the spectrum are coordinated actions that may have a very slow churn, such as an e-mail list of school administrators, whereas on the other end are collaborations where new people are constantly coming and going, such as online discussion boards that don't require registration. For instance, an analysis of the online commenting activity on *The Economist*'s Graphic Detail blog showed that over the course of eight months, about 79% of users who commented did so only once, indicating a smaller group (21% of users) who repeatedly left comments (Hullman et al., 2015). Such a high turnover and steady inflow of new contributors can pose difficulties to developing policies and behavioral expectations and norms for the group. One design approach toward this issue is to give users badges that indicate their tenure within the community or that otherwise mark them as "verified" or "trusted" (see Fig. 11.5). Another approach is to welcome newcomers to a community by creating positive onboarding experiences and interactions with established community members (Morgan et al., 2013).

11.3 Specific Goals and Contexts

People collaborate because doing so is satisfying or productive. Collaboration allows individuals to reap the emotional rewards of socializing and interacting with others, to accomplish greater goals than they could alone, or to meet and transact with people who they otherwise couldn't. In this section, a macro-HCI perspective is taken by exploring the dimensions of the MoCA model in connection with diverse contexts in which collaborations emerge. Contexts vary not only in terms of the goals and tasks that primarily concern users but also according to the social and physical context (e.g., mobile, in a car, in a classroom, in a public space) as well as along the dimensions of MoCA.

11.3.1 Communication and conversation

One of the essential coordinated actions that most people participate in on a daily basis is exchanging ideas, information, and knowledge with other people via conversation. Users can do so via their voice, by writing down those ideas, or by using their faces to emote and their bodies to gesture. Different conversation tools make possible the use of one modality or another, but people will make use of whatever channels they have available. For instance, on the telephone, users lose the ability to scowl at their interlocutor as a sign of disagreement, whereas on a Skype video conference, users do have the ability to use nonverbal visual cues like facial expressions, hand gestures, body posture, or direction of gaze in order to help express an idea. At the same time, while unhappy telephone users can't convey a scowl, they can still modulate their voice to make a disagreement known. Telepresence (see Chapter 7) goes even further by providing a panoramic multi-video view and a more immersive (e.g., 3-D virtual reality) experience or by physically extending one participant into the space of another using robotics. Design dimensions such as the physical environment, mobility of participants, and visual feedback, among others, are factors in such systems (Rae et al., 2015).

Conversation systems often vary in the degree of *synchronicity* they support. Voice or video conferencing systems tend to be highly synchronous, whereas chat systems can support synchronous or asynchronous modes, and discussion boards or e-mail listservs tend to be less synchronous. Some chat systems have explored the implications for how interactions change when communications are ephemeral by design. For instance, SnapChat (shown in Fig. 11.6) is a popular app that allows users to share silly photos, videos, and doodles with their friends that can only be viewed for up to 10 seconds before they disappear. This raises an interesting design question about how limiting the *planned permanence* through a design constraint can lead to interesting new genres of communication.

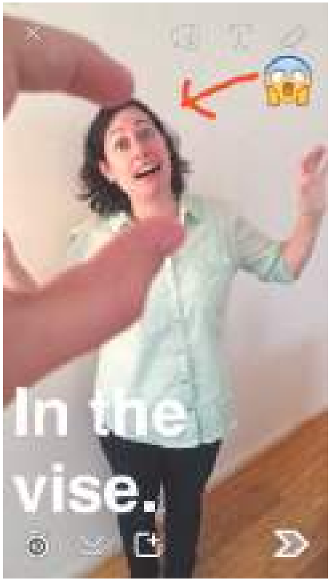


FIGURE 11.6
SnapChat is an app that allows for the composition of photos, doodles, and emojis that when sent to friends can only be viewed for up to 10 seconds as shown by the small clock with the "10" in the left corner of the composition UI.

Conversation systems often also vary along the dimension of *scale*: chatting with a best friend over iMessage is a very different kind of experience from participating in an e-mail group listserv, which is different still from writing a public comment on a *New York Times* article. In large-scale discussions, the pace may be slower and users may have little or no knowledge of the identity, reputation, background, or physical location of other participants. In order to coalesce conversations that are dispersed in a social network site like Twitter, the use of hashtags has emerged so that disconnected users can find each other by referencing and searching for the hashtag. Anonymity or even just the lack of persistent identity can result in flaming behavior, including swearing, name calling, or other ad hominem attacks (Diakopoulos and Naaman, 2011). Missing identity information can mask large disparities in the *number of communities* of practice and the *turnover* of the participants making it difficult to build common ground and shared vocabulary.

11.3.2 Online markets

Buying, selling, and trading—they've been driving commerce for a few thousand years. But the past 20 years have seen the nature of these activities change dramatically as people adopt the internet for shopping, trading, buying, selling, and delivery of goods. Online marketplaces like Amazon, Airbnb, eBay, and Etsy present a multitude of options, allowing buyers to connect and do business not only with traditional corporations but also with individual collectors, artisans, or service providers. This allows for hitherto unseen *scale* and *physical distribution* of participants in a transaction. Etsy users can buy a handcrafted steampunk costume item from the other side of the world as easily as the other side of the city (though maybe with different shipping costs). Markets often tend to be lower on the *synchronicity* scale in order to accommodate convenience for buyers and sellers shopping and fulfilling orders on their own clock as needed. Engendering trust in participants who have never met and may not even have a lot of background on each other is a key challenge for designing effective online marketplaces.

To cope with the scale of people, items, and content that is present in online markets, *collaborative filtering* algorithms have been developed (Linden et al., 2003). One approach to collaborative filtering works by representing the purchase or preference data of each individual in the market and then identifying other people who have a similar profile. Items from other users that are similar to a given user are then ranked and presented to that user as "related." The thinking is that if two users are similar in their preferences and purchases, then they might have good product recommendations for each other. What's interesting about collaborative filtering is that it is an implicit form of collaboration: Two individuals may have never interacted directly or even know about each other—in fact, they may be completely anonymous to one another. Other forms of feedback have also emerged in online markets. eBay, for instance, has a feedback score that is simply calculated as +1 point for a positive rating and -1 point for a

Star	Color	Number of ratings
	Yellow	10 to 49
	Blue	50 to 99
	Turquoise	100 to 499
	Purple	500 to 999
	Red	1,000 to 4,999
	Green	5,000 to 9,999
	Yellow shooting star	10,000 to 24,999
	Turquoise shooting star	25,000 to 49,999
	Purple shooting star	50,000 to 99,999
	Red shooting star	100,000 to 499,999
	Green shooting star	500,000 to 999,999
	Silver shooting star	1,000,000 or more (Wow!)

FIGURE 11.7

The eBay rating scale uses icons that correspond to different levels of positive feedback.

negative rating. Higher scores are signaled via icons of stars of various colors and styles being shown next to a user's account, allowing users to quickly assess whether they want to do business with each other (see Fig. 11.7).

11.3.3 Meeting coordination

Sometimes users merely need the aid of communication tools in order to coordinate a real-world meeting time and location. Such tools can allow users to create groups and communities that can come together as needed to coordinate *IRL* (in real life). A powerful example of this is the Meetup platform, which according to its website in 2015 claims to facilitate more than 9,000 local group meetings every day by helping those groups self-organize. The platform integrates capabilities to schedule and locate meeting events, to send e-mails out to group members, to record RSVPs and manage attendance, to upload and share media like photos and videos after the event, and to comment on events and share information or opinions that are persistent for others. Meetup is squarely targeted at supporting communities that are in a relatively nearby *physical* *distribution*, such as the same city. Its website's features are also mostly oriented toward the asynchronous end of the *synchronicity* spectrum (i.e., asynchronous planning for a synchronous event), making it ideal for casual users to come in and out of the event planning on their own time. The platform is agnostic to dimensions such as *number of communities*, *turnover*, and *planned permanence*, allowing for a great deal of flexibility for group leaders to define the scope of their community as needed.

Another very popular context for meeting coordination platforms is online dating sites. The needs of such tools are very different to a platform like Meetup, though; the *scale* of a meeting, for instance, is (most often) fixed at two. Potential romantic partners need ways to break the ice in what can sometimes be an awkward situation, they need ways to chat sometimes synchronously and sometimes asynchronously, and they need ways to arrange for dates where they feel comfortable and safe. The ways in which users are able to portray their identity and personality are important. In late 2015, eHarmony boasted an average of 438 marriages a day as a result of partners meeting on the site, and the OKCupid site reported more than 7 million messages exchanged per day among hundreds of thousands of users. Yet the demands of the romantic meeting context are quite different than those of a more general-purpose social network site. *Turnover* of users is quite high as old users are matched and drop off the platform. The frequent desire to meet romantic partners face to face means that strong filtering for the *physical distribution* of matches is a key feature. The types of communications afforded need to support an entire range of *planned permanence* from a single message exchange to a single IRL meeting to a more involved relationship that results in a long-term relationship. The communication on such platforms can adapt as the planned permanence of the interactions shifts.

11.3.4 Creative production

Whether it's developing new software, writing an online encyclopedia, remixing or animating a movie, or conducting an international science experiment, big creative projects demand that users work together. Work needs to be broken down into pieces and re-assembled, contingencies and interdependencies require planning, quality must be ensured, different roles and skills must be brought to bear, and supporting administrative duties underlie it all. Because creative productions often involve original and innovative output, there is sometimes no obvious or clear path forward, making group leadership especially important. Creative collaborations touch on and encompass some of the previous contexts mentioned above, namely communication, conversation, and meeting coordination. In addition, there is often an informational substrate, such as data or media, that needs to be managed in the course of the creative work. For instance, Bootlegger is a mobile app (see Fig. 11.8) that facilitates

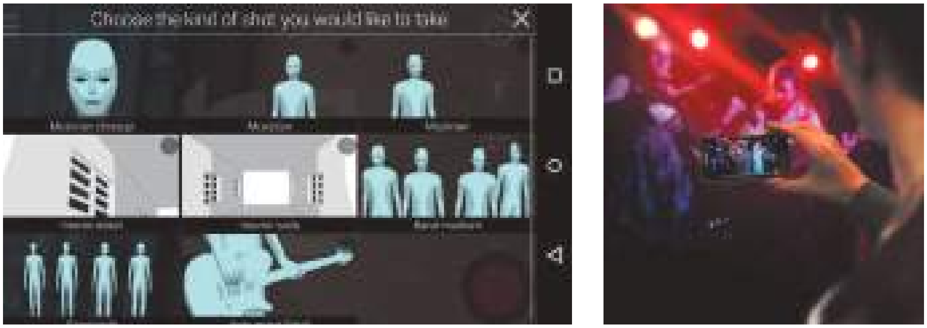


FIGURE 11.8
The Bootlegger app allows users to coordinate the creative production of videos around live events.

collaborative work around the substrate of video, coordinating its collection and editing around live events. In other applications, features like access control can become important so that some participants are only allowed to contribute at particular stages of the process or to specific tasks (Olson and Olson, 2013). Various platforms and tools exist to help facilitate these activities, from social coding platforms like GitHub to file-sharing tools like Dropbox and project management tools like Slack or Basecamp.

Because of the breadth of possibilities, collaborative creative endeavors can more or less exist almost anywhere within the design dimensions of the MoCA framework. For instance, the *scale* could be a couple software developers working on an open source widget, or it could be hundreds contributing to the Mozilla Firefox codebase. The *physical distribution* of work is often wide so that creative projects can tap talent wherever it may reside. Oftentimes the *number of communities* of practice will be greater than one in order to marshal different people with different talents, such as an animator, sound designer, and programmer working on an interactive game together. As a result, leadership and experience are often needed to coordinate different ways of working and thinking about the project (Luther et al., 2010).

11.3.5 Crowdsourcing and crowdwork

A growing number of online services exist to help people find paid work as contractors online. While there are a range of crowdwork platforms available, such as Fiverr, CrowdFlower, or TaskRabbit, a representative example of a paid crowdwork system is the Amazon Mechanical Turk platform, where "requesters" can specify a task (called a HIT which stands for Human Intelligence Task), an amount to pay, and a time frame for the task, and workers (or "Turkers" as they're often called) can browse for available tasks and sign up to complete those that they find interesting or rewarding. This allows requesters to tap labor across a wide *physical distribution*, and it allows workers flexibility to step in and out of working without having a formal employment relationship. Crowdwork is in slight contrast to other forms of crowdsourcing, which includes other activities like serious games or citizen science that motivate users to participate for reasons besides money. A popular crowdsourcing project was the New York Public Library's effort to digitize historical menus from the library's archives. It was able to transcribe 8,700 menus in just four months by placing the digitized images online and allowing visitors to click a menu item, type in what it was, and submit (see Fig. 11.9).

A model of crowdwork recently published by Kittur and colleagues (2013) articulates a number of research areas, including workflows that support task decomposition, dependence, and synthesis; task assignment to match worker abilities and skills to tasks; hierarchy to create leadership structures; latency to support real-time tasks; synchronous collaboration; quality control; job and task design to maintain interest; reputation and credentials; and motivation and rewards. The power imbalance between requesters and workers on many crowdwork platforms also suggests the need to consider ethical issues and

FIGURE 11.9

A historical 1906 menu from Fleischmann's Bakery digitized by the New York Public Library in collaboration with thousands of crowdsource volunteers who helped type in individual menu items. Note the UI list at right reflects and allows for navigation of items to be digitized.

worker rights outside of traditional employment arrangements. Crowdwork may be characterized by high *turnover* and *nascence* since workers may come and go from participating in a given task of their own accord. Such work arrangements tend toward the short-term end of *planned permanence*, as a given requester may need a specific task accomplished after which time the labor force can be disassembled easily. Some of the research challenges posited by Kittur et al. also touch on dimensions of the MoCA framework, such as how to support crowdwork that is high in *synchronicity* and how to assign tasks to a diverse labor force that may draw on a higher number of *communities of practice*.

11.3.6 Entertainment and gaming

It's human nature to seek amusement and mirth in the company of others. Many online services and communities exist to help meet the need to meet and "play" with one another, including, for instance, massive multiplayer online role-playing games (MMORPGs) with titles like *Ultima Online*, *World of Warcraft*, and *Star Wars: The Old Republic*. Whether fantasy, sci-fi, or otherwise themed, people enjoy playing such immersive games as they allow not only for the traditional fun of games, with goals, scoring and advancement, competition, and reward mechanisms, but also for social interaction and teamwork in the pursuit of goals. Each player controls a character in a virtual world and develops an identity and an avatar with a unique constellation of skills and attributes. In *World of Warcraft* (*WoW*), for instance, players form guilds that collaborate to engage in various quests, raids, and role playing (Rheingold, 2014). The guilds coordinate different members' skills in order to succeed and give guild members an opportunity to build team collaboration skills in a safe, playful environment. Early research on *WoW* studied the ways in which people engaged in guilds within the game and found that players used in-game relationships to meet new people as well as extend real-life relationships (Williams et al., 2006). The popular online comedy series *The Guild* lampoons these permeable boundaries between in-game and real-life relationships.

Interestingly, factors like the *scale* of the guild and the *turnover*—or, as Williams calls it the "guild churn"—are key in defining the nature of the interactions that occur. For instance, larger-scale guilds tend to be more goal-oriented toward game goals, whereas smaller guilds tend to be more focused on social bonds (Williams et al., 2006). The *nascence* of a guild that was developing is sometimes found to create conflicts between players with different expectations for friendliness, sharing, and leadership.

Online role-playing and guild behavior are of course just one type of, albeit very popular, online play. Other online social games include those that are integrated into social network sites like Facebook or simply provide a portal through which players can find and compete against others in classic games like poker.

11.3.7 Education

Recent years have witnessed an explosion of interest in interactive course materials available online from platforms like Coursera, Udacity, and EdX. These courses have come to be known as *massive open online courses* (MOOCs) because they often offer open enrollment and attract anywhere from 100 to 10,000 students. Online communication and collaboration systems have become common for distance education courses both as supplements to face-to-face classes as well as stand-alone offerings students can engage with to suit their ongoing learning needs. Such systems not only offer new ways for students to receive information like lectures but also enable possibilities to engage and learn from other students from across the globe, take interactive quizzes and exams, and develop collaborative class projects. One study found that a tool for arranging and guiding synchronous video discussions among culturally and geographically diverse students in a MOOC led to better learning outcomes including higher performance on quizzes and exams (Kulkarni et al., 2015). Even for campus-based courses, communication technology now provides a means for a rich, collaborative learning environment that exceeds the traditional classroom in its ability to connect students and make course materials available on an around-the-clock basis.

Online collaborative education in MOOCs defines a unique coordinate within the MoCA model. The *scale* is potentially immense, and the *physical distribution* can likewise be very broad. Because such courses attract an international audience of users who could not otherwise access educational opportunities and because there may not be a good way to enforce prerequisites, the set of interacting students on these platforms may also come from very diverse *communities of practice*. An educational course is something with a defined start and end date, which defines a distinct *planned permanence* of a few weeks or months and which in turn means that the educational community around a certain course or topic is refreshed, or *turned over,* periodically. MOOCs are an active area of research that demand more study to assess and address how education can scale effectively.

11.4 Design Considerations

There is a catalogue of features that designers might design into communication and collaboration systems. Why are some features important, and how can they support certain types of tasks or interactions? An excellent reference is Kraut and Resnick's book *Building Successful Online Communities* (2012), which lays out a series of evidence-based design claims that connect certain observable conditions of a community to certain expected outcomes. An example of such a design claim from that reference is: "Publicly displaying many examples of inappropriate behavior on the site leads members to believe such behavior is common and expected." The claim makes a specific connection between a possible design feature (i.e., displaying inappropriate behavior) and expected reception of that information by community members. However useful, such design claims can suffer the drawback of not being context-specific, and it is crucial to understand the context in which the designer is designing: not only the tasks but the diversity of participants along all of the dimensions discussed in Chapter 2, such as personality, cultural and international differences, older versus younger users, and cognitive or physical disabilities.

The remainder of this chapter examines design considerations rather than specific claims. Instead of making declarative statements about expected outcomes from a particular feature, the goal is to help with understanding why each design dimension ought to be considered and to see the connection between a feature and the tasks that might need to be accomplished across the range of contexts as articulated in the last section of this chapter. Design considerations are organized according to their impetus: cognitive factors, individual factors, and collective factors.

11.4.1 Cognitive factors

Common ground Establishing common ground—the knowledge that communicators have in common—as well as jointly understood references during communication can be essential for effective collaboration. What do users mean when they say "this button" or "that menu"? If a user is standing next to someone and points with a finger at one of the buttons in an elevator, the user can be pretty sure that's the "this" being referred to. Pointing to and referencing objects is called deictic reference, or deixis. Other forms of reference include general (e.g., "upper left"), definite (e.g., using named entities like organization or place names), or detailed (e.g., described by distinctive attributes like "the red button") (Heer and Agrawal, 2008). When users engage in mediated communication, the channel may support referencing to a greater or lesser extent and thus require different levels of effort for people to achieve common ground (Olson and Olson, 2000). For instance, in video meetings, screens are often shared so that there is a common visual reference for discussion. But it can still be easy to get lost or not understand what users are talking about if they say "this button" because there may not be feedback on the shared screen for pointing to the button that the spoken "this" refers to. In full video meetings, users will often gesture with their hands as they talk, which can provide deictic information that makes what they're saying more easily and precisely understood. On social media, referencing is supported in several standard ways, including the @username syntax, which indicates a person with the given username is being referenced, and the #hashtag syntax, which indicates a topic is being referenced,

FIGURE 11.10
References can be embedded in a tweet on Twitter in several ways, including referencing another person's account (in this example, @FILWD) as well as referencing and quoting another person's entire tweet, providing vital context and citation for the information.

as well as embedding and directly referencing other's posts (see Fig. 11.10). Interfaces often include explicit referencing via threading of comments so that it's clear who is responding to whom in a larger conversation space. In designing communication systems, it's worthwhile to consider the nature of the tasks that need to be accomplished and how different forms of referencing may need to be supported in order to make those tasks efficient—or indeed possible at all.

Social cues Beyond reference, there are a variety of other nonverbal cues that can also enhance communication, including facial expressions, gaze direction, posture, proximity, and bodily orientation (Baym, 2015). For instance, giving two thumbs-up on a video conference sends the message to collaborators that they are in full agreement with a suggestion. A furrowed brow or a smile and nod can convey a wealth of important feedback to a conversational partner who is explaining a difficult concept and wants to gauge understanding. Again, some media provide more possibilities for expressing these other channels of communication than others, yet even in "less rich" channels, like text, users adapt and develop mechanisms to convey social cues and emotions. The most popular of these are the emoticons that have become so prevalent in chat-based systems. Chat users routinely combine regular textual characters on the keyboard to indicate and convey various emotions. Research on Twitter has shown that emoticon use varies across culture, with Asian cultures favoring vertical emoticons using eye shape, (e.g., "^^" is a positive emoticon meaning "happy")

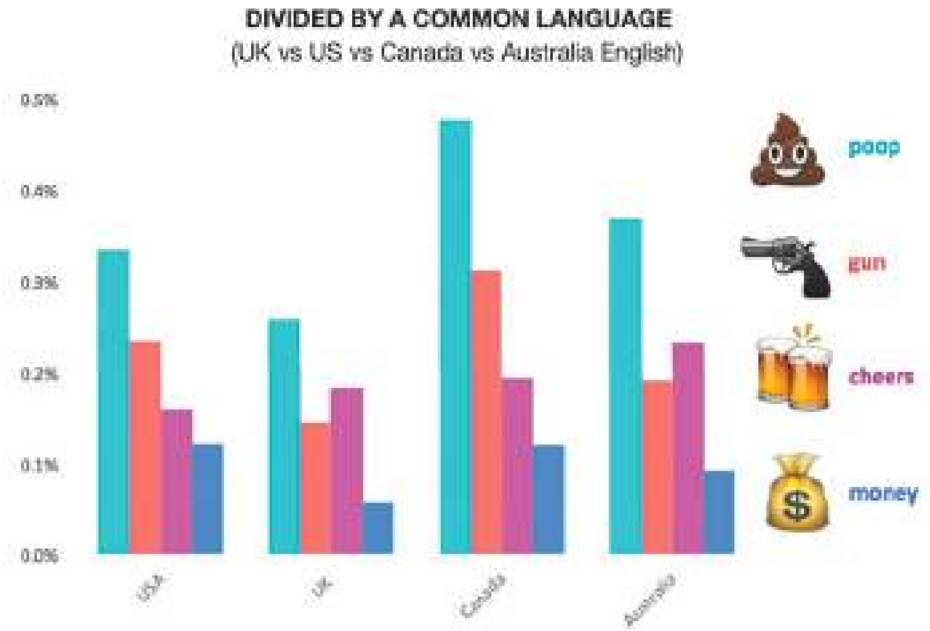


FIGURE 11.11
Emoji use varies across cultures.

and Western cultures favoring horizontal emoticons using mouth shape (e.g., ":p" is positive but with a tongue sticking out) (Park et al., 2013). Emojis are a cousin to emoticons but use actual iconographic representations of items (not just faces) to convey additional meaning and emotion in chat or social media. In an effort to respond to the desire for users to express emotional cues to one another, Facebook began experimenting in late 2015 with emoji reactions to posts that allowed for affective responses to indicate "love," "haha," "yay," "wow," "sad," and "angry." Emojis have proliferated with their integration into popular mobile phone software that allows users to type using the icons. Again, not all emoji use is the same around the world: A 2014 study from Swiftkey found that even within English, there is substantial variation in use of icons across US, UK, Canadian, and Australian users as shown in Fig. 11.11.

Activity awareness The notion of social translucence argues that making social behavior visible facilitates awareness and ultimately accountability for one's actions (Gilbert, 2012). This might include making visible information such as "who sees what," "who's done what," and "who knows that I know." For instance, collaborators often need to maintain an understanding of what others have accomplished in a joint work activity around shared artifacts (Olson and Olson, 2013). Alerts and other interface signals are used to indicate

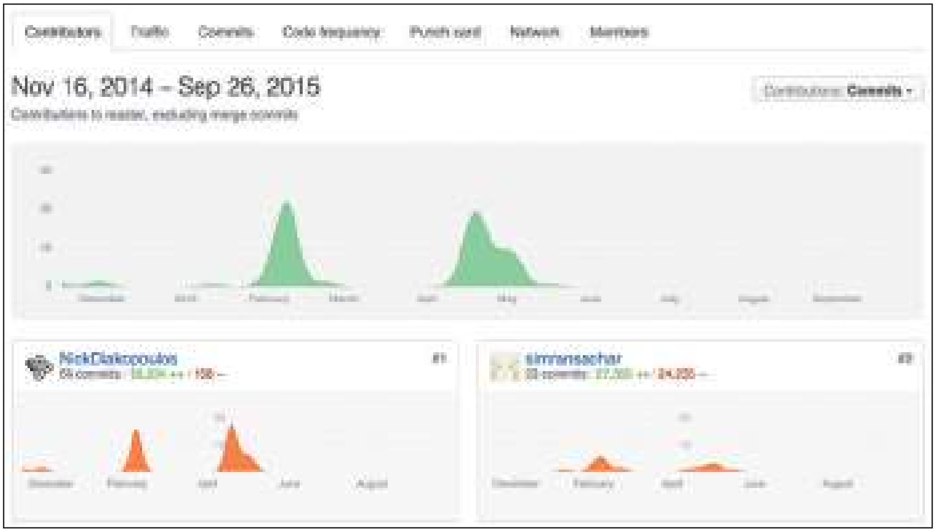


FIGURE 11.12

A Github dashboard charting project activity over time and indicating two users who are active in the project including their overall coding activity as well as volume over time. More details on activity are available by drilling into individual users.

when something has changed and by whom (e.g., "document updated on Monday at 4:23 p.m. by Andy D."). Not only does this allow collaborators to track how the work is evolving, but it also provides some level of accountability. So, for instance, if the last editor made a change that needs to be reverted, then the team knows whom to talk to about it. Awareness information has been studied extensively in collaborative software production, and research shows that cues relating to recency and volume of activity, sequence of actions over time, attention to artifacts or people, and detailed information about an action can help support social inferences like interest and level of commitment, intentions behind actions, importance to community, and personal relevance (Dabbish et al., 2012). Transparency of actions can also support learning by others who can then readily observe how a process works. On Github, a "contributors" tab on each project page offers an overview of activity and opportunities to drill into specific details of how individuals have contributed to the project (see Fig. 11.12).

Interruptions Interruptions in or by communication channels can affect their usability. For instance, during a synchronous phone call or video call, a choppy connection due to poor internet service can cause unnatural breaks in the conversational flow. This can be frustrating to recover from since pausing and turn taking are normal components to conversation, and technical glitches can make it unclear whether a pause is intended (e.g., while someone is thinking of what to say next) or imposed (e.g., by the technology). For communication technologies that tend more toward the asynchronous end of the synchronicity dimension, interruptions pose a different problem: When a new message, text, or e-mail arrives, a new alert is usually generated to notify the user. In an office environment, it's not uncommon to have the phone blinking at the same time as the phone or tablet beep and an icon starts bouncing at the bottom of the laptop screen. Research has examined the design space of interruptions and articulated various dimensions such as the symmetry of interruptions, the obtrusiveness (e.g., focal or peripheral), and the temporal gradient (e.g., historical, current, or predicted availability) among others (Hincapié-Ramos et al., 2011). Where a communication technology fits within this design space will affect how users integrate it into their workflow.

11.4.2 Individual factors

Privacy One of the issues that arises in conjunction with greater activity awareness in a system is the concomitant loss of privacy. If activity is collected implicitly based on actions within the system—rather than explicitly recorded or set by the user—this could affect the use or adoption of the system, as users may not want others to see every little action they take. In many cases, it's important to know who contributed what in a collaborative work project so that issues, corrections, or applause can be directed and so that the provenance of the result can be better understood. In more open forums like social network sites, there may also be a need for privacy in cases where users want to communicate more sensitive information only to certain connections. The idea of *context collapse* reflects the possibility that communications meant for a limited audience might in fact be visible beyond that audience. For instance, a user might not want her mother or boss to see the photo she posted at 4 a.m. from a club on Mallorca, but she could be entirely okay with her close friends seeing it. On Facebook, users can tweak their privacy settings for a number of things such as who can see a post, what people can see on their profile, whether to hide a given post on their timeline, and whether the system suggests tags for photos based on facial recognition technology. In some cases, users' privacy may be violated because of algorithmic inferences. For instance, by analyzing what users "like" on Facebook, algorithms can be used to predict a range of sensitive personal information like sexual orientation, personality traits, ethnicity, and mental health (Lee, 2014). When designing communication systems, it's important to consider situations or contexts in which users may want different amounts of privacy and to offer some degree of control, adaptability, or facility to opt out.

Identity Online communities open up a raft of questions regarding how people represent and portray themselves when people's physical bodies are not shown and text or avatars become the primary medium of communication. In an online game like *World of Warcraft*, an older man could play the character of a young woman, or a teenager could role-play as a sage and aged magician. Less-media-rich channels provide flexibility in how people choose to express their identity or identities. One of the most crucial elements to identity is the name chosen (Baym, 2015). In some cases, such as in financial transactions, real names are necessary, whereas sometimes pseudonyms (i.e., unique monikers not tied to real names) or even full anonymity is more appropriate. Some social network sites, like Facebook, have a real name policy, but oftentimes forums allow people to use pseudonyms, which allow people to have one or more identities that they can use to interact in the same community but in different ways. For instance, a well-to-do attorney in town occasionally comments on the local paper's business articles, but sometimes also really wants to trash talk the local sports team without that being tied back to his lawyer identity. Having a different pseudonym for each type of comment supports the user's needs in this case. Research has shown that certain topics on the anonymous messaging app Whisper, including NSFW ("Not Safe For Work"), LGBTQ ("Lesbian Gay Bisexual Transgender and Queer"), "Drugs and Alcohol," and "Black Markets," reflect considerably more desire for anonymity, with older users being more sensitive to the need for anonymity in these categories of content than younger users (Correa et al., 2015). The disinhibition afforded by anonymity, while it can lead to crude and anti-social behaviors and cue de-individuation and mob behavior, may also contribute to experimentation and creativity (Bernstein et al., 2011). Moreover, anonymity in online communication can reflect legitimate human needs, such as a desire to make a confession of guilt or shame, share sensitive personal health information, test an idea without fear of reputational harm, or supply information for which users may be punished or fear other retribution if their true identity would become public (Diakopoulos and Naaman, 2011).

Trust and reputation Related to identity is the notion of reputation and the ability to develop a sense of trust around that reputation. Trust can be defined as a reliance on a piece of information (or a person) and is particularly important in marketplace contexts where goods or services may be sought or exchanged. For instance, Yelp is an online listing service that helps consumers find and evaluate services or businesses that they may be interested in patronizing. It allows prior patrons to write reviews and to leave a 1-to-5-star rating. These ratings and reviews are then aggregated and presented back to others who are searching for that type of service. If users see a restaurant with 742 reviews and an average 4.5 stars, it's a pretty strong signal that they can trust that they will have a good meal there. Moreover, the interface allows the user to drill into the reviews and see individual write-ups, ratings, social activity, and feedback about other users, which can be helpful for evaluating their credibility (see Fig. 11.13). If someone who left a lousy review for a restaurant has previously written 20 other 4- and 5-star reviews, this is an indicator that the person has a reputation for writing very positive reviews and that the 1-star restaurant review must have been a spectacularly bad experience. When users engage in large-scale marketplaces

FIGURE 11.13

A user page on Yelp showing a variety of social activity context including volume of activity like reviews and photos, an "elite" badge, a graph of ratings previously made, feedback on the user's reviews (including if they were useful, funny, or cool), and other compliments. This rich information can help others understand the reliability of this user.

like this, reputation systems that track and reflect the ratings and previous behaviors of other actors become important components to communicating trustworthiness of those actors.

Motivation As in any interactive system, it is crucial to understand why people engage in collaboration and communication. There is a strong intrinsic motivation for interpersonal connectedness (Deci and Flaste, 1995), but there are a wealth of other reasons people also partake in and sustain interest in collaboration and communication, such as altruism, reciprocity, reputation or status, and habit (Preece and Shneiderman, 2009). One way to understand this is using the Uses and Gratifications framework (Ruggiero, 2000), which describes how and why active media consumers engage media in order to satisfy specific needs. The framework offers a typology of gratifications that people typically seek from media, including exposure to information, an opportunity to test their personal identity and see where it fits, a chance to interact socially, and pure amusement and entertainment. In a study of an online news commenting community, all of these gratifications were observed when participants were asked about what motivated them to either write comments or read them (Diakopoulos and Naaman, 2011). Other studies of online communities have also applied the Uses and Gratifications framework and found that motivations can shift: Gratifications sought can differ from the gratifications that are ultimately obtained (Lampe et al., 2010). Motivations can also change over time, behooving designers to understand how this affects turnover in a community. For instance, in the context of citizen science, research has shown that initial participation is motivated by things like personal interest, self-promotion, and social responsibility, whereas sustained longer-term participation relies on users receiving acknowledgment for their efforts and mentorship and being reminded of common goals (Rotman et al., 2014). In still other contexts of collaboration, like crowdwork, monetary rewards also come into play (Kittur et al., 2013). The diversity of motivations in online communities underscores the mandate for designers to understand that diversity, to conduct surveys or focus groups to better understand the value that users are receiving, and to consider universal usability in designing user experiences that accommodate different motivations.

Leadership Leadership constitutes an ability to guide and direct a group's activities. It's complicated in online scenarios because it can be harder to maintain awareness of others' activities and to develop and maintain rapport and trust (Olson and Olson, 2013). Leadership is particularly important when there is a high degree of task interdependence such that different team members are relying on each other for intermediate work products. Leaders are often responsible for developing and managing a work plan, mediating disputes or other problems as they arise, clarifying roles and objectives, making sure the right information gets to the right team members, monitoring progress and quality, and enforcing policies. Leaders are often also the members of a group who synthesize and articulate higher-level group ideas or goals and who tend to take responsibility when problems emerge (Preece and Shneiderman, 2009). Research in collaborative creativity tasks suggests that although formal leaders are present and will initiate projects, tools can support alternative leaderships styles where, for instance, leadership responsibilities can also be delegated and redistributed across the group (Luther et al., 2013). For designers of systems that enable creative production or crowdwork, careful consideration should be given to how leaders can be empowered to initiate and lead groups, accomplish the other demands of managing group work, and maintain their motivation to continue in their role.

11.4.3 Collective factors

Deviance A social norm can be defined as "a stable, shared conception of the behavior appropriate or inappropriate to a given social context, that dictates expectancies of others' behavior, and provides 'rules' for one's own behavior"

(McKirnan, 1980). Different societies, cultures, and sub-cultures may have their own social norms for what constitutes acceptable behavior within that group, but when a member of a group violates a social norm it is considered a socially deviant action. People who are known to intentionally violate group norms are often termed “trolls” and their activities are referred to as “flaming”. Trolls will post inflammatory comments that poke, prod, and antagonize other community members for their own amusement (Lee and Kim, 2015). Another form of deviant behavior is based around selfish manipulation—for instance, in online marketplaces a manipulator might create shill accounts and leave fake reviews in order to falsely hurt or help the reputation of another (Kiesler et al., 2012). In crowdwork platforms, deviant behavior might involve signing up for work and then doing just enough to make it appear as though the shoddy or rushed work is acceptable. People aren’t always perfect, and in some cases they may not be aware of the specific norms of the community they’re participating in. For these reasons, designers must be keen to consider various ways in which deviant behavior can be regulated or to make social norms more apparent and salient so that non-normative behaviors are reduced and their impacts on the community lessened.

Moderation Given that deviant behavior is to be expected in some measure within online communities, one of the approaches to cope with the issue is to have moderators evaluate contributions and take various actions on the postings. For instance, a moderator could delete a post that harasses another user, or the moderator could demote the post and make it less visible. Moderators can be professionals, as is the case for the commenting system at the *New York Times*, or they could be members of the community itself, such as on the Slashdot site. In some cases, automated text analysis algorithms are used to assess posts and determine if they use language in an unsavory and potentially inappropriate way. On the Yelp platform, algorithms are used to automatically identify reviews that may be fake in an effort to minimize their impact; the fake reviews are de-emphasized in the interface but not deleted entirely. Oftentimes community members are able to flag certain content that they believe is in violation of the community norms. These flags are then reviewed by professionals in order to make a final determination of whether the content should remain published (Diakopoulos and Naaman, 2011); however, such approaches struggle to scale for very large communities. Of course, no moderation system is perfect, and people who have their postings removed will likely want to know why. Transparency in the moderation criteria can lend legitimacy to the process so that users understand how the decision was made (Kiesler et al., 2012). Another technique that can be used to moderate a community conversation is to gag or ban users either temporarily or permanently. Sometimes a cool-down period can be an effective way for signaling to users that they need to reform their behavior.

Policies and norms Policies, rules, and norms can be important signals to users in online communities so that they know what constitutes acceptable versus unacceptable behavior and so that protocols for adjudication of moderation or other decisions are apparent. Knowing the etiquette for a given channel or community may not always be immediately apparent. Thus, policy documents are often posted in places where users can easily find them. For instance, on Reddit, a social commenting site, the Reddiquette for the site lists a variety of guidelines for behavior, including "Use proper grammar and spelling," "Look for the original source of content," and "Search for duplicates before posting." These rules of good behavior are useful for newcomers as well as existing users. Another way for users to learn about accepted norms is to observe and understand others' behavior, including which behaviors are sanctioned and which are praised. System designers can make such behaviors more or less salient to ease the learnability of the system. For instance, on the *New York Times* site, norms about the acceptability of comments are communicated by labeling exceptional comments as "Times Picks," which offers a valuable feedback signal to both the commenter as well as to the rest of the community (see Fig. 11.14). Policies and norms can be enforced either through technical regulation (i.e., the system makes it difficult to violate rules) or through social processes of regulation (i.e., rules may be broken but later sanctioned through social processes such as the moderation discussed above). The method of application of policies and rules is an important component to consider in a broader sociotechnical design of a communication technology. The way that users will behave using a tool is not just a matter of the tool's features but also of how other actors, like administrators, moderators, and other users, are perceived and act, including the ways in which policies are enforced and norms are made salient.

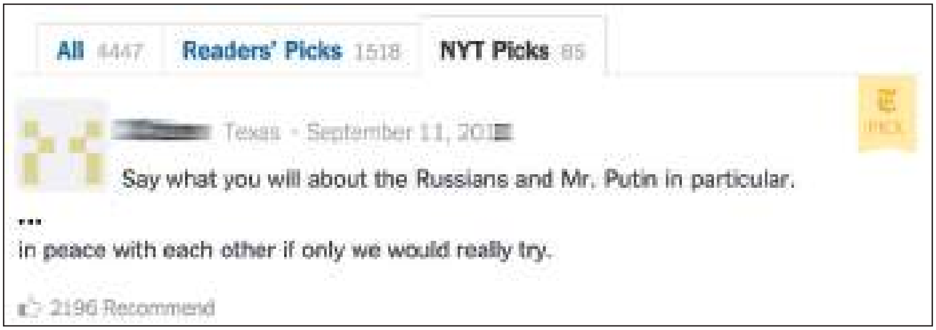


FIGURE 11.14

In the *New York Times* commenting system, moderators mark some comments as "Times Picks" with a bright yellow badge, indicating they are exceptional comments and signaling norms about what constitutes an interesting and valuable contribution to the comment thread.

Practitioner's Summary

Communication and collaboration tools are continually evolving to support human interactions across the full range of human experience. While there are many positive outcomes derived from using such tools, designers must also be aware that negative behaviors are possible and should be prepared to consider mitigating design alternatives. It is essential to understand the myriad contexts in which users may employ communication and collaboration systems from conversations to markets, meetings and creative work, entertainment, crowdsourcing, and education. Models such as MoCA can help in thinking through these various contexts during the design process to understand what may be similar or different about the particular design context being addressed. Other design considerations that can affect the user experience and usability of communication and collaboration systems include common ground, social cues, activity awareness, interruption, privacy, identity, reputation, motivation, leadership, deviance, moderation, and norms. Interface and experience design of communication tools must be an iterative endeavor as people and technology adapt and co-evolve.

Researcher's Agenda

There remains a rich variety of open questions that relate to the design and understanding of communication and collaboration tools. Perhaps most importantly, predictive theories that connect design decisions to specific outcomes still need to be developed. Designers will benefit from improved theories that can guide their work in making decisions in the contexts described in this chapter. In line with this are questions that approach the larger macro questions of organizational and societal impacts of communication and collaboration systems: How will home and work life be changed? Can such technologies restore community social capital, or will time online only increase distance from neighbors and colleagues? Will patients, consumers, and students become more or less informed and trusting? How will important social issues relating to public health, democracy, international relations, and humanitarian crises be affected? Answering such questions will require taking a long view and examining behavior at the macro level. Some of the attraction for researchers stems from this vast uncharted territory: Theories are still needed, controlled studies are difficult to arrange, and analysis of big data has its own challenges. In short, there is a grand opportunity for researchers to influence a still-emerging field and study some of the biggest questions of our time.

WORLD WIDE WEB RESOURCES

www.pearsonglobaleditions.com/shneiderman

Using communication and collaboration tools is an effective way to develop intuition for their design. Many sites and apps exist to experiment with, including:

- Facebook: http://www.facebook.com
- Twitter: http://www.twitter.com
- Reddit: http://www.reddit.com
- Slack: http://www.slack.com
- eBay: http://www.ebay.com

To get started with social network analysis, NodeXL is a powerful tool that facilitates both gathering and visualizing data:

- NodeXL: http://www.smrfoundation.org/tools/

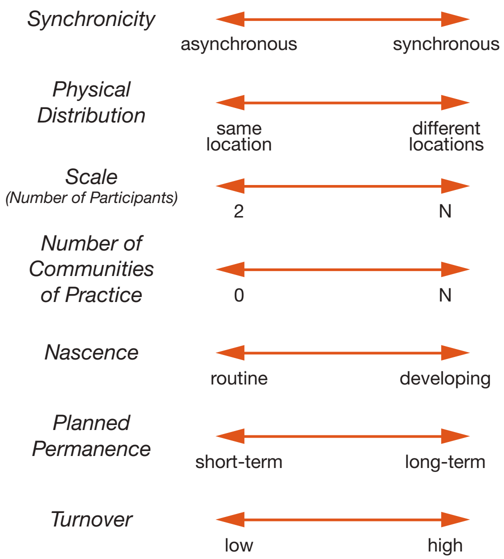
An evolving list of data collection and analysis tools for social media is curated by:

- Deen Freelon:
  http://dfreelon.org/2015/01/22/social-media-collection-tools-a-curated-list/

Discussion Questions

1. Take a position on whether you feel user interfaces for work will remain isolated or if they will become more collaborative. Present evidence to support your argument.
2. How does collaborative filtering contribute to online marketing?
3. Differentiate the roles of face-to-face encounters and collaborative interfaces. Explain the limitations and benefits of each type of communication.

4. Below are the seven dimensions of the MoCA model (Lee and Paine, 2015). Cite examples of each and how you feel it might influence a successful collaboration.



5. Explain how collaborative interfaces can improve or harm teamwork.
6. Explain how an interface designer can protect users of a collaborative interface from hostile or malicious behavior.

References

Baym, Nancy, *Personal Connections in the Digital Age*. Polity, 2nd Edition (2015).

Bernstein, M., Monroy-Hernández, A., Harry, D., André, P., Panovich K., Vargas, G. 4chan and /b/: An analysis of anonymity and ephemerality in a large online community. *Proceedings of the International Conference on Web and Social Media (ICWSM)*. (2011).

boyd, danah m., and Ellison, Nicole B., Social network sites: Definition, history, and scholarship, *Journal of Computer-Mediated Communication 13,* 1 (2007).

Bruckman, A., Luther, K., and Fiesler, C., When should we use real names in published accounts of Internet research?, in Hargittai, E., and Sandvig, C. (Editors), *Digital Research Confidential: The Secrets of Studying Behavior Online*, MIT Press (2015).

Burke, M., and Kraut, R., Using Facebook after losing a job: Differential benefits of strong and weak ties, *Proceedings of the Computer Supported Cooperative Work and Social Computing (CSCW)* (2013).

Correa, Denzil, Araújo Silva, L., Mondal, M., Benevenuto, F., and Gummadi, K. P., The many shades of anonymity: Characterizing anonymous social media content, *Proceedings of the International Conference on Web and Social Media (ICWSM)* (2015).

Deci, Edward, and Flaste, Richard, *Why We Do What We Do: Understanding Self-Motivation*, Penguin Books (1995).

Dabbish, L, Stuart, C., Tsay, J., and Herbsleb, J., Social coding in GitHub: Transparency and collaboration in an open software repository, *Proceedings of the Computer Supported Cooperative Work (CSCW)* (2012).

Diakopoulos, N., De Choudhury, M., and Naaman, M., Finding and assessing social media information sources in the context of journalism, *Proceedings of the ACM Conference on Human Factors in Computing Systems*, ACM Press, New York (2012).

Diakopoulos, N., and Naaman, M., Towards quality discourse in online news comments, *Proceedings of the Conference on Computer Supported Cooperative Work (CSCW)* (2011).

Diakopoulos, N., Picking the NYT picks: Editorial criteria and automation in the curation of online news comments, *#ISOJ Journal* (2015).

Ellison, N., Steinfield, C., and Lampe, C., The benefits of Facebook "friends:" Social capital and college students' use of online social network sites, *Journal of Computer-Mediated Communication (JCMC) 12* (2007).

Gilbert, E., Designing social translucence over social networks, *Proceedings of the ACM Conference on Human Factors in Computing Systems*, ACM Press, New York (2012).

Hansen, D., Shneiderman, B., and Smith, M., *Analyzing Social Media Networks with NodeXL: Insights from a Connected World*, Morgan Kaufmann (2011).

Heer, J., and Agrawal, M., Design considerations for collaborative visual analytics, *Information Visualization 7* (2008), 49–62.

Hincapié-Ramos, J. D., Voida, S., and Mark, G., A design space analysis of availability-sharing systems, *Proceedings of the Symposium on User Interface Software and Technology (UIST)* (2011).

Hullman, J., Diakopoulos, N., Momeni, E., and Adar, E., Content, context, and critique: Commenting on a data visualization blog, *Proceedings of the Computer Supported Cooperative Work and Social Computing (CSCW)* (2015).

Kiesler, S., Kraut, R., Resnick, P., and Kittur, A., Regulating behavior in online communities, in Kraut, R., and Resnick, P. (Editors), *Building Successful Online Communities: Evidence-Based Social Design*, MIT Press (2012).

Kittur, A., Nickerson, J., Bernsetin, M., Gerber, E., Shaw, A., Zimmerman, J., Lease, M., and Horton, J., The future of crowd work, *Proceedings of the Computer Supported Cooperative Work and Social Computing (CSCW)* (2013).

Kooti, Farshad, Yang, Haeryun, Cha, Meeyoung, Gummadi, Krishna P., and Mason, Winter A., The emergence of conventions in online social networks, *Proceedings of the International Conference on Web and Social Media (ICWSM)* (2012).

Kramer, A., Guillory, J., and Hancock, J., Experimental evidence of massive-scale emotional contagion through social networks, *Proceedings of the National Academies of Science (PNAS) 111*, 24 (2014).

Kraut, R., and Fiore, A., The role of founders in building online groups, *Proceedings of the Computer Supported Cooperative Work and Social Computing (CSCW)* (2014).

Kraut, R., and Resnick, P., *Building Successful Online Communities: Evidence-Based Social Design*, MIT Press (2012)

Kuklarni, C., Cambre, J., Kotturi, Y., Bernstein, M., and Klemmer, S., Talkabout: Making distance matter with small groups in massive classes, *Proceedings of the Computer Supported Cooperative Work and Social Computing (CSCW)* (2015).

Lampe, C., Wash, R., Velasquez, A., and Ozkaya, E., Motivations to participate in online communities, *Proceedings of the ACM Conference on Human Factors in Computing Systems*, ACM Press, New York (2010).

Lee, Charlotte, and Paine, Drew, From the matrix to a model of coordinated action (MoCA): A conceptual framework of and for CSCW, *Proc. Computer Supported Cooperative Work and Social Computing (CSCW)* (2015).

Lee, Newton, *Facebook Nation: Total Information Awareness*, 2nd Edition, Springer (2014).

Lee, So-Hyun, and Kim, Hee-Woong, Why people post benevolent and malicious comments online, *Communications of the ACM 58*, 11 (November 2015).

Leetaru, K., *Data Mining Methods for the Content Analyst: An Introduction to the Computational Analysis of Content*, Routledge Communication Series (2011).

Linden, G., Smith, B., and York, J., Amazon.com recommendations: Item-to-item collaborative filtering, *IEEE Internet Computing 7*, 1 (2003).

Lofland, L., and Lofland, L., *Analyzing Social Settings: A Guide to Qualitative Observation and Analysis*, 4th Edition, Wadsworth Publishing (2005).

Luther, K., Caine, K., Ziegler, K., and Bruckman, A., Why it works (when it works): Success factors in online creative collaboration, *Proceedings of the ACM Conference on Supporting Group Work (GROUP)* (2010).

Luther, K., Fiesler, C., and Bruckman, A., Redistributing leadership in online creative collaboration, *Proceedings of the Computer Supported Cooperative Work and Social Computing (CSCW)* (2013).

McKirnan, D., The identification of deviance: A conceptualization and initial test of a model of social norms, *European Journal of Social Psychology 10*, 1 (1980), 75–93.

Morgan, J., Bouterse, S., Walls, H., and Stierch, S., Tea and sympathy: Crafting positive new user experiences on Wikipedia, *Proceedings of the Computer Supported Cooperative Work* (CSCW) (2013).

Olson, Gary, M., and Olson, Judith S., Distance matters, *Human-Computer Interaction 15,* 2 (2000), 139–178.

Olson, Judith S., and Olson, Gary M., *Working Together Apart: Collaboration over the Internet,* Synthesis Lectures on Human-Centered Informatics (Ed. John M. Carroll) (2013).

Park, J., Barash, V., Fink, C., Cha, M. Emoticon Style: Interpreting differences in emoticons across cultures. *Proceedings of the International Conference on Web and Social Media (ICWSM)*. (2013).

Phua, J., Participating in health issue-specific social networking sites to quit smoking: How does online social interconnectedness influence smoking cessation self-efficacy? *Journal of Communication 63* (2013), 933–952.

Preece, J., and Shneiderman, B., The Reader-to-Leader Framework: Motivating technology-mediated social participation, *AIS Transactions on Human-Computer Interaction 1*, 1 (2009).

Rae, I., Venolia, G., Tang, J., and Molnar, D., A framework for understanding and designing telepresence, *Proceedings of the Computer Supported Cooperative Work and Social Computing (CSCW)* (2015).

Rainie, Lee, and Wellman, Barry, *Networked: The New Social Operating System*, MIT Press (2012).

Rheingold, Howard, *Net Smart: How to Thrive Online*, The MIT Press (2014).

Riffe, D., Lacy, S., and Fico, F., *Analyzing Media Messages: Using Quantitative Content Analysis in Research*, 3rd Edition, Routledge Communication (2013).

Ronson, J., *So You've Been Publicly Shamed*, Riverhead Books (2015).

Rotman, D., Hammock, J., Preece, J., Hansen, D., Boston, C., Bowser, A., and He, Y., Motivations affecting initial and long-term participation in citizen science projects in three countries, *Proc. iConference* (2014).

Ruggiero, T., Uses and Grats theory in the 21st century, *Mass Communication & Society 3,* 1 (2000).

Smith, M. A., Rainie, L., Himelboim, I., and Shneiderman, B., Mapping Twitter topic networks: From polarized crowds to community clusters, *Pew Research Center Report* (2014).

Smith, A., and Page, D., U.S. smartphone use in 2015, *Pew Research Center Report* (2015).

Turkle, S., *Reclaiming Conversation: The Power of Talk in a Digital Age*, Penguin Press (2015).

Vieweg, S., Hughes, A. L., Starbird, K., and Palen, L., Microblogging during two natural hazards events: What Twitter may contribute to situational awareness, *Proceedings of the Conference on Human Factors in Computing Systems (CHI)* (2010).

Vitak, J., Shilton, K., and Ashktorab, Z., Beyond the Belmont Principles: Ethical challenges, practices, and beliefs in the online data research community, *Proceedings of the Computer Supported Cooperative Work and Social Computing (CSCW)* (2016).

Wenger, Etienne, *Communities of Practice: Learning, Meaning, and Identity*, Cambridge University Press (1998).

Williams, D., Ducheneaut, N., Xiong, L., Zhang, Y., Yee, N., and Nickell, E., From tree house to barracks: The social life of guilds in *World of Warcraft*, *Games and Culture 1* 4 (2006).

PART 4

Design Issues

PART OUTLINE

Design issues can both complicate and enhance the user experience. The designer has determined what combination of interaction styles to use and followed a development process that yielded a preliminary version of the user interface, obtaining feedback along the way. The designer is ready to make modifications and deliver features that will add value to the design and increase the likelihood of system acceptance and success.

The ultimate goal is to improve the user experience. Chapter 12 describes functional issues within user experience evaluation criteria but also leaves room for varying styles to suit a variety of customers: display design, window design, webpage design, color, nonanthropomorphic design, and error messages. Chapter 13 takes the user experience a step further and illustrates that time sensitivity is key to a successful user-interface design—it is human nature that sluggish system responses can lead to frustration and errors. This chapter presents considerations to ensure system response times meet or exceed user needs.

Chapter 14 shows the importance of documentation and user support. Although there is a movement away from packaged volumes of print documentation and no place to put the printed documentation with mobile devices, there is still a need for well-written and comprehensive help in alternative formats including online. With the correct set of tools, users can rapidly master system features new to them, ultimately improving their user experience.

How information is presented and managed in a user-interface design is critical to its success. Being able to find what you want and quickly understand the depth of data behind that which is displayed on the screen improves user productivity and can make using the system fun and enjoyable. Chapter 15 first illustrates the importance of strong search capability. It includes techniques for searching text, databases, and multimedia. Chapter 16 illustrates how the presentation of data in graphical form can take advantage of human perception characteristics to easily grasp what data are available and understand the data. A taxonomy on how to structure information visualization design is presented. Ultimately, users can easily manipulate the data without having to re-process or re-structure those data on their own.

CHAPTER 12

Advancing the User Experience

> Words are sometimes sensitive instruments of precision with which delicate operations may be performed and swift, elusive truths may be touched.
>
> **Helen Merrell Lynd**
> *On Shame and the Search for Identity*

CHAPTER OUTLINE

12.1 Introduction

Interface design is edging closer to match the art, trendiness, and techniques taught in design schools. In an era of smartphones, tablets, the thinnest of laptops, and wearables, competition over design has intensified. Early automobiles were purely functional, and Henry Ford could joke about customers getting any color as long as it was black, but modern car designers have learned to balance function and fashion. This chapter deals with seven design matters that are functional issues within user experience (UX) evaluation criteria but also leave room for varying styles to suit a variety of customers: display design, view (window) management, animation, webpage design, color, nonanthropomorphic design, and error messages.

Another opportunity for design improvements lies in the layout of information on a display (Section 12.2). Cluttered displays may overwhelm even knowledgeable users, but with only modest effort, designers can create well-organized, information-abundant layouts that reduce search time and increase subjective satisfaction. Issues related to universal usability, user-generated content, and the proliferation of web design and development techniques are also addressed.

View (window) management has become more standardized, but an understanding of the motivations for multiple-window coordination could lead to improvements and to novel proposals (Section 12.3). Animation enhances the user experience when used wisely (Section 12.4). Webpage designs are improving as standards and tools emerge to address webpage design and development, user-generated content, and universal usability (Section 12.5).

High-resolution color displays, large or small, offer many possibilities and challenges for designers. Guidelines for color design are useful, but experienced designers know that repeated testing is needed to ensure success (Section 12.6).

Messages are sometimes meant to be conversational, as modeled by human-human communication, but this strategy has limits because people are different from computers. This fact may be obvious, but a section on nonanthropomorphic design (Section 12.7) seems necessary to steer designers toward comprehensible, predictable, and controllable interfaces.

User experiences with computer-system prompts, explanations, error diagnostics, and warnings play a critical role in influencing acceptance of software systems. The wording of messages is especially important in systems designed for novice users, but experts also benefit from improved messages (Section 12.8).

Recognition of the creative challenge of balancing function and fashion might be furthered by having designers put their names and photos on a title or credits page, just as authors do in a book. Such acknowledgment is common in games and in some educational software, and it seems appropriate for all software. Credits provide recognition for good work and identify the people responsible. Having their names in lights may also encourage designers to work even harder, since their identities will be public.

See also:

Chapter 6, Design Case Studies
Chapter 8, Fluid Navigation
Chapter 13, The Timely User Experience
Chapter 14, Documentation and User Support (a.k.a. Help)

12.2 Display Design

For most interactive systems, the displays are a key component of successful designs (Fig. 12.1) and are the source of many lively arguments. Dense or cluttered displays can provoke anger, and inconsistent formats can inhibit performance. The complexity of this issue is suggested by the 162 classic guidelines

FIGURE 12.1
Webby Award winner NASA climate website illustrating an elegant visual design.

(see Box 12.1) for data display offered by Smith and Mosier (1986). Display design will always have elements of art and require invention, but perceptual principles are becoming clearer, and theoretical foundations have emerged (Galitz, 2007; Johnson, 2014). Innovative information visualizations with user interfaces to support dynamic control are a rapidly emerging theme (Chapter 16).

Designers should begin, as always, with a thorough knowledge of the users' tasks, free from the constraints of display size or available fonts. Effective display designs must provide all the necessary data in the proper sequence to carry out the task. Meaningful groupings of items (with labels suitable to the users' knowledge), consistent sequences of groups, and orderly formats all support task performance. The Gestalt Laws of Perception (rules of the organization of perceptual scenes) apply here and are discussed further in Johnson (2014). Groups can be surrounded by blank spaces or boxes. Alternatively, related items can be indicated by highlighting, background shading, color, or special fonts. Within a group, orderly formats can be accomplished by left or right justification, alignment on decimal points for numbers, or markers to decompose lengthy fields.

Graphic designers, originally working in a world of print media, adapted principles for display design. Mullet and Sano (1995) offer thoughtful advice with examples of good and bad design in commercial systems. They propose six categories of principles that reveal the complexity of the designer's task:

1. *Elegance and simplicity:* Unity, refinement, and fitness
2. *Scale, contrast, and proportion:* Clarity, harmony, activity, and restraint
3. *Organization and visual structure:* Grouping, hierarchy, relationship, and balance
4. *Module and program:* Focus, flexibility, and consistent application
5. *Image and representation:* Immediacy, generality, cohesiveness, and characterization
6. *Style:* Distinctiveness, integrity, comprehensiveness, and appropriateness

As expected, a mobile (small) screen display requires special treatment due to display real estate available. One study looks at phones and tablets and how the user experience varies among these devices. For example, the convenience of touchscreens and portability must be traded off with smaller screen size, connectivity issues, wide variability in app design, and one window open on a smartphone. Nielsen Norman Group (2015) published a detailed definition of design guidelines for the mobile user experience.

This section continues with some additional display design issues, offering empirical support for concepts where available. Keep in mind that there is not consensus on how user experience is measured (Law, 2011).

BOX 12.1

Examples of the 162 data-display guidelines from Smith and Mosier (1986). (Courtesy MITRE Corporate Archives: Bedford, MA).

- Ensure that any data that a user needs, at any step in a transaction sequence, are available for display.
- Display data to users in directly usable forms; do not require that users convert displayed data.
- Maintain a consistent format for any particular type of data display from one display to another.
- Use short, simple sentences.
- Use affirmative statements rather than negative statements.
- Adopt a logical principle by which to order lists; where no other principle applies, order lists alphabetically.
- Ensure that labels are sufficiently close to their data fields to indicate association yet are separated from their data fields by at least one space.
- Left-justify columns of alphabetic data to permit rapid scanning.
- Label each page in multi-paged displays to show its relation to the others.
- Begin every display with a title or header, describing briefly the contents or purpose of the display; leave at least one blank line between the title and the body of the display.
- For size coding, make larger symbols be at least 1.5 times the height of the next-smaller symbol.
- Consider color coding for applications in which users must distinguish rapidly among several categories of data, particularly when the data items are dispersed on the display.
- For a large table that exceeds the capacity of one display frame, ensure that users can see column headings and row labels in all displayed sections of the table. Note that one study at the University of Texas published guidelines for tables to reduce any patient safety impacts when using large, tabular displays of patient data (University of Texas, 2013).
- Provide a means for users (or a system administrator) to make necessary changes to display functions, as data-display requirements may change (as is often the case).

12.2.1 Field layout

Exploration with a variety of layouts can be a helpful process. These design alternatives should be developed directly on a display screen. An employee record with information about a spouse and children could be displayed crudely as follows:

Poor:
```
TAYLOR,SUSAN      34787331      WILLIAM TAYLOR
THOMAS            10292014
ANN               08212015
ALEXANDRA         09082012
```

This record may contain the necessary information for a task, but extracting the information will be slow and error-prone. As a first step at improving the format, blanks and separate lines can distinguish fields:

Better:
```
TAYLOR,SUSAN        34787331          WILLIAM TAYLOR
THOMAS              10292014
ANN                 08212015
ALEXANDRA           09082012
```

The children's names can be listed in chronological order, with the dates aligned. Familiar separators for the dates also aid recognition:

Better:
```
TAYLOR,SUSAN        34787331          WILLIAM TAYLOR
ALEXANDRA           09-08-2012
THOMAS              10-29-2014
ANN                 08-21-2015
```

The reversed order of "last name, first name" for the employee may be desired to highlight the lexicographic ordering in a long file, but the "first name, last name" order for the spouse is more readable. Consistency is important, however, so a compromise might be made:

Better:
```
SUSAN TAYLOR        34787331          WILLIAM TAYLOR
ALEXANDRA           09-08-2012
THOMAS              10-29-2014
ANN                 08-21-2015
```

For frequent users, this format may be acceptable, since labels have a cluttering effect. For most users, however, labels will be helpful. Indenting the information about children will also help to convey the grouping of these repeating fields:

Better:
```
Employee:  SUSAN TAYLOR  ID Number: 34787331
Spouse:    WILLIAM TAYLOR
Children:  Names         Birthdates
           ALEXANDRA     09-08-2012
           THOMAS        10-29-2014
           ANN           08-21-2015
```

Mixed upper- and lowercase letters have been used for the labels to distinguish them from the record information, but the coding might be switched to use boldface and mixed upper- and lowercase for the contents. The employee name and ID number can also be placed on the same line to tighten up the display:

Better:
```
Employee:  Susan Taylor  ID Number: 34787331
Spouse:    William Taylor
Children:  Names         Birthdates
           Alexandra     09-08-2012
           Thomas        10-29-2014
           Ann           08-21-2015
```

Finally, logical groupings can be created by using shading or borders to delineate sets of related information:

Better:

```
Employee:  Susan Taylor ID Number: 34787331
Spouse:    William Taylor
```

```
Children:  Names        Birthdates
           Alexandra    09-08-2012
           Thomas       10-29-2014
           Ann          08-21-2015
```

For an international audience, the date format might need to be clarified (month-day-year). Even in this simple example, the possibilities are numerous. Further improvements could be made with other coding strategies, such as the use of background shading, color, or graphic icons. In any situation, a variety of designs should be explored. An experienced graphic designer can be a great benefit to the design team. Pilot testing with prospective users can yield subjective satisfaction scores, objective times to complete tasks, and error rates for a variety of proposed formats.

12.2.2 Empirical results

Guidelines for display design were an early topic in human-computer interaction research because of the importance of displays in control-room and life-critical applications (Section 3.3). As technology evolved and high-resolution graphical color displays became available, new empirically validated guidelines became necessary. Then web-based markup languages, user-generated content, and the need to accommodate older adults and provide universal usability presented further design challenges. User control of font size, window size, and brightness meant designers had to ensure that the information architecture could be understood, even as some display elements changed. Now, as small-, wall-, and mall-sized displays have opened further possibilities, there is again renewed interest in display design.

Early studies with alphanumeric displays laid the foundation for design guidelines and predictive metrics. These studies clearly demonstrated the benefits of eliminating unnecessary information, grouping related information, and emphasizing information relevant to required tasks. Simple changes could cut task performance times almost in half.

Expert users can deal with dense displays and may in fact prefer these displays because they are familiar with the format and they must initiate fewer actions. Performance times are likely to be shorter with fewer but denser displays than with more numerous but sparse displays. This improvement will be especially noticed if tasks require comparison of information across displays. Systems for stock-market data, air-traffic control, and airline reservations are

FIGURE 12.2

U.S. Navy air-traffic-control work environment, with multiple, specialized, data-intensive displays.

examples of successful applications that have dense packing, multiple displays, and highly coded fields (see Fig. 12.2).

Award-winning designs, for example in websites and mobile apps, can be exciting, captivating, and efficient to use. One NASA site (Fig. 12.1) illustrates such a design. The site won a Webby Award (NASA, 2015). Groups that select award-winning designs are worth noting, and their websites contain many examples. These award organizations include the Webbys (http://webbyawards.com) and the Awwwards (http://www.awwwards.com - yes, with WWW as part of its name).

12.2.3 Sequence of displays

Sequences of displays should be similar throughout the system for similar tasks, but exceptions will certainly occur. Within a sequence, users should be offered some sense of how far they have come and how far they have to go to reach the end (Figs. 12.3 and 12.4). It should be possible to go backward in a sequence to correct errors, to review decisions, or to try alternatives.

Relevant to display design is a discussion on scrolling. For example, scrolling on smaller screens (mobile and tablet) is a constant challenge. The next section discusses how to manage multiple views and windows. See also Chapter 8 (Fluid Navigation) for additional discussion on scrolling.

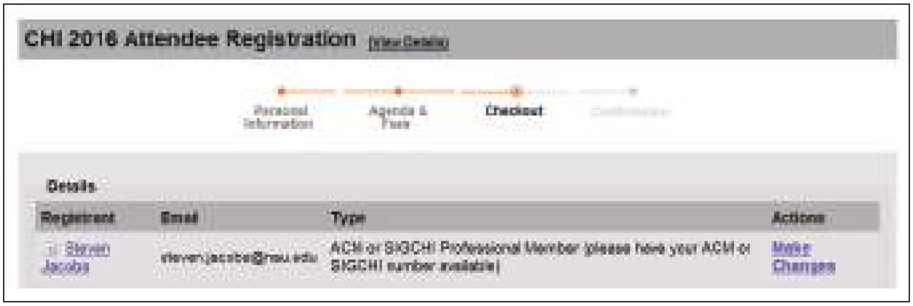


FIGURE 12.3

This page from ACM SIGCHI CHI 2016 web site allows users to register for the conference. The progress indicator in the middle indicates the user is working in the 3rd step of a 4-step sequence of displays, giving users a sense of how far they have gone.

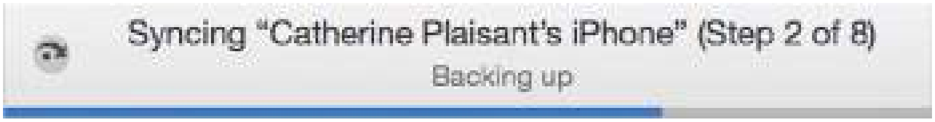


FIGURE 12.4

An example of a progress indicator with the status of a backup process in iTunes.

12.3 View (Window) Management

Computer users frequently have to consult documents, forms, e-mail messages, webpages, and more to complete their tasks. Designers have long struggled with strategies to offer users sufficient information and flexibility to accomplish their tasks while reducing window-housekeeping actions and minimizing distracting clutter. If users’ tasks are well understood and regular, there is a good chance that an effective *multiple-window display* strategy can be developed.

If window-housekeeping actions can be reduced, users can complete their tasks more rapidly and probably with fewer mistakes. The visual nature of window use has led many designers to apply direct-manipulation strategies (Chapter 7) to window actions. To stretch, move, and scroll a window, users can point at appropriate icons on the window border and simply click on the mouse button and drag. Since the dynamics of windows have a strong effect on user perceptions, the animations for transitions (zooming boxes, repainting when a window is opened or closed, blinking outlines, or highlighting during dragging) must be designed carefully.

Window design evolved rapidly in the 1980s from influential designs at Xerox PARC to innovative syntheses by Apple (Figs. 1.1 and 1.2) and finally Microsoft's modest refinements, which led to the highly successful Windows series (Fig. 1.3). Overlapping, draggable, resizable windows on a broad desktop have become the standard for most users. Advanced users who work on multiple tasks can switch among collections of windows called "workspaces" or "rooms"; each workspace holds several windows whose states are saved, allowing easy resumption of activity. Much progress has been made, but there is still an opportunity to reduce dramatically the housekeeping chores tied to individual windows and to provide task-related, multiple-window coordination.

12.3.1 Coordinating multiple views (windows)

Designers may break through to the next generation of window managers by developing *coordinated windows*: windows that appear, change contents, resize automatically, and close as a direct result of user actions in the task domain. For example, in a medical insurance claims-processing application, when the agent retrieves information about a client, such fields as the client's address, telephone number, and membership number should be automatically filled in on the display. Simultaneously, and with no additional commands, the client's medical history might appear in a second window, and the record of previous claims might appear in a third window. A fourth window might contain a form for the agent to complete to indicate payment or exceptions. Scrolling the medical-history window might produce a synchronized scroll of the previous-claims window to show related information. When the claim is completed, all window contents should be saved and all the windows should be closed with one action. Such sequences of actions can be established by designers or by users with end-user programming tools.

Similarly, for web browsing, job-hunting users should be able to select the five most interesting position-description links and open them all with a single click. Then it should be possible to explore all of them synchronously to compare the job details (description, location, salary, etc.) using one scrolling action. When one position is selected, it should fill the screen, and the other four should close automatically.

Coordination is a task concept that describes how information objects change based on user actions. A careful study of user tasks can lead to the development of task-specific coordinations based on sequences of actions. The especially interesting case of work with large images such as maps, circuit diagrams, or magazine layouts is covered in the next section. Other important coordinations that might be supported by interface designers include:

- *Synchronized scrolling.* A simple coordination is synchronized scrolling, in which the scroll bar of one window is coupled to another scroll bar, and action on one scroll bar causes the other window's contents to scroll in parallel. This technique is useful for comparing two versions of a program or document. Synchronization might be on a line-for-line basis, on a proportional basis, or keyed to matching tokens in the two windows.
- *Hierarchical browsing.* Coordinated windows can be used to support hierarchical browsing (Fig. 12.5). For example, if one window contains a book's table of contents, selection of a chapter title by a pointing device should lead to the display, in an adjoining window, of that chapter's contents. Hierarchical browsing has been integrated into Windows File Explorer to allow users to browse hierarchical directories, into Outlook to enable browsing of folders of e-mails, and into many other applications.
- *Opening/closing of dependent windows.* One option on opening a window is to simultaneously open dependent windows in a nearby and convenient location. For example, when a user browsing a program opens a main software procedure, the dependent set of procedures could open up automatically,

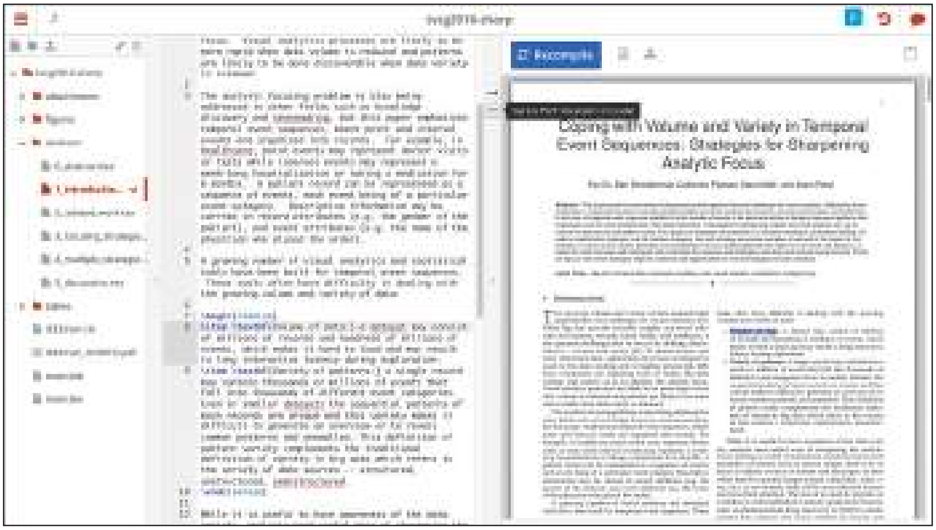


FIGURE 12.5

ShareLaTeX allows users to edit a structured LaTeX document and see the resulting formatted document. On the left is the hierarchical list of document sections. The "1. Introduction" section is selected and highlighted in red, and its text can be edited in the middle. The preview of the output is shown on the right. After selecting a passage in one view, it is possible to see the corresponding location on the other view.

for example, in a development tool environment. Similarly, when filling in a form, a user might automatically be presented with a dialog box listing a choice of preferences. That dialog box might lead the user to activate a pop-up or error-message window, which in turn might lead to an invocation of the help window. After the user indicates the desired choice in the dialog box, it would be convenient to have automatic closing of all the windows.

- *Saving/opening of window state.* A natural extension of saving a document or a set of preferences is to save the current state of the display with all the windows and their contents.
- *Tabbed browsing.* Browser tabs allow users to view multiple webpages in the same browser without the need to open a new browser session.
- *Tiled or overlapping windows.* Windows can automatically be resized and arranged so that they do not overlap each other. Overlapping windows are sometimes referred to as *overlaid* or *cascading windows* (Fig. 12.6).
- *Ribbon interface.* The Microsoft Office interface is designed to make it easier for users to find the features they need to get their work done. Microsoft calls this its "Fluent" user interface.

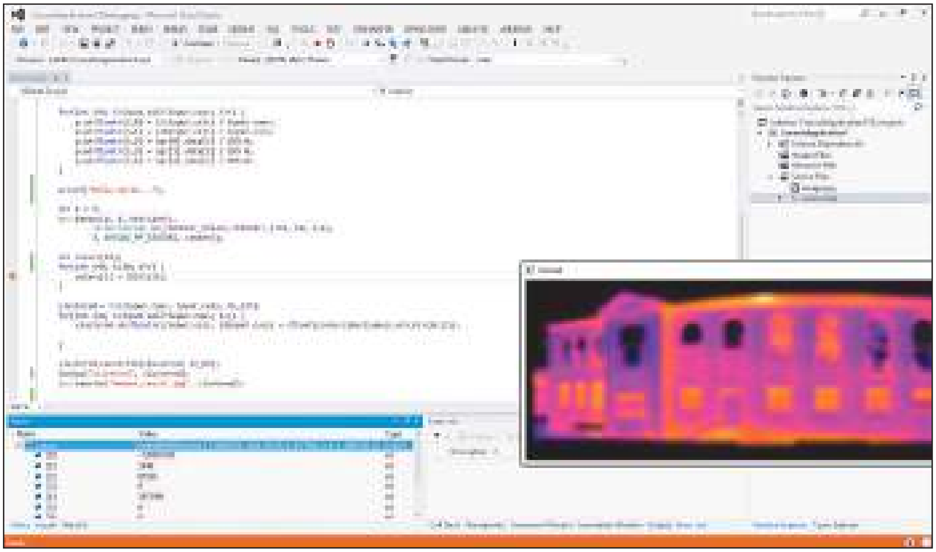


FIGURE 12.6

Example of Visual Studio Integrated Development Environment (IDE) illustrating coordinated window views. The project files are listed on the right in a hierarchical browser. The selected file (i.e., "source .cpp") is highlighted in the list and displayed on the left. A breakpoint was set in the code (red dot) at the line starting with "color[i]" so the bottom left window shows the values of the color array of the breakpoint. All windows are titled, except for the output window (the colorful image of heat sensor data), which overlaps all the other windows.

- *Design patterns.* Managing interface design elements and design patterns are described in Chapter 4 and in Tidwell (2011). In software design, design patterns are a reusable solution to a commonly occurring problem within a given context.
- *Start menus.* Like the Windows Start menu, the start-up process for software or devices needs to be easy to use (restarts and shutdowns, too).

There are many interface schemes that allow users to work in and move between focused and contextual views of screens. These interface schemes have been studied and categorized according to the interface mechanisms used to separate and blend views (Cockburn et al., 2009). The four approaches are spatial separation, typified by overview + detail interfaces; temporal separation, typified by zoomable interfaces; seamless focus + context, typified by fisheye views; and cue-based techniques that selectively highlight or suppress items within the information space.

12.3.2 Browsing large views

A 2-dimensional cousin of hierarchical browsing is image browsing, which enables users to work with large maps, circuit diagrams, magazine layouts, photos, or artwork. Users see an overview in one window and the details in a second window. They can move a field-of-view box in the overview to adjust the detail-view content. Similarly, if users pan in the detail view, the field-of-view box should move in the overview. Well-designed coordinated windows have matching aspect ratios in the field-of-view box and the detail view, and a change to the shape of either produces a corresponding change in the other.

Similarly, examples appear in Google and MapQuest maps. The magnification from the overview to the detail view is called the *zoom factor*. When the zoom factors are between 5 and 30, the coordinated overview and detail view pair are effective; for larger zoom factors, however, an additional intermediate view is needed. For example, if an overview shows a map of France, a detail view showing the Paris region is effective. However, if the overview were of the entire world, intermediate views of Europe and France would preserve orientation (Fig. 12.7).

Side-by-side placement of the overview and detail views is the most common layout, since it allows users to see the big picture and the details at the same time. However, some systems provide a single view, either zooming smoothly to move in on a selected point (Bederson, 2011) or simply replacing the overview with the detail view. This zoom-and-replace approach is simple to implement and gives the maximal screen space for each view, but it denies the users the chance to see the overview and detail view at the same time. A variation is to have the detail view overlap the overview, but it may obscure key items. Semantic zooming, in which the way objects are represented changes depending on their magnification, might help users see an overview by rapidly zooming in and out (Hornbaek et al., 2002).

Attempts to provide detail views (focus) and overviews (context) without obscuring anything have motivated interest in *fisheye views* (Sarkar and Brown, 1994; Bartram et al., 1995; Baudisch et al., 2004). The focus area (or areas) is

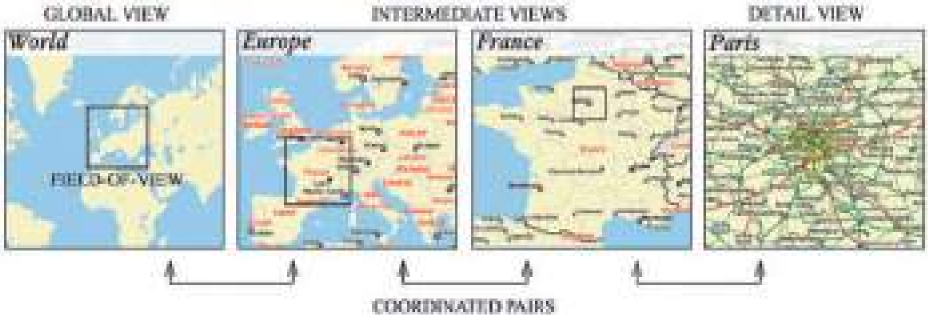


FIGURE 12.7

Global and intermediate views, which provide overviews for the detail view of Paris. Movements of the field-of-view boxes change the content in the detail view (Plaisant et al., 1995).

magnified to show detail while preserving the context, all in a single display (Fig. 12.8). This distortion-based approach is visually appealing, even compelling, but the continually changing magnified area may be disorienting, and the zoom factor in published examples rarely exceeds 5.

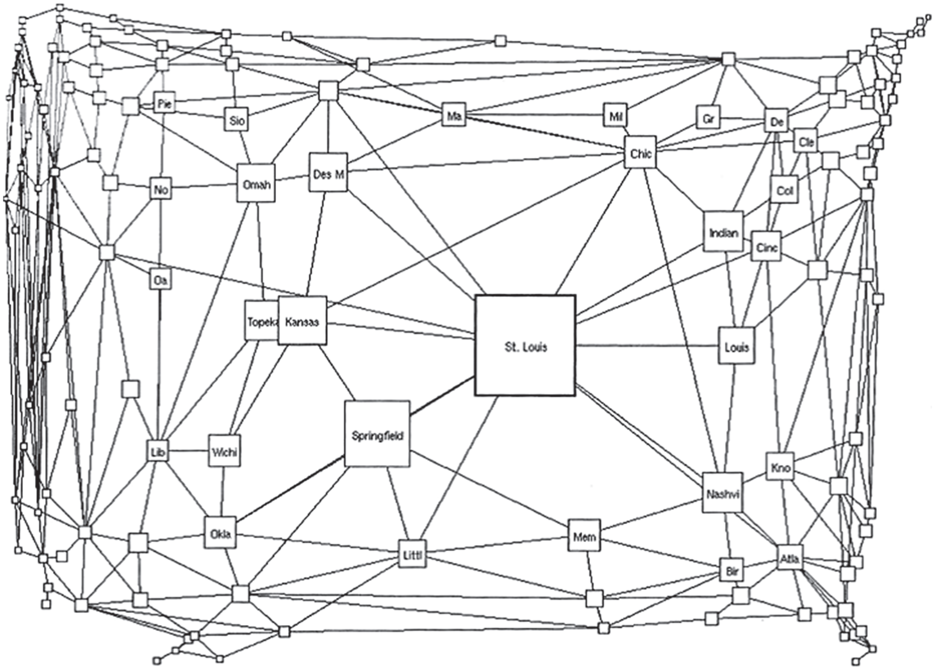


FIGURE 12.8

Fisheye view of U.S. cities, with the focus on St. Louis. The context is preserved, although the distortions can be disorienting (Sarkar and Brown, 1994).

The bifocal display (similar to the fisheye approach) was arguably the first demonstration of the use of distortion to provide a focus + context view of information. A user can focus on one or two documents, for example, and at the same time have an overall view of the whole of the information space (Spence, 2001).

One success story with a fisheye view approach is the Mac OS X Dock (Fig. 1.2), a menu of program and file icons that appears typically at the bottom of the screen. Scrolling over an icon magnifies the item to highlight readiness for that icon (application/file) to be selected. The distortion is still present, but the highlighting is visually apparent.

The design for image browsers should be governed by the users' tasks, which can be classified as follows (Plaisant et al., 1995):

- *Image generation.* Paint or construct a large image or diagram.
- *Open-ended exploration.* Browse to gain an understanding of the map or image.
- *Diagnostics.* Scan for flaws in an entire circuit diagram, medical image, or newspaper layout.
- *Navigation.* Have knowledge of the overview, but need to pursue details along a highway or vein.
- *Monitoring.* Watch the overview and, when a problem occurs, zoom in on details.

Within these high-level tasks, users carry out many low-level actions, such as moving the focus (jumping from city to city on a map), comparing (viewing two harbors at the same time to compare their facilities or viewing matching regions in x-ray images of left and right lungs), traversing (following a vein to look for blockages), or marking locations to return to them at a later time.

12.4 Animation

Small animations now pepper many applications to create an attractive look and feel. Perceiving and interpreting motion is a fundamental element of human perception. Our eyes are attracted to moving objects in the real world, and items moving in the same direction are interpreted as a group. Done well, animation is compelling. Done poorly, it can distract, irritate, and and waste time (see Box 12.2).

Smooth animations are now expected to keep users oriented during screen layout reorganizations such as zooming and panning on maps and opening and closing of windows. Individual screen elements use animation to guide users; for example, a flashing down arrow encourages tablet users to scroll down, and a left-to-right animation illustrates how a swiping movement is needed to

BOX 12.2

Possible roles for animation.

- Keeping user oriented during transition
- Indicating an affordance, inviting interaction
- Entertaining
- Indicating background activity (e.g., progress bar)
- Storytelling
- Alerting
- Providing a virtual tour (e.g., for architectural designs)
- Explaining a process
- Conveying uncertainty and randomness

unlock a phone. Showing changes over time is a natural application and particularly effective when coupled with voice narration (Rosling, 2015).

Application software used by Gapminder (2015) illustrates major global development trends using statistical data. That statistical data are visualized with an excellent use of animation (Fig. 12.9). See more examples of data visualization in Chapter 16.

Complex animations benefit from being decomposed in multiple clearly separated steps. For example, to facilitate the comparison of two lists of items, animation may be used to first move the identical terms to the center of the screen, then (after a short pause) move the unique terms further to the right or left, and after another pause bring similar items closer together (Plaisant et al., 2015). If instead all the objects had moved at the same time, it would be difficult to follow them and understand the meaning of the final layout. User controls may be needed to pause, skip, or replay the animation. The timing of the animation needs to be carefully crafted and tested. For example, an alert box sliding slowing into view can provide an effective, calm notification.

Still, animation can be hard to design and is not always helpful. A website may fail if the flashy animations take too long to load or users become frustrated by gratuitous animations that distract or slow down interaction. *Parallax scrolling* (which involves the background moving at a slower rate to the foreground, creating a 3-D effect as you scroll down the page) is the latest fashion (Parallax, 2015) but can also distract when overdone and often fails on small displays. However, some web designers use parallax scrolling effectively. Recent awards for parallax scrolling websites appear at http://www.awwwards.com/websites/parallax/.

Doubts remain about the usefulness of animation to explain complex processes, even if creating the animation itself may be beneficial to students and facilitate learning (Tversky et al., 2002). Zooming as a general navigation paradigm is key

FIGURE 12.9
Gapminder uses animation to compare how the income per person of five countries changed over time. The round markers leave a trail that can also be seen in the printed version. Here the user has selected India and is replaying the animation for that country (http://www.gapminder.org).

to map or document navigation, and tools such as Prezi (http://www.prezi.com) make very engaging presentations, but the vision of a zooming desktop has not been materialized (Bederson, 2011).

An interesting study of communicative functions (e.g., signal change of status, signal different context) versus animation types (e.g., change of place, change of size) yielded an animation use model of the appropriateness of the various combinations of animation types and communicative functions (Novick et al., 2011). While animation is often engaging, further studies are needed to sharpen our understanding of which situations benefit from animation and which ones do not.

12.5 Webpage Design

Webpage designers have dramatically improved their output in recent years. Numerous guidelines and internet sources contribute to maturity in this discipline. Visual layout has a strong impact on (human) performance and is a critical factor in webpage design. Newer studies illustrate more specific patterns of

BOX 12.3

The top 10 mistakes of web-based presentation of information (from Tullis, 2005).

Top 10 Mistakes

1. Burying information too deep in a website
2. Overloading pages with too much material
3. Providing awkward or confusing navigation
4. Putting information in unexpected places on the page
5. Not making links obvious and clear
6. Presenting information in bad tables
7. Making text so small that many users cannot read it
8. Using color combinations for text that many users cannot read
9. Using bad forms
10. Hiding (or not providing) features that could help users

performance with webpages that may reflect some differences between webpage and traditional GUI design.

Webpage designers can easily make mistakes that may not be immediately apparent but can result in distracting or misleading the user. Tullis (2005) has compiled a top 10 list of common design mistakes made in presenting information on the web (Box 12.3), based on the human-factors literature. Nielsen Norman Group (2011) also has a number of lists documenting top 10 mistakes of web design, which are insightful and certainly worth reviewing. Note that "bad search" is the number-one mistake item listed (also see Chapter 15 regarding information search).

Web-based designs are dramatically different because the broader consumer-oriented audience appreciates colorful graphics and many site designers use eye-catching photos. The race is on to create cool designs, enticing images, and attention-grabbing layouts. User preferences become crucial, especially if market researchers can demonstrate that site visitors stay longer and buy more products at visually compelling websites. The downside of the increased use of graphics is download times. Interesting design choices have to be made especially when how much data a website uses is an issue for laptop or mobile devices (Chetty et al., 2012; Mathur et al., 2015). Constraints of data usage and optimization have to play a role in the design.

Webpage designs have recently changed to accommodate tablets, smartphone displays, and other touchscreen interaction. The term *responsive design* refers to developing websites that provide an optimal viewing and interaction experience. There is a trend to push much more content onto each webpage.

That content is designed for more touchscreen scrolling and with larger buttons, graphics, or photos for selection tasks. People browsing the internet using a more traditional desktop with mouse input have had to adjust to this new design paradigm. There are not yet many studies on this transition, but it is certain that in this case the pervasive touchscreen technology available has driven webpage design decisions to work with these devices.

Also technology-related, webpage designs provide more animation (e.g., animated GIFs, animation, and video links). Icon links to social media sites for ease in sharing and commenting are a webpage design requirement that has become the de facto standard. Designers need to be sensitive to accessibility issues and potential distraction of constant webpage animations. Constantly running advertisements with repeated animations or eye-catching auto-play video windows (often using unnecessary data) prompt the user to click for sound or additional information about the topic. This not only can be an accessibility challenge but also is distracting to all users. One may find it annoying to read news on some news sites while a video starts automatically when the page is loaded—in this case, the user will have to stop the video in order to concentrate on the written content below the video. The user wants to limit the data, but the design prevents it.

More recently, seemingly endless scrolling webpage design has become the trend (which contradicts some of the ideas proposed by Tullis (Box 12.3) only a decade earlier). Best practices in recent years had suggested much less scrolling, particularly on home pages—that users would have preferred quicker point-and-click, menu-style navigation to the desired information. With the advent of mobile devices, people are given large buttons and the ability to scroll on one very, very long page to navigate to the desired information. News and sports websites jumped on this approach as their user studies proved more people were viewing their content on mobile devices (smartphones and tablets). Those without touchscreens (desktops or laptops not enabled for touch) are forced to scroll long pages using the scroll bar or spinning the mouse thumbwheel. When Windows 8 was first introduced, there was shock by some that the Start button was essentially replaced by a large, tiled "metro" interface (and later returned in a subsequent version of Windows). It almost seems like a battle of dialog styles was held, where the direct manipulation proponents beat the menu-driven, mouse-clicking navigation team as the prevailing dialog style.

Another recent trend is toward dynamic and real-time updates to webpages. There is a convergence of web design toward other application design, while other apps (e.g., mobile apps) incorporate web views that may just load HTML-based webpages. Actually, the evolution of HTML both enables and limits web design in terms of widgets, layouts, and so on.

Universal usability again is a key factor in webpage design. Web usability for low-vision users can be enhanced by larger display fonts, brightness settings, and audio assistance.

Web content presentation issues can be broken up into site-level issues, page-level issues, and "special" types of information (Tullis, 2004). Site-level issues are, obviously, apparent throughout the entire website rather than only on individual pages; they include the depth versus breadth of the site, the use of frames, and the presentation of navigation options. Page-level issues are observed at the individual page level; they include components of pages, such as tables, graphs, forms, and controls, as well as issues such as page layout and presentation of links. "Special" web content can include site maps, search functions, user assistance, and feedback.

Many users have indicated preferences for large pages that have a columnar organization, limited animated graphical ads, average link text kept to two to three words, sans serif fonts, and colors used to highlight text and headings. Usability study results may indicate designers accommodating high personal preference—for example, comprehensibility, predictability, familiarity, visual appeal, and relevant content.

Numerous guidelines for web designers are available on the web and can be incorporated into the design process to ensure consistency and adherence to emerging standards. Examples include but are not limited to:

- *The Java Look and Feel Design Guidelines* (Oracle, 2015)
- *Research-Based Web Design & Usability Guidelines* (Usability, 2015)
- *The World Wide Web Consortium's Web Accessibility Initiative* (Web Accessibility Initiative, 2015)
- *The Web Style Guide* (Lynch and Horton, 2008)

There are numerous websites that address web design, some of which were created as companions to relevant books:

- *Web 2.0 How-To Design Guide* (Hunt, 2015)
- *Web Bloopers* (Johnson, 2003—updated 2008 online)
- *Building Scalable Web Sites: Building, Scaling, and Optimizing the Next Generation of Web Applications* (Henderson, 2006)

12.6 Color

Color displays are attractive to users and can often improve task performance, but the danger of misuse is high. Color can do all of the following:

- Soothe or strike the eye
- Add accents to an uninteresting display

- Facilitate subtle discriminations in complex displays
- Emphasize the logical organization of information
- Draw attention to warnings
- Evoke strong emotional reactions of joy, excitement, fear, or anger

The principles developed by graphic artists for using color in books, magazines, highway signs, and other print media have been adapted for user interfaces (Marcus, 1992; MacDonald, 1999; Stone, 2003). Interactive-systems designers have learned how to create effective computer displays and to avoid pitfalls (Brewer et al., 2003; Galitz, 2007).

There is no doubt that color makes video games more attractive to users, conveys more information on power-plant or process-control diagrams, and is necessary for realistic images of people, scenery, or 3-dimensional objects (Foley et al., 2002; Weinman, 2002). These applications require color. However, greater controversy exists about the benefits of color for alphanumeric displays, spreadsheets, graphs, and user-interface components.

No simple set of rules governs use of color, but these guidelines are a starting point for designers:

- *Use color conservatively.* Many programmers and novice designers are eager to use color to brighten up their displays, but the results are often counterproductive. One home-information system displayed the seven letters of its name in a large font, each in a different color. At a distance, the display appeared inviting and eye-catching; up close, however, it was difficult to read.

  When the colors do not show meaningful relationships, they may mislead users into searching for relationships that do not exist. Using a different color for each of 12 unrelated items in a menu produces an overwhelming effect. However, using four colors (such as red, blue, green, and yellow) for the 12 items will mislead users into thinking that all the similarly colored items are related. An appropriate strategy might be to show all the menu items in one color, the title in a second color, the instructions in a third color, and error messages in a fourth color, but even this strategy can be overwhelming if the colors are too visually striking. A safe approach is to always use black letters on a white background, with italics or bold for emphasis, and to reserve color for special highlighting.
- *Limit the number of colors.* Many design guides suggest limiting the number of colors in a single display to four, with a limit of seven colors in the entire sequence of displays. Experienced users may be able to benefit from a larger number of color codes, but for novice users, too many color codes can cause confusion.
- *Recognize the power of color as a coding technique.* Color speeds recognition for many tasks. For example, in an accounting application, if data lines with accounts overdue more than 30 days are coded in red, they will be readily visible among the non-overdue accounts coded in green. In an air-traffic-control

system, high-flying planes might be coded differently from low-flying planes to facilitate recognition. In some programming languages, keywords are color-coded differently from variables.

- *Ensure that color coding supports the task.* Be aware that using color as a coding technique can inhibit performance of tasks that go against the grain of the coding scheme. If, in the above accounting application with color coding by days overdue, the task is to locate accounts with balances of more than $55, the existing coding scheme may inhibit performance on the second task. Designers should attempt to make a close linkage between the users' tasks and the color coding and offer users control where possible.
- *Have color coding appear with minimal user effort.* In general, users should not have to activate color coding each time they perform a task; rather, the color coding should appear automatically. For example, when the users start the task of checking for overdue accounts, the color coding should be set automatically; when they click on the task of locating accounts with balances of more than $55, the new color-coding scheme should also take effect automatically.
- *Place color coding under user control.* When appropriate, the users should be able to turn off color coding. For example, if a spelling checker highlights possibly misspelled words in red, the user should be able to accept or change the spelling and to turn off the coding. The presence of the highly visible red coding is a distraction from reading the text for comprehension.
- *Design for monochrome first.* The primary goal of a display designer should be to lay out the contents in a logical pattern. Related fields can be shown by contiguity or by similar structural patterns; for example, successive employee records may have the same indentation pattern. Related fields can also be grouped by a box drawn around the group. Unrelated fields can be kept separate by inserting blank space (at least one blank line vertically or three blank characters horizontally).
- *Consider the needs of color-deficient users.* One important aspect to consider is readability of colors by users with color-vision impairments (either red/green confusion, the most common case, or total color blindness). Color impairment is a very common condition. Approximately 8% of males and less than 1% of females in North America and Europe have some permanent color deficiency in their vision. Many others have temporary deficiencies due to illness or medications. They may, for example, confuse some shades of orange or red with green or not see a red dot on a black background. Designers can easily address this problem by limiting the use of color, using double encoding when appropriate (that is, using symbols that vary in both shape and color or location and color), providing alternative color palettes to choose from, or allowing users to customize the colors themselves. For example, the SmartMoney® Map of the Market provides two choices of color schemes: red/green and blue/yellow. Various tools, such as Vischeck, are available to

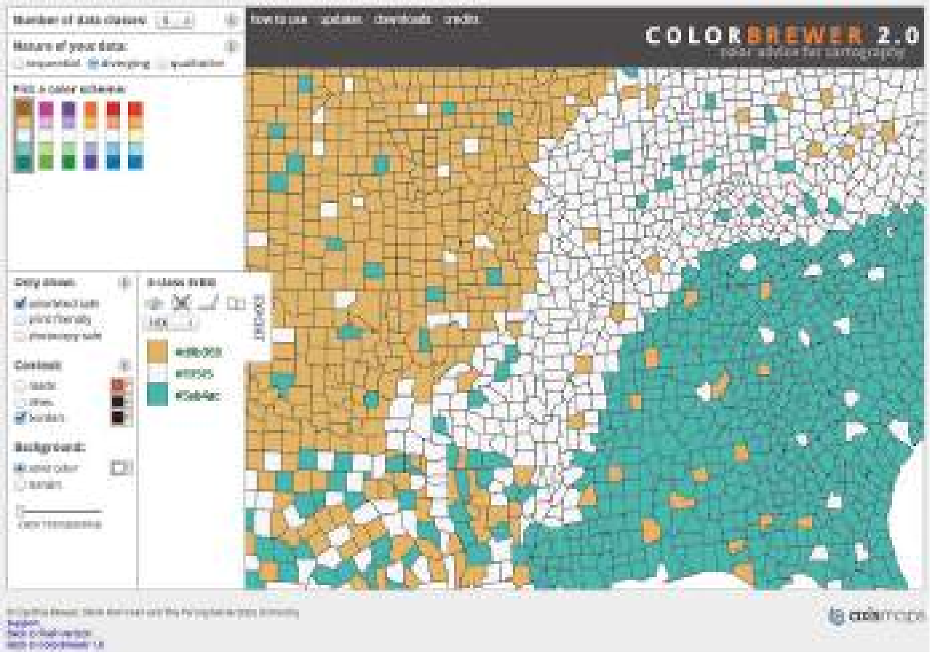


FIGURE 12.10

ColorBrewer helps designers select color schemes for maps and other graphics (Brewer et al., 2015). Controls also allow users to narrow down the choices to colors that are visible to users with color impairments.

both simulate color-vision impairments and optimize graphics for some of the various forms of color impairment that exist. Black on white or white on black will work well for most users. ColorBrewer (see Fig. 12.10), an online tool designed to help people select good color schemes for maps and other graphics, offers guidelines on color schemes that work for those with color-vision impairments (Brewer et al., 2015; Brewer, 2015).

- *Use color to help in formatting.* In densely packed displays where space is at a premium, similar colors can be used to group related items. For example, in a police dispatcher's tabular display of assignments, the police cars on emergency calls might be coded in red, and the police cars on routine calls might be coded in green. Then, when a new emergency arises, it will be relatively easy to identify the cars on routine calls and to assign one to the emergency. Dissimilar colors can be used to distinguish physically close but logically distinct fields. In a block-structured programming language, designers could show the nesting levels by coding the statements in a progression of colors—for example, dark green, light green, yellow, light orange, dark orange, red, and so on.

- *Be consistent in color coding.* Use the same color-coding rules throughout the system. If some error messages are displayed in red, make sure that every error message appears in red; a change to yellow may be interpreted as a change in the importance of the message. If the various system designers use colors differently, users will hesitate as they attempt to assign meaning to the color changes.
- *Be alert to common or cultural expectations about color codes.* Designers need to speak to users to determine what color codes are applied in the task domain. From automobile-driving experience, red is commonly considered to indicate stop or danger, yellow is a warning, and green is all clear or go. In investment circles, red is a financial loss and black is a gain. For chemical engineers, red is hot and blue is cold. For mapmakers, blue means water, green means forests, and yellow means deserts. These differing conventions can cause problems for designers. Designers might consider using red to signal that an engine is warmed up and ready, but users might interpret the red coding as an indication of danger. A red light is often used to indicate that power is on for electrical equipment, but some users are made anxious by this decision since red has a strong association with danger or stopping. When appropriate, indicate the color-code interpretations on the display or in a help panel.
- *Be alert to problems with color pairings.* If saturated (pure) red and blue appear on a display at the same time, it may be difficult for users to absorb the information. Red and blue are on the opposite ends of the visible spectrum, and the muscles surrounding the human eye will be strained by attempts to produce a sharp focus for both colors simultaneously: The blue will appear to recede and the red will appear to come forward. Blue text on a red background would present an especially difficult challenge for users to read. Other combinations (yellow on purple, magenta on green) will similarly appear garish and be difficult to read. Too little contrast also is a problem: Imagine yellow letters on a white background or brown letters on a black background.
- *Use color changes to indicate status changes.* For automobile speedometers with digital readouts and a wireless receiver of the driving speed limits, it might be helpful to change from green numbers below the speed limit to red above the speed limit to act as a warning. Similarly, in an oil refinery, pressure indicators might change color as the value goes above or below acceptable limits. In this way, color acts as an attention-getting method. This technique is potentially valuable when there are hundreds of values displayed continuously.
- *Use color in graphic displays for greater information density.* In graphs with multiple plots, color can be helpful in showing which line segments form the full graph. The usual strategies for differentiating lines in black-on-white graphs—such as dotted lines, thicker lines, and dashed lines—are not as effective as using separate colors for each line. Architectural plans benefit from color coding of electrical, telephone, hot-water, cold-water, and natural-gas lines. Similarly, maps can have greater information density when color coding is used.

Color displays are becoming nearly universal, even in mobile devices, and designers usually make heavy use of color in interface designs. There are undoubtedly benefits in terms of increased user satisfaction and often enhanced performance, but there are also real dangers in misusing color. Care should be taken to make appropriate design choices and to conduct thorough evaluations (Box 12.4).

BOX 12.4

Guidelines that highlight the complex potential benefits and dangers of using color coding.

Guidelines for using color

- Use color conservatively: Limit the number and amount of colors.
- Recognize the power of color to speed or slow tasks.
- Ensure that color coding supports the task.
- Make color coding appear with minimal user effort.
- Keep color coding under user control.
- Design for monochrome first.
- Consider the needs of color-deficient users.
- Use color to help in formatting.
- Be consistent in color coding.
- Be alert to common expectations about color codes.
- Be alert to problems with color pairings.
- Use color changes to indicate status changes.
- Use color in graphic displays for greater information density.

Benefits of using color

- Various colors are soothing or striking to the eye.
- Color can improve an uninteresting display.
- Color facilitates subtle discriminations in complex displays.
- A color code can emphasize the logical organization of information.
- Certain colors can draw attention to warnings.
- Color coding can evoke more emotional reactions of joy, excitement, fear, or anger.

Dangers of using color

- Color pairings may cause problems.
- Color fidelity may degrade on other hardware.
- Printing or conversion to other media may be a problem.

12.7 Nonanthropomorphic Design

There is a great temptation to have computers "talk" as though they are people. Children accept human-like references and qualities for almost any object, from Humpty Dumpty to SpongeBob™. Some adults still reserve *anthropomorphic* references for objects of special attraction, such as cars, ships, computers, and even cellphones.

"People treat computers as if they were people" was a quote from the late Clifford Nass at Stanford who had performed significant research on human-computer interaction, people multitasking, improved automobile design, and more. One book of his insights deals with computers and relationships (Nass and Yen, 2010).

The words and graphics in user interfaces can make important differences in people's perceptions, emotional reactions, and motivations. Attributions of intelligence, autonomy, free will, or knowledge to computers are appealing to some people, but to others such characterizations may be seen as deceptive, confusing, and misleading. The suggestion that computers can think, know, or understand may give users an erroneous model of how computers work and what the machines' capacities are. Ultimately, the deception becomes apparent, and users may feel poorly treated.

A second reason for using nonanthropomorphic phrasing is to clarify the differences between people and computers. Relationships with people are different from relationships with computers. Users operate and control computers, but they respect the unique identities and autonomy of individuals. Furthermore, users and designers must accept responsibility for misuse of computers rather than blaming the machines for errors.

A third motivation is that, although an anthropomorphic interface may be attractive to some people, it can be distracting or produce anxiety for others. Some people express anxiety about using computers and believe that computers "make you feel dumb." Presenting the computer through the specific functions it offers may be a stronger stimulus to user acceptance than promoting the fantasy that the computer is a friend, parent, or partner. As users become engaged, the computer becomes transparent, and they can concentrate on their writing, problem solving, or exploration. At the end, they have the experience of accomplishment and mastery rather than the feeling that some magical machine has done their job for them. Anthropomorphic interfaces may distract users from their tasks and waste their time as they consider how to please or be socially appropriate to the onscreen characters.

Experts in consumer product interface design often speak of *making technology invisible* (Bergman, 2000). Designers have fielded interfaces on such items as mall kiosks, postage dispensers, and interactive voice response (IVR) systems that anthropomorphize, giving an impression to the novice user that the computer systems are doing some intelligent reasoning while adding stress on and disempowering the user. IVR systems are notorious for anthropomorphizing interfaces. Here are some current examples: One airline reservation system says, "OK, I can help with that…" after you request to initiate a domestic reservation; an automated banking system says, "Please hold while I check your account balance…"; a mail-order pharmacy says, "Would you like me to send your prescription to the address on record?" Think about experiences users may have had that are humorous interactions with Siri™. The speech-recognition technology for many of these systems is improving (see Chapter 9) but still may cause user frustration.

Individual differences in the desire for internal locus of control are important, but there may be an overall advantage to clearly distinguishing human abilities from computer powers for most tasks and users (Shneiderman, 1995). On the other hand, there are persistent advocates of creating anthropomorphic interfaces, often called virtual humans, lifelike autonomous agents, "chatbots," or embodied conversational agents (Cassell et al., 2000; Gratch et al., 2002; D'Mello et al., 2007).

Advocates of anthropomorphic interfaces assume that human-human communication is an appropriate model for human operation of computers. It may be a useful starting point, but some designers pursue the human-imitation approach long after it becomes counterproductive. Mature technology has managed to overcome the *obstacle of animism* (animism refers to non-human things like animals, plants, and inanimate objects possessing a spiritual essence), which has been a trap for technologists for centuries (Mumford, 1934); a visit to the Museum of Automata in York, England, reveals the ancient sources and persistent fantasies of animated dolls and robotic toys.

Historical precedents of failed anthropomorphic bank tellers, such as Tillie the Teller, Harvey Wallbanker, and BOB (Bank of Baltimore), and of abandoned talking automobiles and soda machines do not seem to register with some designers. The bigger-than-life-sized Postal Buddy was supposed to be cute and friendly while providing several useful automated services, but users rejected this pseudo–postal clerk after the project had incurred costs of more than $1 billion. Advocates of anthropomorphic interfaces suggest that they may be most useful as teachers, salespeople, therapists, or entertainment figures. Animated characters that range from cartoon-like to realistic have been embedded in many interfaces, but evidence is growing that they increase anxiety and reduce performance, especially for users with an external locus of control.

One specific design controversy is over the use of first-person pronouns in an interface. Advocates believe it makes the interaction friendly, but such interfaces may be counterproductive because they can deceive, mislead, and confuse users. It may seem cute on the first encounter to be greeted by "I am SOPHIE, the sophisticated teacher, and I will teach you to spell correctly." By the second session, however, this approach strikes many people as silly; by the third session, it can be an annoying distraction from the task. The alternative for the interface designer is to focus on the user and to use third-person-singular pronouns or to avoid pronouns altogether. Improved messages may also suggest a higher level of user control. For example:

Poor: `I will begin the lesson when you press RETURN.`
Better: `You can begin the lesson by pressing RETURN.`
Better yet: `To begin the lesson, press RETURN.`

The *you* form seems preferable for introductions; however, once the session is under way, reducing the number of pronouns and words avoids distractions from the task.

The issue of pronoun usage reappears in the design of interactive voice-response telephone interfaces, especially if speech recognition is employed. Advocates argue that greetings from a rental-car reservation service, for example, might be more appealing if they simulate a human operator: “Welcome to Thrifty Car Rentals. I’m Emily, let me help you reserve your car. In what city will you need a car?” While most users won’t care about the phrasing, opponents claim that this deception does annoy and worry some users and that the expedient solution of deleting the chatty second sentence produces higher customer satisfaction.

Some designers of children’s educational software believe that it is appropriate and acceptable to have a fantasy character, such as a teddy bear or busy beaver, serve as a guide through a lesson. A cartoon character can be drawn on the screen and possibly animated, adding visual appeal, speaking to users in an encouraging style, and pointing to relevant items on the display. Successful educational software packages such as Leap Frog® and some empirical research (Mayer, 2009) provide support for this position. Child-computer interaction is now a research and development field unto itself—see the text by Hourcade (2015) and by Druin (2009).

Unfortunately, cartoon characters were not successful in BOB, a heavily promoted but short-lived home-computing product from Microsoft. Users could choose from a variety of onscreen characters that spoke in cartoon bubbles with phrases such as: “What a team we are,” “Good job so far, Ben,” and “What shall we do next, Ben?” This style might be acceptable in children’s games and educational software, but it is probably not acceptable for adults in the workplace. Interfaces should neither compliment nor condemn users, just provide comprehensible feedback so users can move forward in achieving their goals. However, anthropomorphic characters will not necessarily succeed here, either. Microsoft’s ill-fated Clippy character (a lively paper-clip cartoon character) was designed to provide helpful suggestions for users. It amused some but annoyed many, and it was soon demoted to an optional extra. Defenders of anthropomorphic interfaces found many reasons to explain Clippit’s rejection (primarily its disruptive interference with users). Others believe that successful anthropomorphic interfaces require socially appropriate emotional expressions as well as well-timed head movements, nods, blinks, and eye contact.

An alternative instructional design approach that seems acceptable to many users is to present the human author of a lesson or software package. Through an audio or video clip, the author can speak to users, much as television news announcers speak to viewers. Instead of making the computer into a person, designers can show identifiable and appropriate human guides. For example, the Secretary-General might record a video welcome for visitors to a website about the United Nations, or Bill Gates might provide greetings for new users of Windows.

Once past these introductions, several styles are possible. One is a continuation of the guided-tour metaphor, in which the respected personality introduces segments but allows users to control the pace, to repeat segments, and to decide when they are ready to move on. A variant of this approach creates an interview-like experience in which users read from a set of three prompts and issue spoken commands to get pre-recorded video segments by noted figures such as former Senator John Glenn. This approach works for museum tours, tutorials on software, and certain educational lectures.

Another strategy is to support user control and enhance the user experience by showing an overview of the modules from which users can choose. Users decide how much time to spend visiting parts of museums, browsing a timeline with details of events, or jumping between articles in a hyperlinked encyclopedia. These overviews give users a sense of the magnitude of information available and allow them to see their progress in covering the topics. Overviews also support users' needs for closure, give them the satisfaction of completely touring the contents, and offer a comprehensible environment with predictable actions that foster a comforting sense of control. Furthermore, they support the need for replicability of actions (to revisit an appealing or confusing module or to show it to a colleague) and reversibility (to back up or return to a known landmark). While in games users may enjoy the challenge of confusion, hidden controls, and unpredictability, this is not the case in most applications; rather, designers must strive to make their products comprehensible and predictable. A summary of nonanthropomorphic guidelines appears in Box 12.5.

BOX 12.5

Guidelines for avoiding anthropomorphism and building appealing interfaces.

Nonanthropomorphic guidelines

- Be cautious in presenting computers as people, either with synthesized or cartoon characters.
- Design comprehensible, predictable, and user-controlled interfaces.
- Use appropriate humans for audio or video introductions or guides.
- Use cartoon characters in games or children's software, but avoid them elsewhere.
- Provide user-centered overviews for orientation and closure.
- Do not use pronoun *I* when the computer responds to human actions.
- Use *you* to guide users, or just state facts.

12.8 Error Messages

Error messages are a key part of an overall interface design strategy of guidance for the user. The strategy should ensure integrated, coordinated error messages that are consistent across one or multiple applications.

Design disasters have appeared in systems and websites where error messages written by multiple authors read quite obviously as though they were written by multiple authors. There are several "hall of shame" error-message websites where communities of users and developers share bizarre and misleading error-message experiences. Some are critical and humorous, while others are informative, providing lessons learned and suggestions for improvement; an example is the error-message discussion on the Microsoft Developers Network (2015).

Solutions for avoiding error-message design disasters include discussing help and error handling in a style guide for all designers to review and follow and ensuring that error messages are designed into a computing system or website rather than being added as a final step or afterthought.

One problem that is sometimes seen is when error messages do not clearly correspond with the help provided, illustrating an obvious information gap in transitioning from the error message to assisting the user in performing the corrective action. With respect to international user interfaces, designers can run into trouble when having third-party language experts translate error messages, help text, prompts, and other support features. Experienced designers isolate the error messages and help text information into separate files (not hard-coded) for ease of translation during the development phase and later maintenance updates. This also permits on-site, local language selection when a system is installed in a country other than the one where the software was originally created.

Normal prompts, advisory messages, and system responses to user actions may influence user perceptions, but the phrasing of error messages and diagnostic warnings is critical. Since errors occur because of lack of knowledge, incorrect understanding, or inadvertent slips, users are likely to be confused, to feel inadequate, and to be anxious when they encounter these messages. Error messages with an imperious tone that condemn users can heighten anxiety, making it more difficult to correct the problems and increasing the chances of further errors. Unhelpful help content is another source of stress.

These concerns are especially important with respect to novices, whose lack of knowledge and confidence amplifies the stress that can lead to a frustrating sequence of failures. The discouraging effects of a bad experience in using a computer are not easily overcome by a few good experiences. In some cases, interfaces are remembered more for what happens when things go wrong than for when things go right. Although these concerns apply most forcefully to novice computer users, experienced users also suffer. Experts in one interface or part of an interface are still novices in many situations.

Improving error messages is one of the easiest and most effective ways to improve an existing interface. If the software can capture the frequency of errors, designers can focus on optimizing the most important messages. Error-frequency distributions also enable interface designers and maintainers to revise error-handling procedures, to improve documentation and tutorials, to alter online help, or even to change the permissible actions. The complete set of messages should be reviewed by peers and managers, tested empirically, and included in user documentation.

Specificity, constructive guidance, a positive tone, a user-centered style, and an appropriate physical format are recommended as the bases for preparing error messages (Box 12.6). These guidelines are especially important when

BOX 12.6

Error-message guidelines for the end product and for the development process. These guidelines are derived from practical experience and empirical data.

End product

- Be as specific and precise as possible. Determine necessary, relevant error messages.
- Be constructive. Indicate what the user needs to do.
- Use a positive tone. Avoid condemnation. Be courteous.
- Choose user-centered phrasing. State the problem, cause, and solution.
- Consider multiple levels of messages. State brief, sufficient information to assist with the corrective action.
- Maintain consistent grammatical forms, terminology, and abbreviations.
- Maintain consistent visual format and placement.

Development process

- Increase attention to message design.
- Establish quality control.
- Develop and enforce guidelines.
- Carry out usability tests.
- Consider conducting “error handling” reviews.
- Record the frequency of occurrence for each message.

the users are novices, but they can benefit experts as well. The phrasing and contents of error messages can significantly affect user performance and satisfaction, thereby advancing the overall user experience.

12.8.1 Specificity

Messages that are too general make it difficult for the novice to determine what has gone wrong. Simple and condemning messages are frustrating because they provide neither enough information about what has gone wrong nor the knowledge to set things right. Therefore, the right amount of specificity is important. Here are some examples:

Poor: `SYNTAX ERROR`
Better: `Unmatched left parenthesis`

Poor: `INVALID DATA`
Better: `Days range from 1 to 31`

Poor: `BAD FILE NAME`
Better: `The file C:\demo\data.txt was not found`

Poor: `???`
Better: `Touch icon twice to start app`

One interface for hotel check-in required the desk clerk to enter a 40- to 45-character string containing the name, room number, credit-card information, and so on. If the clerk made a data-entry error, the only message was `INVALID INPUT. YOU MUST RETYPE THE ENTIRE RECORD`. This led to frustration for users and delays for irritated guests. Interactive systems should be designed to minimize input errors by use of proper form fill-in strategies (Chapter 8); when an error occurs, the users should have to repair only the incorrect part.

For example, one good design that is apparent on some websites prompts a user to complete a form fill-in task and then submit the form. If there are any errors, the form is again presented to the user highlighting what information needs to be correctly entered. The user makes the corrections and then resubmits the form (without being forced to complete the entire form again). Using a combination of form fill-in and menu interaction styles (Chapter 8) can also reduce errors (e.g., selecting a country from a list rather than having the user type the country name into the form).

12.8.2 Constructive guidance and positive tone

Rather than condemning users for what they have done wrong, messages should, where possible, indicate what users need to do to set things right:

Poor: `Run-Time error '-2147469 (800405)': Method 'Private Profile String' of object 'System' failed.`
Better: `Virtual memory space consumed. Close some programs and retry.`

Poor: `Network connection refused.`
Better: `Password was not recognized. Please retype.`

Poor: `Invalid date.`
Better: `Arrival date must come after departure date.`

Unnecessarily hostile messages using violent terminology can disturb nontechnical users. An interactive legal-citation–searching system uses this message: `FATAL ERROR, RUN ABORTED`. Similarly, an early operating system threatened many users with `CATASTROPHIC ERROR; LOGGED WITH OPERATOR`. There is no excuse for these hostile messages; they can easily be rewritten to provide more information about what has happened and what must be done to set things right. Where possible, be constructive and positive. Such negative words as `ILLEGAL`, `ERROR`, `WRONG`, `INVALID`, or `BAD` should be eliminated or used infrequently.

It may be difficult for the software writer to create a program that accurately determines the user's intention, so the advice to "be constructive" is often difficult to apply. Some designers argue for automatic error correction, but the disadvantage is that users may fail to learn proper syntax and may become dependent on alterations that the system makes. Envision the elementary-school language teacher trying to train schoolchildren in spelling when the papers students hand in are spell-checked, grammar-checked, and even autocorrected as they are typed; the students will have no incentive to self-correct, and the teacher will not be able to see what mistakes they are making. Another approach is to inform users of the possible alternatives and to let them decide. A preferred strategy is, where possible, to prevent errors from being made (Section 3.3.5).

12.8.3 Appropriate physical format

Most users prefer and find it easier to read mixed uppercase and lowercase messages. Uppercase-only messages should be reserved for brief, serious warnings. Messages that begin with a lengthy and mysterious code number serve only to remind the user that the designers were insensitive to the users' real needs. If code numbers are needed at all, they might be enclosed in parentheses at the end of a message or as a "provide more details" function.

There is disagreement about the optimal placement of messages in a display. One school of thought argues that the messages should be placed on the display near where the problem has arisen. A second opinion is that the messages clutter the display and should be placed in a consistent position on the bottom of the display. The third approach is to display a dialog box near to, but not obscuring, the relevant problem. It seems all too common to have an error message pop up in the center of the page, halting all activity and forcing the user to move the dialog box or to click a confirmation (e.g., "OK") to continue.

Some applications sound a tone when an error has occurred. This alarm can be useful if the operator might otherwise miss the error, but it can be embarrassing if other people are nearby, and it is potentially annoying even if the operator is alone.

Designers must walk a narrow path between calling attention to a problem and avoiding embarrassment to users. Considering the wide range of experience and temperament in users, maybe the best solution is to offer users control over the alternatives—this approach coordinates well with the user-centered principle.

Improved messages will be of the greatest benefit to novice users, but regular users and experienced professionals will also benefit. As examples of excellence proliferate, complex, obscure, and harsh interfaces will seem increasingly out of place. The crude environments of the past will be replaced gradually by interfaces designed with the users in mind. Resistance to such a transition should not be allowed to impede progress toward the goal of serving the growing user community.

Practitioner's Summary

Pay careful attention to display design, and develop a local set of guidelines for all designers. Use spacing, indentation, columnar formats, and field labels to organize the display for users. Denser displays, but fewer of them, may be advantageous. Organizations can benefit from careful study of display-design guidelines documents and from the creation of their own sets of guidelines tailored to local needs (Section 3.2). These documents should also include lists of local terminology and abbreviations. Consistency and thorough testing are critical.

Current web technologies and new web design guidelines provide new tools and emerging methods for user-generated content to be easily and rapidly inserted into websites. Universal usability is also addressed by current web design guidelines. Good window design methods can enhance the user experience.

Color can improve some displays and can lead to more rapid task performance with higher satisfaction. However, improper use of color can mislead, distract, and slow users.

When giving instructions, focus on the user and the user's tasks. In most applications, avoid anthropomorphic phrasing. Use the *you* form to guide the novice user. Avoid judging the user. Simple statements of status are more succinct and usually are more effective.

The wording of system messages may have an effect on users' performance and attitudes, especially for novices, whose anxiety and lack of knowledge put them at a disadvantage. Designers might make improvements by merely using more specific diagnostic messages, offering constructive guidance rather than focusing on failures, employing user-centered phrasing, choosing a suitable physical format, and avoiding vague terminology or numeric codes.

Researcher's Agenda

Increase designer knowledge utilizing World Wide Web tools and methods as they apply to website design to advance the user experience, facilitate user-generated content, and promote universal usability.

Basic understanding and cognitive models of visual perception of displays as display technology improves continue to be a need. Do users follow a scanning pattern from the top left? Would research using eye-tracking systems clarify reading and focus of attention patterns? Do users whose native language reads from right to left or users from different cultures scan displays differently? Does use of white space around or boxing of items facilitate comprehension and speed interpretation? When is a single dense display preferable to two sparse displays? How does color coding reorganize or disrupt the pattern of scanning? What about research in recent trendy webpage designs like those using parallax scrolling: easy (and fun) to use or confusing for a novice user?

Experimental testing could refine the error-message guidelines proposed here and could identify the sources of user anxiety and confusion. Message placement, highlighting techniques, and multiple-level message strategies are candidates for exploration. Improved analysis of sequences of user actions to provide more effective messages automatically would be useful. Since anthropomorphic designs are rarely successful, believers in human-like agents should conduct empirical studies to test their efficacy.

WORLD WIDE WEB RESOURCES

www.pearsonglobaleditions.com/shneiderman

Guidelines for display design, webpages, and window management exists on the World Wide Web with some empirical results, but the most informative and enjoyable experience is simply browsing through the lively and colorful websites. Web and other interface styles and fashions come and go quickly.

Discussion Questions

1. Effective display designs must provide all the necessary data in the proper sequence to carry out the task. Identify a recent personal user experience where it either was very clear or very unclear about what the sequence of steps was necessary to complete a task. What made that experience memorable?
2. List a few reasons for using animation in display design.
3. Briefly explain the term “responsive design”.
4. Color displays are attractive to users and can often improve task performance, but the danger of misuse is high. List five guidelines for using color and give an example of each.
5. For the following three items, find the problems in each of the following error messages produced by a source code compiler. Briefly state your reasoning and suggest a better message.

   SYNTAX ERROR

   INCORRECTLY FORMED RHS OF EXPRESSION. BAILING OUT ON LINE 6.

   ILLEGAL OPERATION. PROCESS KILLED!!!
6. What problems do anthropomorphic interfaces pose for the users?

References

Bartram, Lyn, Ho, Albert, Dill, John, and Henigman, Frank, The continuous zoom: A constrained fisheye technique for viewing and navigating large information spaces, *Proceedings of the ACM Symposium on User Interface Software and Technology (UIST)*, ACM Press, New York (1995), 207–215.

Baudisch, P., Lee, B., and Hanna, L., Fishnet, a fisheye web browser with search term popouts: A comparative evaluation with overview and linear view, *Proceedings of the*

Working Conference on Advanced Visual Interfaces, ACM Press, New York (May 2004), 133–140.

Bederson, Ben, The promise of zoomable user interfaces, *Behaviour & Information Technology 30,* 6 (2011), 853–866.

Bergman, Eric, *Information Appliances and Beyond: Interaction Design for Consumer Products*, Morgan Kaufmann, San Francisco, CA (2000).

Brewer, Cynthia A., Hatchard, Geoffrey W., and Harrower, Mark A., ColorBrewer in print: A catalog of color schemes for maps, *Cartography and Geographic Information Science 30,* 1 (2003), 5–32.

Brewer, C., Harrower, M., Sheesley, B., Woodruff A., and Heyman, D. (2015). Available at http://colorbrewer2.org/.

Brewer, C., *Designing Better Maps*, 2nd Edition, ESRI Press, Redlands, CA (2015).

Cassell, Justine, Sullivan, Joseph, Prevost, Scott, and Churchill, Elizabeth, *Embodied Conversational Agents*, MIT Press, Cambridge, MA (2000).

Chetty, M., Banks, R., Bernheim Brush, A. J., Donner, J., and Grinter, R. E., You're capped: Understanding the effects of broadband caps on broadband use in the home, *Proceedings of the ACM Conference on Human Factors in Computing Systems*, ACM Press, New York (2012).

Cockburn, A., Karlson, A., and Bederson, B., A review of overview + detail, zooming, and focus + context interfaces, *ACM Computing Surveys 41*, 1, ACM Press, New York (2009), Article 1.

D'Mello, S. K., Picard, R., and Graesser, A. C., Toward an affect-sensitive AutoTutor, *IEEE Intelligent Systems 22,* 4 (July/August 2007), 53–61.

Druin, A., *Mobile Technology for Children: Designing for Interaction and Learning*, Morgan Kaufmann (2009).

Foley, James D., van Dam, Andries, Feiner, Steven K., and Hughes, John F., *Computer Graphics: Principles and Practice,* 2nd Edition in *C*, Addison-Wesley, Reading, MA (2002).

Galitz, Wilbert O., *The Essential Guide to User Interface Design: An Introduction to GUI Design Principles and Techniques,* 3rd Edition, John Wiley & Sons, New York (2007).

Gapminder (2015). Available at http://www.gapminder.org/about-gapminder/.

Gratch, J., Rickel, J., Andre, E., Badler, N., Cassell, J., and Petajan, E., Creating interactive virtual humans: Some assembly required, *IEEE Intelligent Systems 17,* 4 (2002), 54–63.

Henderson, Cal, *Building Scalable Web Sites: Building, Scaling, and Optimizing the Next Generation of Web Applications*, O'Reilly Media, Inc. (2006).

Hornbaek, K., Bederson, B. B., and Plaisant, C., Navigation patterns and usability of zoomable user interfaces with and without an overview, *ACM Transactions on Computer-Human Interaction 9*, 4 (December 2002), 362–389.

Hourcade, J. P., *Child Computer Interaction* (2015). Available at http://homepage.cs.uiowa.edu/~hourcade/book/index.php.

Hunt, Ben, *Web 2.0 How-To Design Guide* (2015). Available at http://webdesignfromscratch.com/web-design/web-2-0-design-style-guide/.

Johnson, Jeff, *Web Bloopers*, Morgan Kaufmann, San Francisco, CA (2003). Updates available at http://www.web-bloopers.com/ (2008).

Johnson, Jeff, *Designing with the Mind in Mind: Simple Guide to Understanding User Interface Design Rules:* 2nd Edition, Morgan Kaufmann, San Francisco, CA (2014).

Jones, M. C., Floyd, I. R., Rathi, D., and Twidale, M. B., Web mash-ups and patchwork prototyping: User-driven technological innovation with Web 2.0 and open source software, *Proceedings of the 40th Annual Hawaii International Conference on System Sciences (HICSS '07)*, IEEE Press, Los Alamitos, CA (2007).

Law, E. L. C., The measurability and predictability of user experience, *Proceedings of the 3rd ACM SIGCHI Symposium on Engineering Interactive Computing Systems*, ACM Press, New York (2011).

Lynch, Patrick J., and Horton, Sarah, *Web Style Guide: Basic Design Principles for Creating Web Sites*, 3rd Edition, Yale University Press, New Haven, CT (2008). Available online at http://webstyleguide.com/.

MacDonald, L., Using color effectively in computer graphics, *IEEE Computer Graphics & Applications 19*, 4 (July/August 1999), 20–35.

Marcus, Aaron, *Graphic Design for Electronic Documents and User Interfaces*, ACM Press, New York (1992).

Mathur, A., Schlotfeldt, B., and Chetty, M., A mixed-methods study of mobile users' data usage practices in South Africa, *Proc. ACM International Joint Conference on Pervasive and Ubiquitous Computing (UbiComp 2015)*, Osaka, Japan (2015).

Mayer, Richard E., *Multimedia Learning,* 2nd Edition, Cambridge University Press, New York (2009).

Microsoft Developers Network, Error messages (2015). Available at https://msdn.microsoft.com/en-us/library/dn742471.aspx.

Mullet, Kevin, and Sano, Darrell, *Designing Visual Interfaces: Communication Oriented Techniques*, Sunsoft Press, Englewood Cliffs, NJ (1995).

Mumford, Lewis, *Technics and Civilization*, Harcourt Brace and World, New York (1934), 31–36.

NASA, NASA wins 2015 Webby Awards (2015). Available at http://www.jpl.nasa.gov/news/news.php?feature=4566.

Nass, Clifford, and Yen, Corina, *The Man Who Lied to His Laptop: What Machines Teach Us About Human Relationships*, Penguin Group (2010).

National Cancer Institute, U.S. Department of Health and Human Services, *Research-Based Web Design & Usability Guidelines* (2008). Available at http://www.usability.gov/pdfs/guidelines.html.

Nielsen Norman Group, *Top 10 Mistakes in Web Design* (2011). Available at http://www.nngroup.com/articles/top-10-mistakes-web-design/.

Nielsen Norman Group, *User Experience for Mobile Applications and Websites*, 3rd Edition (2015). Available at https://www.nngroup.com/reports/mobile-website-and-application-usability/.

Novick, D., Rhodes, J., and Wert, W., The communicative functions of animation in user interfaces, *Proceedings of the 29th ACM International Conference on Design of Communication (SIGDOC '11),* New York (2011), 1–8.

Oracle, *Java Look and Feel Design Guidelines* (2015). Available at http://www.oracle.com/technetwork/java/jlf-135985.html.

Parallax, *50 great parallax scrolling websites* (2015). Available at http://www.creativebloq.com/web-design/parallax-scrolling-1131762.

Plaisant, Catherine, Carr, David, and Shneiderman, Ben, Image browsers: Taxonomy and design guidelines, *IEEE Software 12,* 2 (March 1995), 21–32.

Plaisant, C., Shneiderman, B., and Chao, T., *Twinlist: Novel Interfaces for Medication Reconciliation* (2015). Available at http://www.youtube.com/watch?v=YXkq9hQppOw.

Rosling, Hans, 200 years that changed the world (2015). Available at https://www.youtube.com/watch?v=BPt8ElTQMIg.

Sarkar, Manojit, and Brown, Marc H., Graphical fisheye views, *Communications of the ACM 37*, 12 (July 1994), 73–84.

Shneiderman, Ben, Looking for the bright side of agents, *ACM Interactions 2,* 1 (January 1995), 13–15.

Smith, Sid L., and Mosier, Jane N., *Guidelines for Designing User Interface Software*, Report ESD-TR–86–278, Electronic Systems Division, MITRE Corporation, Bedford, MA (1986). Available from the National Technical Information Service, Springfield, VA.

Spence, Robert, *Information Visualization*, Addison-Wesley, Reading, MA (2001).

Stone, Maureen, *A Field Guide to Digital Color*, A. K. Peters, Wellesley, MA (2003).

Tidwell, Jennifer, *Designing Interfaces, Patterns for Effective Interface Design*, 2nd Edition, Sebastopol, CA: O'Reilly Media Inc. (2011), 8–24.

Tullis, T. S., Information presentation, in Proctor, R., and Vu, K. (Editors), *Handbook of Human Factors in Web Design*, Routledge, New York (2004).

Tullis, T. S., Web-based presentation of information: The top ten mistakes and why they are mistakes, *Proceedings of the HCI International 2005*, Las Vegas, NV, Lawrence Erlbaum Associates (July 2005).

Tversky, B., Morrison, J. B., and Betrancourt, M., Animation: Can it facilitate?, *International Journal of Human-Computer Studies 57*, 4 (2002), 247–262.

University of Texas, National Center for Cognitive Informatics & Decision Making in Healthcare, *Effective Table Design SEDB-G02* (2013). Available at https://sbmi.uth.edu/nccd/SED/Briefs/sedb-g02.htm.

Usability, *Research-Based Web Design and Usability Guidelines* (2015). Available at http://guidelines.usability.gov/.

Web Accessibility Initiative, *World Wide Web Consortium (W3C)* (2015). Available at http://www.w3.org/standards/webdesign/accessibility.

Weinman, Lynda, *Designing Web Graphics,* 4th Edition, New Riders, Indianapolis, IN (2002).


excluding merge commits
February
March
May
June
July
August
simransachar
23 commits / 27,350 ++ / 24,236 --
#1

CHAPTER 13

The Timely User Experience

> Stimulation is the indispensable requisite for pleasure in an experience, and the feeling of bare time is the least stimulating experience we can have.
>
> **William James**
> *Principles of Psychology,* Volume I, 1890

> Nothing can be more useful to a man than a determination not to be hurried.
>
> **Henry David Thoreau**
> *Journal*

CHAPTER OUTLINE

13.1 Introduction

For years, user perception of computer speed was determined by response time for mathematical computations, program compilations, or database searches. With the emergence of the World Wide Web, user expectations for expanded services grew, along with still more complex explanations of delays. Consider the speed (and accuracy) of weaving through internet search results. Users who wait many seconds for an app to load on their phones know why response time is an attraction.

Users do not always understand the impact among text, graphics, high-resolution images, video, and animation to appreciate the huge variations in server loads, network congestion, and proximity to wireless and the wireless speed. They also have to understand the multiple sources of problems, such as dropped connections, unavailable websites, and network outages. This complex set of concerns is usually discussed under the umbrella term *Quality of Service* (QoS). This term was originally derived from the telecommunications industry and still "rings true" here (pun not intended) where Quality of Service can be measured in terms of telephone call quality, lost connections, customer satisfaction, connection time, cost, and other factors.

Concern over the timely user experience stems from a basic human value: Time is precious. When externally imposed delays impede progress on a task, many people become frustrated, annoyed, and eventually angry. Apps may be abandoned within minutes if performance is inadequate. Customers will leave the website and order the same product from a competitor if they suspect the user experience will be better. Some users accept the situation with a shrug of their shoulders, but most users prefer to work as quickly as the software and connection allow.

Discussions of the timely user experience must also take into account a second basic human value: Harmful mistakes should be avoided. However, balancing rapid performance with low error rates sometimes means that the pace of work must slow. If users work too quickly, they may learn less, read with lower comprehension, commit more data-entry errors, and make more incorrect decisions. Stress can build in these situations, particularly in life-critical systems.

A third aspect of the timely user experience is reducing user frustration. Users may become frustrated enough to make mistakes or give up working. Delays are often a cause of frustration, but there are others, such as crashes that destroy data, software bugs that produce incorrect results, and poor designs that lead to user confusion. Networked environments generate further frustrations: unreliable service providers, the ever-present e-mail phishing or spam messages, and malicious viruses.

Timely user experience discussions usually focus on the decisions to be made by network designers and operators. This is appropriate because their decisions have a profound influence on many users. They also have the tools and knowledge to be helpful, and increasingly, they must adhere to legal and regulatory controls. Interface designers and builders can also make design decisions that dramatically influence the user experience. For example, they can optimize webpages to reduce byte counts and numbers of files or provide previews (e.g., thumbnails or coverage maps) of materials available in digital libraries or archives to help reduce the number of queries and accesses to the network (Fig. 13.1 and Section 15.1). For users, the main user experience is the *system response time (SRT)*, also referred to simply as "response time."

Human psychophysics provides context for the human factors related to time delays that lead to the timely user experience. People cannot perceive any difference in event times less than 25 milliseconds and have trouble perceiving them until those times approach 100 milliseconds. Human reaction time is another factor. Reaction time varies for each user (e.g., age difference, situation, operating an application in a stressed or life-critical environment versus a more

FIGURE 13.1

The Earthdata Search (http://search.earthdata.nasa.gov) indicates the geographic and temporal coverage of datasets before the data are downloaded. Here the user has selected two datasets. The MODIS dataset is tagged with the color blue, and the AIRS dataset is tagged with orange. Those colors are used on the timelines and on the map revealing where and when the two datasets overlap. Providing previews of data availability helps users find what they need with fewer queries and network accesses.

See also:

Chapter 4, Design

Chapter 12, Advancing the User Experience

casual one). In practice, users do not seem to be bothered much by 1-second delays in changing screens for PC applications and appear to tolerate somewhat longer times for websites to fully load.

Section 13.2 begins by discussing a model of SRT impacts, looks at SRT issues, reviews short-term human memory, and identifies the sources of human error. Section 13.3 focuses on the role of users' expectations and attitudes in shaping their subjective reactions to the timely user experience. Section 13.4 deals with user productivity and variability with respect to SRTs. Section 13.5 examines the severity of frustrating experiences, including spam and viruses.

13.2 Models of System Response Time (SRT) Impacts

SRT is defined as the number of seconds it takes from the moment a user initiates an action—usually by touching an icon, pressing the Enter key, or clicking a mouse—until the computer begins to present feedback. See Fig. 13.2 for the stages of this model.

When the response is completed, the user begins formulating the next action. The *user think time* is the number of seconds that elapse between the computer's response and the user's initiation of the next action. In this simple stages-of-action model, users (1) initiate, (2) wait for the computer to respond, (3) watch while the results appear, (4) scroll through results, (5) think for a while, and then initiate again.

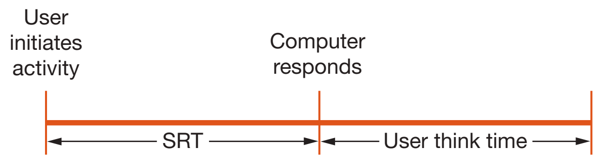


FIGURE 13.2

Simple stages of action model of SRT and user think time.

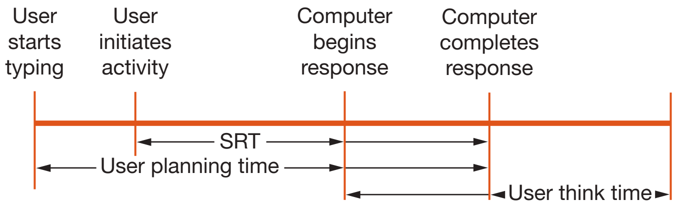


FIGURE 13.3

Model of SRT, user planning time, and user think time. This model is more realistic than the one in Fig. 13.2.

In a more realistic model (Fig. 13.3), users plan while interpreting results, while typing/clicking or touching an icon, and while the computer is generating results or retrieving information across the network. Most people will use whatever time they have to plan ahead; thus, precise measurements of user think time are difficult to obtain. The computer's response is usually more precisely defined and measurable, but there are problems here as well. Some interfaces respond with distracting messages, informative feedback, or a simple prompt immediately after an action is initiated, but actual results may not appear for a few seconds. Users prefer to have minimal, if any, delays for responses from networked devices (e.g., print status).

Designers who specify response times and network managers who seek to provide high system performance have to consider the complex interaction of technical feasibility, costs, task complexity, user expectations, speed of task performance, error rates, and error-handling procedures.

Although some people accept slower responses (latency) for some tasks (e.g., calculations), users prefer rapid interactions. Overall productivity depends not only on the speed of the interface but also on the rate of human error and the ease of recovery from those errors. Lengthy (longer than 5 seconds) response times are generally detrimental to productivity, increasing error rates and decreasing satisfaction. More rapid (less than 1 second) interactions are generally preferred and can increase productivity, but they may also increase error rates for complex tasks. The high cost of providing rapid response times and the loss from increased errors must be evaluated in the choice of an optimum pace.

13.2.1 Website SRT

Website display performance was studied by evaluating delay plus two website design variables (site breadth and content familiarity) to examine interaction effects on user performance, attitudes, stress, and behavioral intentions. The three experimental factors (delay, familiarity, and breadth) were demonstrated to collectively affect the cognitive costs and penalties that users incur when making choices in their search for target information (Galletta, 2006; Galletta et al., 2006). Interactive latency is explored by experiments in user behavior and knowledge discovery by Liu and Heer (2014). They found that interactive visualizations should provide improved SRTs to support effective exploration—that increased delays caused users to shift exploration strategy.

One study looks at measuring website performance after conducting thousands of experiments with users (Kohavi et al., 2014). Example rules of thumbs from the Kohavi et al. (2014) study include "Speed Matters A LOT" and "Small Changes Can Have a Big Impact to Key Metrics." These rules, based on thousands of controlled experiments with large, popular websites, guide designers in improving and organizing their sites so as to improve user satisfaction and performance.

Consumer demand is a key factor in promoting rapid performance. Websites often distinguish themselves with rapid performance—an attribute that surfers expect from Google or Yahoo! and buyers demand at Amazon.com or eBay (King, 2008).

Shortcuts to pre-stored consumer account information (e.g., shipping address) can improve the user experience for the online shopper. Another example is when an online shopper is prompted to enter the postal (zip in the United States) code first and then the city and state are automatically populated from a directory. That is a bit of order shift in entering address data, but the frequent shopper appreciates the keystroke savings. In addition, this promotes better accuracy and consistency in the address database.

13.2.2 Network impacts to SRT

Broadband service providers typically do not offer the same upload and download speeds. Those who require faster upload times—for example, webmasters, software developers working on collaborative projects, those uploading large content (e.g., video) to social media sites, or users who regularly transfer large files—might find that their broadband service providers have left much to be desired in terms of upload times. In an era of user-generated content, for an increasing number of users, it is important for upload speeds to keep pace with download capability.

Broadband service capability varies by country. According to a recent report (Akamai, 2015), South Korea leads the world with average internet connection speeds of 23.1 Mbps. Hong Kong is second with Japan third. The U.S. is 20th.

There are web tools that can permit computer users to assess their download and upload speeds (to find them, run a search for "download upload speed test"). Running this test gives users a better idea of the timely user experience and provides them with useful information to present to their broadband service providers when asking for better service or an upgrade to meet their network response-time needs.

Wired versus wireless networking (or both) is still a decision made by organizations and individuals, but with the vast improvement in wireless capability, the performance differences for many users have become unnoticeable. Entire communities embracing wireless networking aid in universal usability.

Some tasks (for example, videoconferencing, voice over IP telephony, and streaming multimedia) require rapid performance because intermittent delays cause jumpy images and broken sound patterns that seriously disrupt users. Promoters of these services see the need for ever-faster and higher-capacity networks.

While improvements are being made in the technology of computer-to-computer communication, user task performance times will not automatically improve at the same rate. Improved throughput does not necessarily imply improved productivity.

13.2.3 Factors for SRT modeling

A complete predictive model that accounts for SRT, designing interfaces, and formulating management policies may never be realized, but even fragments of such a model are useful to designers. *Cognitive shifts* (e.g., switching from scanning a news story to watching a video of the same news story) also can be difficult to measure in terms of productivity.

When using an interactive device, users may formulate plans and then have to wait while they execute each step in the plan. If a step produces an unexpected result or if the delays are long, the users may forget part of the plan or be forced to review the plan continually. This model leads to the conjecture that, for a given user and task, there is a preferred response time. Long response times lead to wasted effort and more errors because the solution plan must be reviewed repeatedly. On the other hand, short SRTs may generate a faster pace in which solution plans are prepared hastily and incompletely. More data from a variety of situations and users would clarify these conjectures.

The speed/accuracy tradeoff that is a harsh reality in so many domains is also apparent in interface usage. A related factor is performance in paced versus unpaced tasks. In paced tasks, the computer forces decisions within a fixed time period, thereby adding pressure. Such high-stress interfaces may be appropriate with trained users in life-critical situations or in manufacturing, where high productivity is a requirement. However, errors, poor-quality work, and operator burnout are serious concerns. In unpaced tasks, users decide when to respond and can work at a more relaxed pace, taking their time to make decisions.

Life-critical systems (e.g., avionics) provide an environment where a timely user experience can prevent catastrophe. One study looked at mixed criticality systems, where some features are safety-critical and some are not safety-critical.

This study looked at scheduling and execution times resulting in an optimal priority assignment approach to yield sufficient SRTs. (Baruah et al., 2011). Another study found that a reduction of variability in response times led to improved user performance (Weber et al., 2013).

SRT has been investigated and analyzed for years. Many factors have been measured with the results sometimes debated (Dabrowski and Munson, 2011):

- Users' accuracy and error rates with different levels of SRT
- User performance speed and the efficiency of the commands used
- How user interactions with the computer changed as a result of changes in SRT
- How their bodies reacted physiologically to changes
- How happy, satisfied, anxious, or annoyed users were as SRTs changed

This paper does a nice job of addressing the contrast between "control" and "conversation" tasks, where control tasks always want delay minimized but conversation tasks can tolerate modest delays without bothering users.

Accepted wisdom in the field is that car driving offers a useful analogy. Although higher speed limits are attractive to many drivers because they lead to faster completion of trips, they also lead to higher accident rates. Since automobile accidents can have dreadful consequences, drivers accept speed limits. When incorrect use of computer systems can lead to damage to life, property, or data, should speed limits be provided?

13.3 Expectations and Attitudes

How long will users wait for the computer to respond before they become annoyed? This simple question has provoked much discussion and many experiments. There is no simple answer, though, and more importantly, it may be the wrong question to ask. More refined questions focus on users' needs: Will users more happily wait for a valued document than an undesired advertisement?

The first factor influencing acceptable SRT is that people have established expectations based on their past experiences of the time required to complete a given task. If a task is completed more quickly than expected, people will be pleased, but if the task is completed much more quickly than expected, they may become concerned that something is wrong. Similarly, if a task is completed much more slowly than expected, users are likely to become concerned or frustrated.

An important design issue is that of rapid start-up. Users are annoyed if they have to wait for a device to be ready for usage or have to follow a complex set of "Getting Started" steps to get their device operational. Fast starts are a strong distinguishing feature in consumer electronics.

A second factor influencing SRT expectations is the individual's tolerance for delays. There are large variations in what individuals consider acceptable waiting time. These variations are influenced by many factors, such as personality, cost, age, mood, cultural context, time of day, noise, and perceived pressure to complete work. The laid-back web surfer may enjoy chatting with friends while pages load, but the anxious deadline-fighting journalist may start banging on his or her desk, device, or keyboard in a vain attempt to push the computer along.

Other factors influencing SRT expectations are the task complexity and the users' familiarity with the task. For simple, repetitive tasks that require little problem solving, users want to perform rapidly and are annoyed by delays. For complex problems, users will typically perform well even as SRT grows, as they can use the delays to plan ahead. Users are highly adaptive and can change their working styles to accommodate different response times.

Users may achieve rapid task performance, low error rates, and high satisfaction if the following criteria are met:

- Users have adequate knowledge of the objects and actions necessary for the problem-solving task.
- The solution plan can be carried out without delays.
- Distractions are eliminated.
- User anxiety is low.
- There is accurate feedback about progress toward the solution.
- Errors can be avoided or, if they occur, can be handled easily.

These conditions for optimum problem solving, with acceptable cost and technical feasibility, are the basic constraints on design. However, other conjectures may play a role in choosing the optimum interaction speed:

- Novices may exhibit better performance with somewhat slower response times.
- Novices prefer to work at speeds slower than those chosen by knowledgeable, frequent users.
- When there is little penalty for an error, users prefer to work more quickly.
- When the task is familiar and easily comprehended, users prefer more rapid action.
- If users have experienced rapid performance previously, they will expect and demand it in future situations.

BOX 13.1

Primary factors.

1. Previous experiences
2. Individual personality differences
3. Task differences

An increasing number of tasks place high demands on rapid system performance; examples are user-controlled 3-D animations, flight simulations, graphic design, and dynamic queries for data visualization. In these applications, users are continuously adjusting the input controls, and they expect changes to appear with no perceived delay.

Three primary factors that influence user expectations and attitudes regarding SRT appear in Box 13.1.

Three conjectures emerge:

1. Individual differences are large, and users are adaptive. They will work faster as they gain experience and will change their working strategies as SRTs change. It may be useful to allow people to set their own pace of interaction.
2. For repetitive tasks, users prefer and will work more rapidly with short response times.
3. For complex tasks, users can adapt to working with slow SRTs with no loss of productivity, but their dissatisfaction increases as response times lengthen.

13.4 User Productivity and Variability in SRT

Shorter SRTs usually lead to higher productivity, but in some situations, users who encounter long SRTs can find clever shortcuts or ways to do concurrent processing to reduce the effort and time required to accomplish a task. Working too quickly, though, may lead to errors that reduce productivity.

In computing, just as in driving, there is no general rule about whether the high-speed highway or the slower, clever shortcut is better. The designer must survey each situation carefully to make the optimal choice. The choice is not critical for occasional usage, but it becomes worthy of investigation when the frequency is great. When computers are used in high-volume situations, more effort can be expended in discovering the proper response time for a given task and set of users. It should not be surprising that a new study must be conducted when the tasks and users change, just as a new route evaluation must be done for each trip.

An alternative solution is masking delay by displaying important, crucial information first while the background is filling in. Well-designed websites often download critical information first; likewise, web designers may choose to download the intriguing information first so the user is motivated and encouraged to wait during any download delay to see the end result. Some news websites download the textual headlines first to motivate the news reader to remain patient while the remainder of the article is downloaded. The user can then start reading an article while additional animations, advertisements, and so on, download, until eventually the screen is fully painted with its intended information.

The nature of the task has a strong influence on whether changes in SRT alter user productivity. A *repetitive control task* involves monitoring a display and issuing actions in response to changes in the display. With shorter SRTs, the operator picks up the pace of the system and works more quickly, but decisions on actions may be less than optimal. On the other hand, with short SRTs, the penalty for a poor choice may be small because it may be easy to try another action. In fact, operators may learn to use the interface more quickly with short SRTs because they can explore alternatives more easily.

When using computers, users cannot see into the machines to gain reassurance that their actions are being executed properly, but the SRT can provide a clue. If users come to expect a response time of 3 seconds for a common action, they may become apprehensive if this action takes 0.5 or 15 seconds. Such extreme variation is unsettling and should be prevented or acknowledged by the interface, with some indicator for an unusually fast response or a progress indicator for an unusually slow response. Progress indicators need to be truthful representations of the state of affairs (Fig. 13.4). Providing time estimates is best, but when that information is difficult to calculate, other progress indicators—such as the name of the file being processed or percent complete—can be updated at regular intervals. How often have computer users been lulled into an increasingly frustrating state of anticipation, for example, watching a webpage download-indicator status bar show that the page is loading, only to find that the internet connection has been lost or the server is down? A survey of progress indicators with guidance for timing was performed by Sherwin (2014).

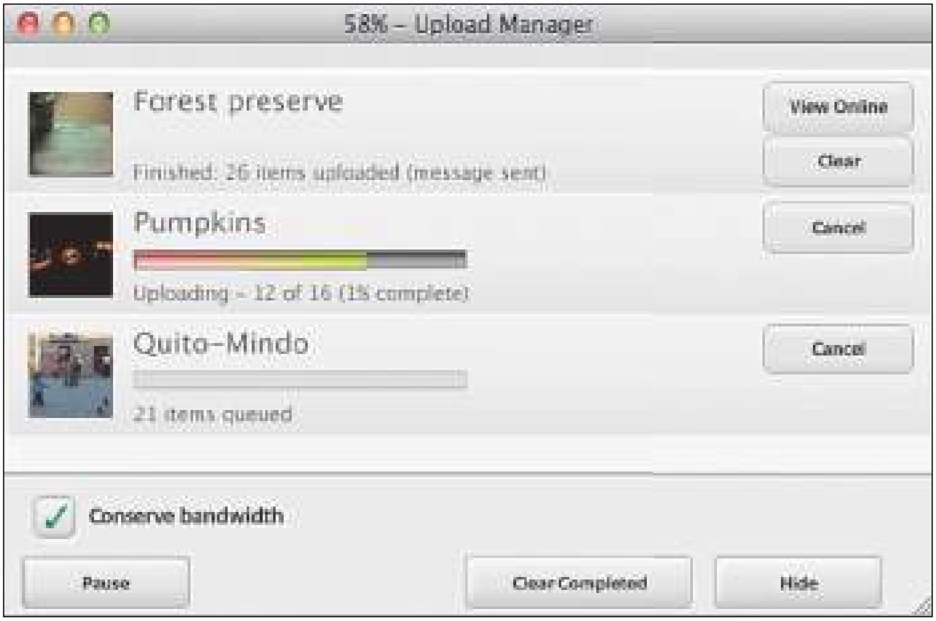


FIGURE 13.4

Progress indicators reassure users that the process of uploading photos from Picasa to the web is under way and how far it has gone already. They also allow users to see the results or to cancel uploads.

13.5 Frustrating Experiences

Many technology promoters argue that the quality of user experiences with computers has been improving, pointing to steadily increasing memory, storage, and network capacities. However, critics believe frustration from interface and technology complexity, network disruptions, and malicious interference has grown. Recent research has begun to document and help explain the sources of user frustration with contemporary user interfaces.

Interruptions appear to be troubling to users regardless of whether they originate from the current task or an unrelated task, but surprisingly, people have been shown to complete interrupted tasks in less time than uninterrupted tasks and with no difference in quality (Mark et al., 2008). The authors of this study conjectured that people compensate for interruptions by working faster; however, this comes at the price of more stress and higher frustration, time pressure, and effort. An appropriate interface design change would allow users to limit interruptions, reducing their negative effects and thus reducing the time it takes to solve problems.

There are numerous reports and trade journal articles about time wasted at work due to distractions, technology, and office layout or people issues. There are differences in the office versus student work environment and how those contribute to user frustration. One study of 107 student computer users and 50 workplace computer users shows high levels of frustration and loss of 1/3 to 1/2 of time spent (Lazar et al., 2006).

Focusing on technological (not people) reasons for poor productivity, frequent complaints include network performance during peak periods, application crashes, long SRTs, and confusing error handling. The major sources of problems were the popular applications for web browsing, e-mail, social media, and word processing. Recommendations for reducing frustration include interface redesign, software quality improvement, and network reliability increases. Other recommendations focus on what users can do through increased learning, careful use of services, and self-control of their attitudes.

Infrastructure improvements to server capacity and network speed and reliability will improve user experiences. Improved network performance and reliability promote trust in users, easing their concerns and ultimately improving work performance and output. Consequently, untimely user experience is a still greater difficulty in emerging markets and developing nations, where infrastructure reliability remains a problem.

Since user training can have a profound influence on reducing frustrating experiences, efforts to improve education through schools and workplaces could improve user experiences. Improved educational programs and refined user interfaces are likely to have the largest effect on poorly educated users, whose difficulties in using internet services undermine efforts to provide e-learning, e-commerce, and e-government services.

Networked services, especially e-mail, are among the most valued benefits of information and communications technologies. There are numerous sources of information on *netiquette*, proper usage, and productivity to guide users in the proper use of e-mail. Many corporations publish e-mail guidelines not only to coach their employees on the proper use of e-mail in the workplace but also to address best practices for reducing e-mail information overload, thus enhancing workplace productivity.

E-mail has become the source of frustrating *spam* (the pejorative term given to unwanted, unsolicited e-mail, including advertisements, personal solicitations, and pornographic invitations). Much spam comes from small companies and individuals who take advantage of the low cost of e-mail to send blanket notices to huge lists of unfiltered e-mail addresses. Anti-spam legislation has passed in many nations, but the internet's international reach and open policies limit the success of legal controls. Much of the spam is intercepted or filtered (by system or user tools), but users still complain of too much spam.

Another frustrating problem for users is the prevalence of malicious viruses that, once installed on a machine, can destroy data, disrupt usage, or produce a cancerous spread of the virus to everyone on the user's e-mail contact list. Viruses are created by malevolent programmers who want to spread havoc, usually via e-mail attachments. Unsuspecting recipients may get an infected e-mail from a known correspondent, but the absence of a meaningful subject line or message is often a clue that the e-mail contains a virus. Deceptive messages that mention previous e-mails or make appealing invitations complicate user decisions, but safety-conscious users will not open attachments from unknown or deceptive senders.

Universal usability presents its own set of challenges in terms of user frustration. In one research project, 100 blind users, using time diaries, recorded their frustrations using the web (Lazar et al., 2007). The top causes of frustration reported were (1) page layout causing confusing screen-reader feedback; (2) conflict between the screen reader and the application; (3) poorly designed, unlabeled forms; (4) no alternative text for pictures; and (5) a three-way tie between misleading links, inaccessible PDFs, and screen-reader crashes. In this study, the blind users reported losing, on average, 30.4% of their time due to these frustrating situations. Web designers concerned with universal usability can improve matters by using more appropriate form and graphic labels and avoiding confusing page layouts.

Since frustration, distractions, and interruptions can impede smooth progress, design strategies should enable users to maintain concentration. Three initial strategies can reduce user frustration: reduce short-term and working memory load, provide information-abundant interfaces, and increase automaticity (Shneiderman and Bederson, 2005). Automaticity in this context is the processing of information (in response to stimuli) in a way that is automatic and involuntary, occurring without conscious control. An example is when a user performs a complex sequence of actions with only a light cognitive load, like a driver following a familiar route to work with little apparent effort.

Practitioner's Summary

The timely user experience is a constant concern for users and providers on networks, computers, and mobile devices. Rapid SRTs with fast screen refreshes are necessary because these factors are determinants of user productivity, error rates, working style, and satisfaction (Box 13.2). In most situations, shorter response times (less than 1 second) lead to higher productivity. For mouse actions, multimedia performances, and interactive animations, even faster performance is necessary (less than 0.1 second). Satisfaction generally increases as the response time decreases, but there may be a danger from stress induced by a rapid pace. As users pick up the pace of the system, they may make more errors. If these errors are detected and corrected easily, productivity will generally increase. However, if errors are hard to detect or are excessively costly, a moderate pace may be most beneficial.

Designers can determine the optimal SRT for a specific application and user community by measuring productivity, errors, and the cost of providing short response times. Managers must be alert to changes in work style as the pace quickens; productivity is measured by correctly completed tasks rather than by interactions per hour. Novices may prefer a slower pace of interaction. Methods for helping the successful user transition from novice to expert are described in one study where the state of the art of user interfaces that promotes expertise development is examined, based on the human factors of learning and skill development (Cockburn et al., 2014).

Modest variations around the mean SRT are acceptable, but large variations (less than one-quarter of the mean or more than twice the mean) should be accompanied by informative messages. An alternative approach for overly rapid responses is to slow them down and thus to avoid the need for explanatory messages.

BOX 13.2

SRT guidelines.

- Users prefer shorter response times.
- Longer response times (> 15 seconds) are disruptive.
- Users' usage profiles change as a function of response time.
- Shorter response time leads to shorter user think time.
- A faster pace may increase productivity, but it may also increase error rates.
- Error-recovery ease and time influence optimal response time.
- Response time should be appropriate to the task:
  - Typing, cursor motion, mouse selection: 50–150 milliseconds
  - Simple, frequent tasks: 1 second
  - Common tasks: 2–4 seconds
  - Complex tasks: 8–12 seconds
- Users should be advised of long delays.
- Strive to have rapid start-ups.
- Modest variability in response time is acceptable.
- Unexpected delays may be disruptive.
- Offer users a choice in the pace of interaction.
- Empirical tests can help to set suitable response times.

BOX 13.3

Reducing user frustration.

- Increase server capacity, network speed, and network reliability.
- Improve user training, online help, and online documentation including tutorials.
- Redesign instructions and error messages.
- Continue the battle to stay ahead of the technology to protect users against spam, viruses, and pop-up advertisements.
- Organize consumer-protection groups.
- Increase research on user frustration.
- Catalyze public discussion to raise awareness.

A continuing concern is the frustration level of the increasingly diverse set of computer users (Box 13.3). In an era of user-generated content and pervasive social media involvement, a satisfying user experience is determined by a preferred or, at least, an acceptable level of system timeliness. Malicious spreaders of spam and viruses are a serious threat to the expanding community of internet users. Application crashes, confusing error messages, and network disruptions are problems that could be addressed by improved interface and software design.

Practitioners could look at other ways of helping users to become experts (e.g., how to provide shortcuts). Reminder regarding universal usability: Aiding new users to learn gestures for new touchscreen or wearable devices should be easier.

Researcher's Agenda

The increased understanding of the timely user experience issues today is balanced by the richness of new technologies and applications. The taxonomy of issues provides a framework for research, but a finer taxonomy of tasks, of relevant cognitive-style differences, and of applications is needed. Next, a refined theory of problem solving and consumer behavior is necessary to generate useful design hypotheses.

It would be productive to study error rates as a function of SRT for a range of tasks and users. Another goal is to accommodate the real-world interruptions that disrupt planning, interfere with decision making, and reduce productivity.

It is understandable that error rates vary with SRT, but how else are users' work styles or consumers' expectations affected? Is employee multitasking coupled with routine office distractions adding stress and drastically reducing productivity, ultimately affecting corporate profits? Can modern users be trained to better manage their time among diverse applications and tasks? Can users be encouraged to be more careful in their decisions by merely lengthening response times and degrading the timely user experience? Does the profile of actions shift to a smaller set of more familiar actions as the SRT shortens?

Many other questions are also worthy of investigation. When technical feasibility prevents short responses, can users be satisfied by diversionary tasks, or are progress reports sufficient? Do warnings of long responses or apologies relieve anxiety or simply further frustrate users?

Future SRT research is needed to analyze the impact of delays (both human and system-caused), cognitive load, and variability to optimize user performance. Methods for assessing user frustration levels are controversial. Time diaries may be more reliable than retrospective surveys, but how could automated logging and observational techniques be made more effective? How can designers more effectively use big data and web analytics and reporting (e.g., Google Analytics) to better manage organizations? How could software developers and network providers construct reliable reports to gauge improvements in the timely user experience and reductions in user frustration?

WORLD WIDE WEB RESOURCES

www.pearsonglobaleditions.com/shneiderman

SRT (System Response Time) issues have an increasing presence on the internet, although the issue of long network delays gets discussed frequently. User frustration is a lively topic, and many websites point out flawed interfaces and related frustrating experiences. The New Computing movement's website suggests ways to help bring about change:

- New Computing: http://www.cs.umd.edu/hcil/newcomputing

Discussion Questions

For questions 1–4, refer to the following scenario:
A group decision system is being built over a network, which has inherent delays due to network lag. Suggest the longest acceptable amount of time the delay could last without affecting the user negatively. Provide an argument for the time you selected.

1. System confirming the user's password when logging on.
2. Synchronous group editing of a document.
3. Asynchronous critiquing of other participants' work.
4. Voting on serious issues.
5. Discuss three human values that are necessary to be understood by interface designers in order to ensure a timely user experience.
6. How are download delays masked by well-designed websites?
7. State a few system response time (SRT) guidelines.

References

Akamai, *State of the Internet: Q3 2015 Report* (2015), 14–23. Available at https://www.stateoftheinternet.com/resources-report-state-of-the-internet.html.

Baruah, S., Burns, A., and Davis, R., Response time analysis for mixed criticality systems, *2011 IEEE 32nd Real-Time Systems Symposium (RTSS) (and IEEE Explore Digital Library)*, (November 2011), 34–43.

Cockburn, A., Gutwin, C., Scarr, J., and Malacria, S., Supporting novice to expert transitions in user interfaces, *ACM Computing Surveys 47*, 2 (November 2014), Article 31.

Dabrowski, J., and Munson, E., 40 years of searching for the best computer system response time, *Interacting with Computers 23*, 5 (2011), 555–564.

Galletta, Dennis F., Understanding the direct and interaction effects of web delay and related factors, in Galletta, Dennis F., and Zhang, Ping (Editors), *Human-Computer Interaction and Management Information Systems: Applications (Advances in Management Information Systems)*, M. E. Sharpe, Armonk, New York (2006), 29–69.

Galletta, Dennis F., Henry, Raymond, McCoy, Scott, and Polak, Peter, When the wait isn't so bad, *Information Systems Research 17*, 1 (March 2006), 20–37.

King, Andrew B., *Website Optimization: Speed, Search Engine & Conversion Rate Secrets*, O'Reilly Media, Sebastopol, CA (2008).

Kohavi, R., Deng, A., Longbotham, R., and Ya Xu, Y., Seven rules of thumb for web site experimenters, *Proceedings of the 20th ACM SIGKDD International Conference on Knowledge Discovery and Data Mining (KDD '14)* (2014), 1857–1866.

Lazar, J., Allen, A., Kleinman, J., and Malarkey, C., What frustrates screen reader users on the Web: A study of 100 blind users, *International Journal of Human-Computer Interaction 22*, 3 (May 2007), 247–269.

Lazar, J., Jones, A., Hackley, M., and Shneiderman, B., Severity and impact of computer user frustration: A comparison of student and workplace users, *Interacting with Computers 18*, 2 (March 2006), 187–207.

Liu, Zhicheng, and Heer, Jeffrey, The effects of interactive latency on exploratory visual analysis, *IEEE Transactions Visualization & Computer Graphics* (Proc. InfoVis) (2014).

Mark, Gloria, Gudith, Daniela, and Klocke, Ulrich, The cost of interrupted work: More speed and stress—Don't interrupt me, *Proceedings of the Conference on Human Factors in Computing Systems*, ACM Press, New York (2008), 107–110.

Sherwin, Katie, Progress indicators make a slow system less insufferable (2014). Available at http://www.nngroup.com/articles/progress-indicators/.

Shneiderman, Ben, and Bederson, Ben, Maintaining concentration to achieve task completion, *Proceedings of the Conference on Designing for User Experiences* 135, ACM Press, New York (November 2005), 2–7.

Weber, F., Haering, C., and Thomaschke, R., Improving the human–computer dialogue with increased temporal predictability, *Human Factors: The Journal of the Human Factors and Ergonomics Society 55* (October 2013), 881–892.

Draw annotations
Make text annotations
Make a new song
Explore the ribbon
Hotel & Flight Info

CHAPTER 14

Documentation and User Support (a.k.a. Help)

> We learn by example and by direct experience because there are real limits to the adequacy of verbal instruction.
>
> **Malcolm Gladwell**
> *Blink: The Power of Thinking without Thinking*, 2005

> Our life is frittered away by detail. Simplify, simplify.
>
> **Thoreau**

CHAPTER OUTLINE

14.1 Introduction

Standardization and improvements in user interfaces have made computer applications easier to use, but using new interfaces is still a challenge. First-time computer users struggle to understand basic interface icons and actions as well as their tasks. Even for experienced users, learning advanced features and understanding novel task domains take commitment and concentration. Many users learn from someone else who knows the interface; others learn by trial and error, while others use the supplied (typically online) documentation; and yet others use outside sources, be they online or externally produced books. Over time, the reliance on formal user manuals, printed documentation, and tutorials has been less frequent. They are being replaced by customizable requests for help using the easier-to-use (and usually available) online search facilities. But some users still want this type of documentation as can be seen in the success of publishing groups with series such as *For Dummies*, *Missing Manuals*, and *Teach Yourself Visually*.

Learning anything new is a challenge. Challenges can be joyous and satisfying, but when it comes to learning about computer systems, many people experience anxiety, frustration, and disappointment. Much of the difficulty stems directly from the poor design of the menus, displays, or instructions or simply from users' inability to easily determine what to do next. As the goal of providing universal usability becomes more prominent, online help services are increasingly necessary to bridge the gap between what users know and what they need to know.

Studies have shown that well-written and well-designed user manuals, on paper or online, can be effective, but today's users show little interest in detailed manuals. Today's user interfaces are expected to provide everything online, supplemented by quick-start guides and interactive tutorials that serve user needs for training and later on for continued reference. In fact, as displays appear in cars, phones, cameras, public kiosks, mobile devices, and elsewhere, ubiquitous and customizable online help should be the norm. Increasing attention is being paid to improving user-interface design, but the complexity and diversity of interactive applications are also growing. Evidence suggests that there will always be a need for supplemental materials that aid users in various formats (though the use of printed manuals seems to be fading).

There are diverse ways of providing guidance to users online, ranging from a simple pop-up box (often called a *tool tip*, *ScreenTip*, or *balloon help*) to more advanced assistants and wizards. Most user interfaces have frequently asked questions (FAQs) and lively user communities that provide a more "grassroots" type of help and support. These communities are often supported or sponsored by companies to enable users to solve problems and provide product

See also:
Chapter 5, Evaluation and the User Experience
Chapter 8, Fluid Navigation
Chapter 11, Communication and Collaboration
Chapter 15, Information Search

improvements. This user assistance may be available through formal online and structured user communities and newsgroups or more informal e-mail, chat, and instant messaging. A broad variety of formats and styles are used to meet the ever-present need for documentation and user support (Earle et al., 2015).

This chapter starts by discussing shaping the contents of documentation including writing for the web (Section 14.2) and then continues with accessing the documentation (Section 14.3). Section 14.4 explores the differences found reading from displays versus reading from paper. Next, tutorials and demonstrations are discussed in Section 14.5, and online communities for user assistance and support (Section 14.6) are reviewed. The chapter closes with a brief section on the development process for user documentation (Section 14.7).

14.2 Shaping the Content of the Documentation

Traditional training and reference materials for computer systems were paper manuals, frequently left to the most junior member of the development team as a low-effort task to be completed at the end of the project. As a result, the manuals often were poorly written, were not suited to the users' backgrounds, were delayed or incomplete, and were tested inadequately if at all. Documentation has been referred to as "The Cinderella of Information Systems" (van Loggem, 2013). Today, managers recognize that designers may not fully understand the users' needs, that system developers might not be good writers, and that it takes time and skill to write effective documentation. They have also learned that testing and revision must be done before widespread dissemination and that system success is closely coupled to documentation quality. Users show little interest in reading manuals or documentation from cover to cover; their interest is focused on gathering information or completing a task (Redish, 2012). Content is now considered important enough that it has morphed into its own field, with books and guides about writing good content (Redish, 2012; Handley, 2014). Writing content exclusively for the web has its own guidelines (see Section 14.2.2).

Many users are technologically sophisticated. There is no appeal to reading or browsing through many pages of documentation; users want to get going quickly with their technology products. Users expect an easy learning curve and the actions to perform the tasks at hand to be easy to find and discover. Only with the most sophisticated computer systems is there a need or desire for extensive training and long start-up times. Users want quick-start guides (Fig. 14.1), easy-to-navigate material, and lots of examples (Novick and Ward, 2006a); they also want content on their preferred medium including, video and audio, topic-based information, embedded help, animations, and 3-D graphics (Hackos, 2015) Some of these attributes are difficult to produce in general documentation for a wide-ranging audience.

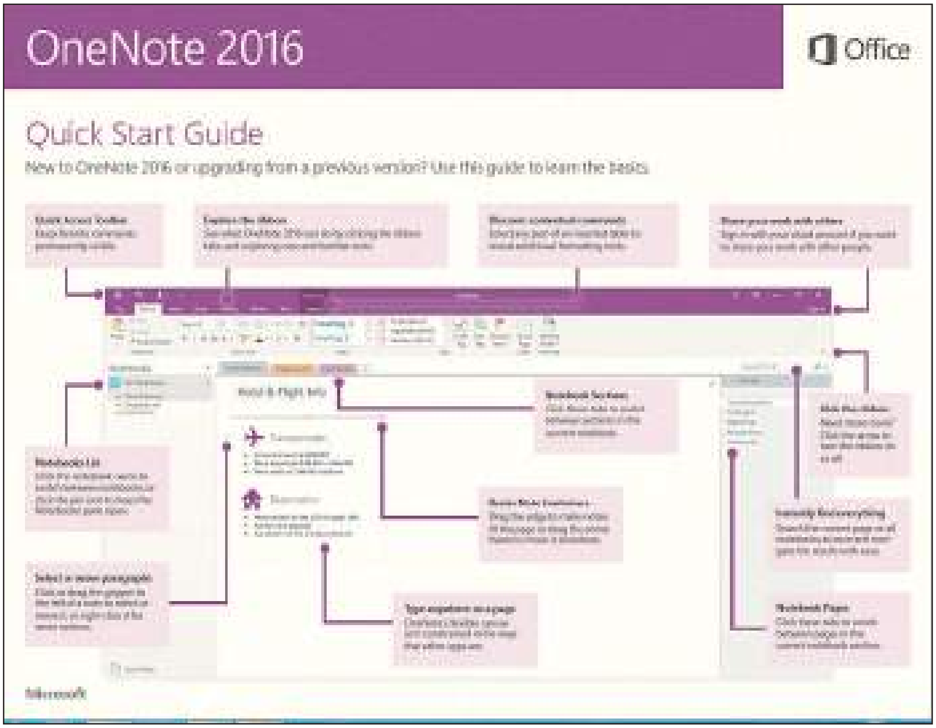


FIGURE 14.1

This particular quick-start guide for Microsoft OneNote 2016 points out common actions for users. Its goal is to minimize the learning curve for users upgrading from a previous version. The quick-start guide is four pages long; this is the first page. Explanations are provided for the toolbar and the ribbon in addition to other commonly used features.

14.2.1 Organization and writing style

Designing documentation is a challenging endeavor. The author must be knowledgeable about the technical content; sensitive to the background, reading level, native language, and intellectual ability of the reader; and skilled in writing lucid prose. Assuming that the author has acquired the technical content, the primary job in creating documentation is to understand the readers and the tasks that they must perform.

A precise statement of the *instructional objectives* is an invaluable guide to the author and the reader. The sequencing of the instructional content should be governed by the reader's current knowledge and ultimate objectives. Precise rules are hard to identify, but the author should attempt to present concepts in a logical sequence in increasing order of difficulty to ensure that each concept is explained before being used in subsequent sections, to avoid forward references, and to construct sections that contain approximately equal amounts of new material. In addition to these structural requirements, the documentation should have sufficient examples and complete sample sessions. These guidelines are valid for manuals and other documentation that is typically read sequentially. When writing for sophisticated interfaces, where approaches are not sequential, new techniques may need to be tried. A mantra of many technical writers is "write once, publish many places," as they are aware that the content will be published in multiple formats and in multiple places. Companies such as Acrolinx can help the writer with this process.

Before starting to write any documentation, it's important to thoroughly understand who the intended users are and how and where the documentation will be used. Frampton (2008) suggests numerous questions that should be considered: Who is the intended audience for the documentation? What are the market expectations for the documentation? What amount of the budget has been allocated for it? Are some components of the documentation considered essential or required and others supplemental or nice to have? How will the documentation be used? Will it be used once and thrown away or used repeatedly over a long period of time? What is the reading level of the potential users? Is the documentation written in their native language? What is the intended user's level of comfort with technology? Following the user-centered design process is a good way for the documentation authors to communicate and discuss requirements with the users, and this approach will yield better-designed documentation.

Redish (2012) encourages authors to divide documentation into topics and subtopics. Topics can be organized according to time or sequence, tasks, people, type of information presented, or questions people ask. Today's online world is one of agile information development, and technical information development is undergoing many changes (Hackos, 2015). Development cycles are short, and competition is fierce. A trend in technical communication since the late 1990s is to write documentation based on standards, such as Darwin Information Typing Architecture (DITA). The DITA standard (http://dita.xml.org) emphasizes developing and organizing content based on three information types: concept, task, and reference. These information types represent "chunks" of content that can be reused and, importantly, published across multiple platforms (Fig. 14.2). Adhering to a standard also enables sharing of content between different groups within and outside an organization. For example, a training group can reuse content from the technical writing group.

Users interact with the documentation on several different cognitive levels. They go to the documentation to find information that is relevant to accomplishing a task. They need to understand what the documentation is explaining, and they then need to apply that understanding to the task that caused them to consult the documentation in the first place. In this process, there are lots of places for misunderstandings and increased cognitive load. Additionally, users may already be in a stressful situation and frustrated because the interface is not letting them accomplish their task.

The choice of words and their phrasing is as important as the overall structure. A poorly written sentence mars a well-designed manual, just as an incorrect note mars a beautifully composed sonata. *The Elements of Style* (Strunk et al., 2000) and *On Writing Well* (Zinsser, 2006) are two classic references. Other more

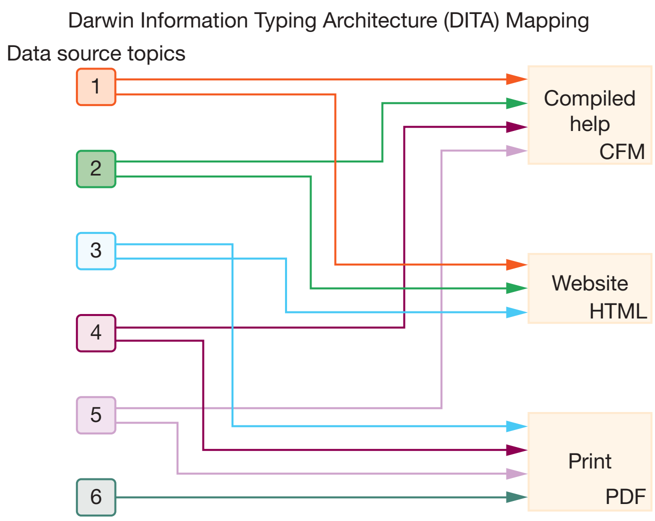


FIGURE 14.2

The documentation content is organized as single source topics created, during the authoring process, with the intention of reuse. The various documentation end products (compiled help, website, print) are mapped from the various combinations of the single source topics.

recently updated books on writing include *Everybody Writes* (Handley, 2014), *Letting Go of the Words* (Redish, 2012), *Every Page Is Page One* (Baker, 2013), *Designing for Digital Reading* (Pearson et al., 2014), *Conversation and Community* (Gentle, 2012), *Designing Information* (Katz, 2012), *The Essentials of Technical Communication* (Tebeaux and Dragga, 2014), *Solving Problems in Technical Communication* (Johnson-Eilola and Selber, 2013), and *The Sense of Style* (Pinker, 2014). Specialized books such as *The Global English Style Guide* (Kohl, 2008) can help with writing for global audiences, and *ReaderCentric Writing for Digital Media* (Hailey, 2015) provides a theoretical framework to improve online media content. Style guides for organizations represent worthy attempts at ensuring consistency and high quality (see Chapter 1). But, of course, no set of guidelines can turn a mediocre writer into a great writer. Writing is a highly creative act; effective content and documentation writers are needed.

There are numerous resources available for professional communicators, with an emphasis on technical communication. Formal courses and degree programs exist as well as specialized institutes and workshops. Books (as mentioned above) exist to explain techniques for writing documentation (and specifically web content) as well as formal pedagogy. The IEEE (through its Professional Communication Society), the ACM's Special Interest Group on Design of Communication (SIGDOC), and the Society for Technical Communication (STC) provide theoretical publications as well as information of a more practical nature. Redish (2010) discusses the intertwining of technical communication and usability, how both fields can and do learn from one another, and how the combination adds strength to both. Studies are also under way to see how crowd sourcing can be used to improve writing (Bernstein et al., 2015).

14.2.2 Writing for the web

When writing for the web, it is important to be aware of the differences from conventional writing and reading practices. When the web is utilized on mobile devices, screen sizes can be small and screen real estate is critical, so the designer needs to be aware and make every word count. In journalism, there is the ongoing discussion about the pros and cons of the *inverted pyramid* style of writing. In the inverted pyramid the most important information or the main point comes first, then the remaining information follows with the least important information at the end (Redish, 2012). This works well for the web since it has been shown that users typically don't scroll.

Gaining a better understanding of users' reading patterns in online environments is important. A study using eye-tracking and displaying where and how users viewed a webpage clearly shows users read following an F-shaped pattern (Fig. 14.3). This indicates that users do not read online text word by word. The beginning paragraphs should contain the most important information; after reading them, users tend to continue to scan down the left side of the page, so

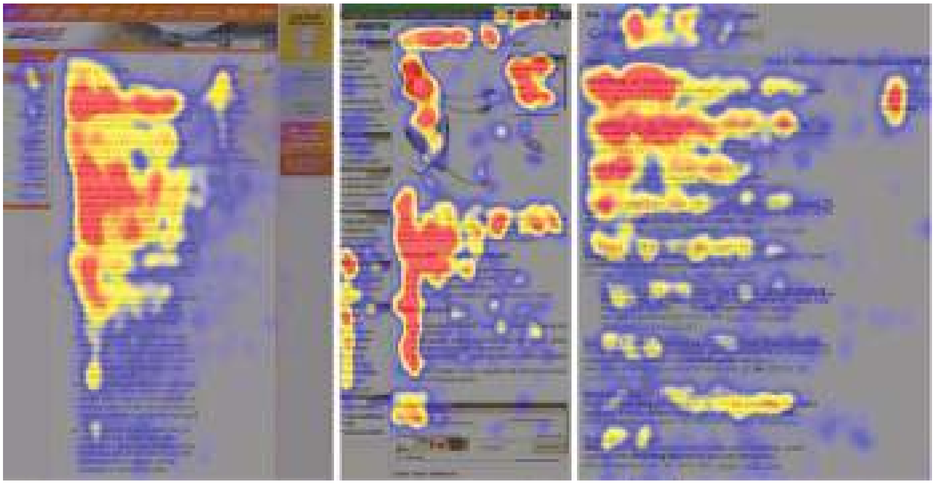


FIGURE 14.3

In heat maps from the eye-tracking study, red indicates the area where the user looked most, yellow indicates fewer views, and blue indicates the fewest views. Gray is used for areas that were not viewed. The image on the left is from an article in the "About Us" section of a corporate website, the center image is a product page on an e-commerce website, and the image on the right is from a search engine results page (https://www.nngroup.com/articles/f-shaped-pattern-reading-web-content/ and Jakob Nielsen).

the words they see there should also carry the important information content (Nielsen, 2006).

Another approach coined by Leslie O'Flahavan (founder of ewriteonline.com) is "a bite, a snack, and a meal." This describes the varying amount of content presented to the reader. A bite is the headline or main point. A snack is a bit more detail and perhaps an article summary or some cogent points. A meal is the entire article. The way content is read is changing, and that affects how content is written. Now readers not only view the content but also want to easily share the content with friends, coworkers, and so on. Web content has to be fluid across platforms.

Box 14.1 offers some design maxims to follow when writing for the web. It is important to understand the reason the user is coming to the website. Users may be coming for content; they are not there to reflect upon the design. Users come for information to answer a question or to help them complete a task; that information should be easy to find and easy to understand. Think about every time the web content is accessed as the beginning of a conversation started by the site visitor (user). Obtain as much information as you can about your users.

BOX 14.1

Helpful advice when writing for the web (Redish, 2012).

- Break your text into short sections with clear headings.
- Start with your key message.
- Write short sentences and short paragraphs.
- Use lists and tables—tables may not work well for mobile devices.
- Write meaningful links (do not use "click here").
- Illustrate your content.

14.3 Accessing the Documentation

Studies in the past have confirmed that well-designed documentation can be very effective. In spite of improvements, however, most users avoid user manuals if they can. If users read the documentation, they "satisfice," skip, scan, and skim (Mehlenbacher, 2009). Users typically do not want to sift through voluminous user manuals that can be difficult to navigate. Instead, they want quick and easy access to instructions for the specific tasks they are trying to carry out (Redish, 2012). Even when problems arise, many users are reluctant to consult written documentation and may do so only as a last resort. Although guidelines have been applied to improve the design of online components to take advantage of their unique media, studies still show low use of documentation (Novick et al., 2007).

Standard formats such as HTML Help, Java Help, and DITA have stimulated development of a growing number of software tools to generate help files, such as Adobe RoboHelp™ (http://adobe.com/products/robohelp.html) and oXygen WebHelp (https://www.oxygenxml.com/xml_author/webhelp.html). These tools facilitate coordination among teams of authors in creating interactive online help in multiple formats for multiple platforms.

Documentation is often placed online for good reasons. The issue becomes one of making the best use of the online environment when accessing the material. Multiple ways are available to search and traverse the online information differently from paper documentation. One feature of online documentation is the ability to offer pop-up help as well as customized help for various user populations, such as users with disabilities, international users, and users in varying age ranges.

14.3.1 Online documentation

The low production and shipping costs of CD-ROMs initially encouraged hardware suppliers to produce online documentation that was an exact duplicate of the paper documentation and/or manuals. Most manufacturers today put their user documentation directly online and no longer create supporting paper-based documentation. Modern designs assume that online documentation or web-based documentation will be available, usually with standard browsing interfaces to reduce learning effort. For mobile devices, small displays limit the possibilities, but providing helpful instructions on the device to complement printed user documentation should still be a priority. To keep this information up to date, users are often referred to the manufacturer's website, where downloadable manuals and other forms of documentation, such as frequently asked questions (FAQs), are often readily available.

A vital feature for online documentation—especially manuals—is a properly designed table of contents that can remain visible to the side of the displayed page of text. Selection of a chapter or other entry in the table of contents (Fig. 12.5) should immediately result in the appropriate page being displayed. Use of expanding or contracting tables of contents (the common use of the plus and minus signs) or multiple panes to show several levels at once can be beneficial. Being able to conveniently and easily navigate using hyperlinks through large volumes of online documentation is vital for the user.

Although they are frequently generated from the same source document (usually an XML or XHTML document), online documentation now tends to differ from paper documentation, benefitting from all the physical advantages, navigation features, and interactive services available online (see Box 14.2). On the other hand, paper documents have traditionally housed supplementary local information that is often written in margins or included on slips of paper stuck in at the appropriate pages. Some printed documentation remains pristine, neatly encased in its original shrink-wrapped packaging, but dog-eared, stained, and worn pages are often seen on well-used documentation. Online documentation that allows for local annotations, synonyms, alternate phrasing, or translations has enhanced value. Additional desirable services include support for bookmarking and automatic history keeping, which allows backtracking. Designers will be most effective when they design online documentation to fit the electronic medium and take advantage of text highlighting, color, sound, animation, and string search with relevant feedback. In some cases, provisions may be needed to provide the documentation offline, if access is not allowed outside of internal networks. Researchers caution that words should be treated as valuable commodities and used sparingly in documentation, whether in printed form or online (Redish, 2012).

BOX 14.2

Advantages of online documentation.

- **Physical advantages:** Available whenever on web-connected electronic device, can't get lost or misplaced, physical workspace not needed, can be updated rapidly.
- **Navigation features:** Can provide index and other search facilities, can link to other external materials and sources.
- **Interactive services:** Can bookmark, annotate, and tag; can include graphics, sound, color, and animation; screen readers or other tools can be provided for users with disabilities.

14.3.2 Online help

Users typically seek out help in solving specific problems and will want to go directly to the information that is needed. The traditional approach with online documentation is to have users type in or select a help-menu item; the system then displays a list of alphabetically arranged topics that can be clicked on to read a paragraph or more of helpful information. This method can work, but it is often frustrating for those users who are not sure of the correct terms to use to search for information on the tasks they wish to accomplish. They may see several familiar terms (search, query, select, browse, find, reveal, display, info, or view) but not know which one to select. Worse still, there may not be a single command that accomplishes the task, and there is usually little information about how to assemble actions to perform tasks (such as converting graphics into a different format). Online help that offers concise descriptions of the interface icons and actions (Fig. 14.4) is probably most effective for intermittent knowledgeable users; it is likely to be less useful for novices, who have more need for tutorial training.

Sometimes simple lists—for example, of *keyboard shortcuts*, *menu items*, *icon glossaries*, or *mouse shortcuts*—can provide the necessary information. Each item in the list might have an accompanying feature description. However, many designers recognize that such lists can be overwhelming and that users usually want guidance for accomplishing their specific intended tasks (for example, printing on envelopes).

The online help and support center found with most Microsoft products offers many ways of finding relevant articles, called *topics*. Users can browse an organized table of contents that lists the topics hierarchically or search the text of the articles. Finally, Microsoft's current approach allows users to

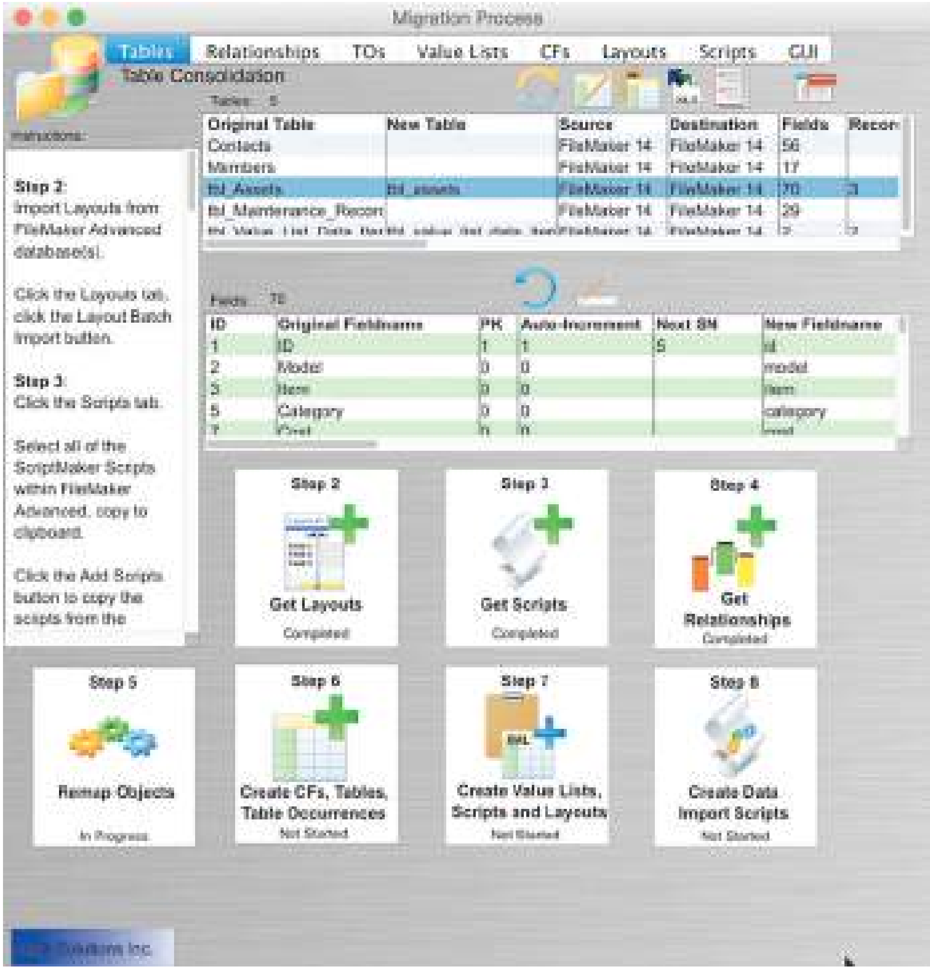


FIGURE 14.4

This figure provides a step-by-step approach to the activities in a database migration application. The details involved in each step can be found on the left-hand side. The actual action and activity are in the middle. On the bottom is a road map for the steps involved; the steps are marked as to their status (completed, in progress, not started). For novices, it would be easy to insert links to more detailed tutorial-type information (http://www.fmpromigrator.com).

type requests into a search box using natural language statements; the program then selects the relevant keywords and offers a list of topics organized into categories.

Interactive help The simplest way to take context into account is to monitor the cursor location and provide helpful information about the icon under the cursor. This form of user-controlled interactive help is readily understandable to users and even fun to use. This information should have three qualities: It should be available without interfering, succinct yet descriptive, and nonintrusive (Sherwin, 2015). Users position the cursor on a widget (or other visible interface icon) and then press a help key or hover the mouse over the icon for a couple of seconds to produce information about the icon on which the cursor is resting. In a common version of this technique, users simply move the cursor to the desired location and hover over the icon, causing a small pop-up box (often called a tool tip, ScreenTip, or balloon help) to appear with an explanation of that icon (Fig. 14.5). Alternatively, all of the balloons may be displayed at once so that users can see all of the explanations simultaneously (Fig. 14.6). Another approach is to dedicate a portion of the display to help, which is updated automatically as users hover over or select interface widgets or icons. Yet another approach is to use two monitors, where one monitor displays the user's activity and the other monitor displays the help information with videos and help pages (Matejka et al., 2011). User-controlled help can also be used for more complex items such as control panels or forms. These features provide a narrower window into the extensive volume of help that is available to the user. By having the users control the action, it is less disruptive to them. Designers can create a small overlay help window that allows users to adjust the size as well as allowing them to minimize, close, or move the window (Sherwin, 2015).

System-initiated help Another approach is to provide system-initiated help, often called "intelligent help," that tries to make use of the interaction history, a model of the user population, and a representation of its tasks to make assumptions about what users need. By keeping track of user actions, some researchers believe that they can provide effective system guidance, such as suggesting that users redefine their margins since they are indenting every line. Research in computer-based intelligent user interfaces still continues to have mixed results.

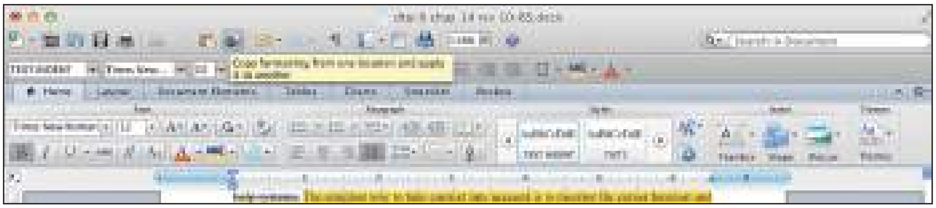


FIGURE 14.5

A small tool tip or fly-over help on Microsoft Word. Moving the mouse (cursor) over a particular icon causes an explanation of the icon to appear on the screen.

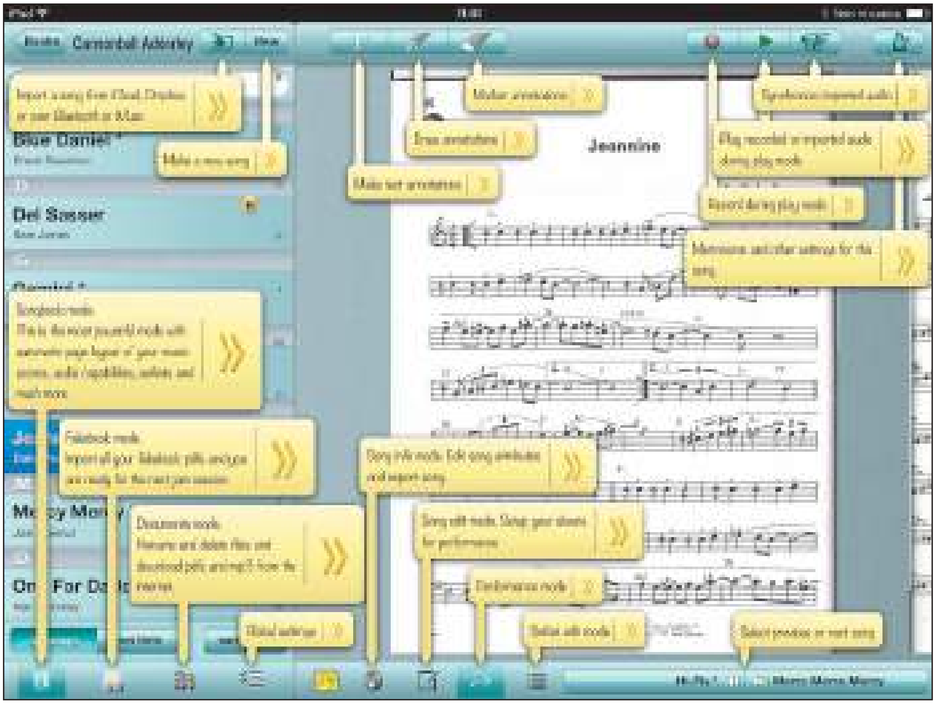


FIGURE 14.6
Since this is a complicated music interface, interactive help is provided for many of the fields. This figure shows all the help balloons open. Several standard icons are used such as new, import, record, and play, but other specific icons explain the various modes that are available while using the tool (http://www.saxopedia.com/2013/12/07/calypso-score-the-perfect-ipad-music-score-reader/).

14.4 Reading from Displays versus Reading from Paper

The technology of printing text on paper has been evolving for more than 500 years. The paper surface and color, the typeface, character width, letter sharpness, text contrast with the paper, width of the text column, size of margins, spacing between lines, and even room lighting all have been experimented with in efforts to produce the most appealing and readable format.

Visual fatigue and stress from reading computer displays are common problems, but these conditions respond well to rest, frequent breaks, and task diversity. Even if users are not aware of visual fatigue or stress, their capacity to work with displays may be below their capacity to work with paper documentation.

The interest in reading from displays has increased as mobile devices, tablets, specialized electronic book platforms, and web-based libraries have become more common. Being able to download a customized morning newspaper onto a pocket-sized electronic device to read while standing in a crowded subway or to carry a full city guide on such a device while touring are strong attractions and often expected by today's users (Fig. 14.7). If users are to read large amounts of material online, high-resolution and larger displays are recommended. Other studies recommend that quick response times, fast display rates, black text on a white background, and page-sized displays are important considerations if displayed text is meant to replace paper documents. Some applications provide a dedicated reading layout view that limits the number of controls and increases the space available for viewing the text. Dynamic resizing can take into account the display size to facilitate paging through the document instead of scrolling. *Sans serif* fonts (Fig. 14.8) should be used online, as they are crisper and produce a less cluttered appearance. Sans serif fonts are also preferred for low-vision readers. The term *sans serif* comes from the French and means "without serifs," where serifs are the ornamentations on each letter.

FIGURE 14.7

Users read their favorite news sites on a variety of media. Sometimes it may be their phone, other times their computer. The content needs to adjust (dynamic resizing) to the chosen medium. Note how the display on the laptop is in a landscape (horizontal) format, but the display on the table is portrait (vertical). On the phone, some of the material is not available on the first screen but may be available as users scroll or select a link to display further information. It is important that the format of the content is tested and reviewed on the different-sized devices.

FIGURE 14.8

In September 2015, Google changed its standard logo. Note the new logo (on the right) uses a sans serif font. One reason for this change is to make the logo easier to scale on small screens like mobile devices. Sans serif fonts appear cleaner and less cluttered and are easier to read on screens.

Large online libraries of books—whether they are made available for free, as by the Gutenberg archives or the Library of Congress; by library subscription; or for pay, a service offered by numerous publishers—promote efforts to improve the reading experience. Publishers of newspapers and scientific journals are evolving to satisfy the intense demand for online access to articles while struggling to ensure a way to recover their costs. Plasticity of documents (responsive web design) is becoming a requirement. The ability to automatically sense the correct orientation for reading a document has become a standard feature. Content designers have to structure their materials so that they can be read on small, medium, and large displays and at different font sizes to accommodate vision-impaired users. Designers need to be aware of their target audience, including accommodations for age and educational levels (Fig. 14.9).

FIGURE 14.9

The National Institutes of Health's site for seniors (http://www.nihseniorhealth.gov) has controls to adjust the text size and adjust the contrast. The font used is a sans serif font, and the font size is larger than the typical size used on the web (being aware that the site is targeted at older adults). Several ways are provided to navigate through the information (alphabetical, grouping by category, etc.). There are controls provided to resize the text as well as change the contrast.

14.4.1 Extended reading on displays

Although earlier displays did not always provide the best resolution for reading, the digital ink technology used on many of today's reading devices has changed that. Yet researchers are still trying to understand and see what issues, if any, exist from extended reading on displays.

The Pew Research Center (2014) states that 50% of Americans have a dedicated handheld device used for reading (Fig. 14.10). Another finding from its research is that those who read ebooks read more books in all formats. People like ebooks for their speedy access and portability. Print books are still preferred in certain situations, such as reading to children. Often people prefer print to digital for sustained periods of reading (Stone and Baker-Eveleth, 2013).

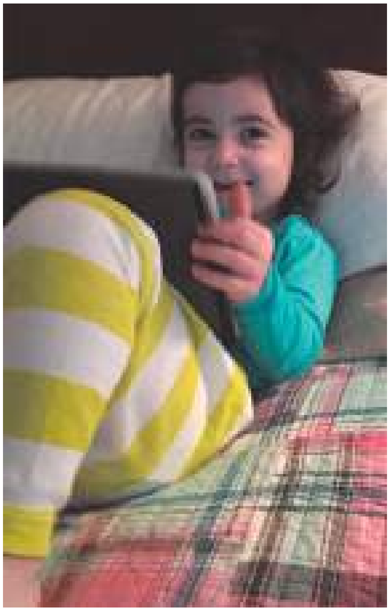


FIGURE 14.10

Although conventional books are relatively easy to transport and read in different places, reading an ebook has become very convenient as well (including reading in bed). The ebook devices are lightweight, the size of the characters can be easily adjusted, and today's technology has even removed much of the glare, so the ebook can even be read at a sunny location like the beach.

Johnson (2014) states that reading is an artificial skill that must be learned by instruction and practice. Designers should be aware how poor information design can make reading more difficult and disrupt the process. Some guidelines for supporting reading are:

- Don't use uncommon or unfamiliar vocabulary.
- Avoid difficult-to-read typefaces (ALL CAPS is harder to read).
- Avoid text on busy backgrounds.
- Avoid information buried in repetition.
- Don't use centered text.

In a study on reading comprehension performed with 10th graders in Norway comparing reading texts in print versus PDF files on a computer, it was found that reading from the texts in print produced better scores on reading comprehension. Reading in print gives users a better mental representation of where they are on the page, providing fixed spatial cues (Mangen et al., 2013). There are some concerns that the way users read online is different from how they read in print (see Section 14.2.2). Users often skim information on the web. In Western cultures, users read from left to right, and the scanning across each line gets shorter as users move down the page (Fig. 14.2).

No one yet has answers about what will happen to people as they continue to receive such a large volume of information from so many different sources. When Maryanne Wolf initially published her work in 2007, the electronic tsunami had just begun. It seems that as more reading content moves online, studies seem to show a decrease in reading comprehension. Just like we listen to "sound bytes," we now read with "eye bytes." Studies seem to show readers are losing some of their deep reading skills (Wolf as cited in Konnikova, 2014). Other studies are under way such as Chen and Chen (2014) that show how a collaborative reading annotation system can enhance digital reading. Baron (2015) adds that technology is reshaping what it means to read, and with that we need new innovative devices that support reading in the online environment (Fig. 14.11).

FIGURE 14.11

New ebook devices will be created that support the metaphor from physical books, allowing the user to flip the pages to scan the contents of the book. In this photo the user reveals a page that has previously been "bookmarked" by the simple act of resting his thumb on the margin of the page. But the tipping alone does not actually navigate back to the previous page in and of itself. The user has to confirms the action by tapping the thumb against the edge of the device, so as to actually navigate back to the location of the bookmark. This serves to reveal the interaction, while also ensuring that it is an overt act that is not likely to be triggered by accident when one simply rests their hands on the device (Yoon, et al. 2015).

14.5 Online Tutorials and Animated Demonstrations

An online tutorial is an interactive training environment in which users can view explanatory descriptions of user-interface icons and actions, often tied to realistic task scenarios. There are many approaches to the use of electronic media to teach users how to master an interface. Depending on the complexity of the interface and the amount of time users are ready to spend absorbing the tutorial materials, they might be served well by an extensive computer-based training module, an animated demonstration of features, or a recorded welcome message by a familiar person. The challenge often is to prepare materials that will satisfy users who want a three-minute introduction as well as those who need a one-hour in-depth treatment. This section reviews a range of online possibilities, from textual and graphical tutorials to fully animated demonstrations.

A more ambitious approach to training is based on a complex model of learning patterns tied to carefully designed educational tutorials that guide users and correct their mistakes. These have demonstrated impressive outcomes, but the success stories are based on years of development, testing, and refinement. The successful designs provide clear challenges, helpful tools, and excellent feedback. This can be contrasted with the many YouTube tutorials available, many by do-it-yourselfers with minimum exposure to any learning principles and pedagogy.

14.5.1 Online tutorials

One introductory tutorial for the Adobe Photoshop® package displays the exact steps users must make and then shows the actions being carried out using a recorded demonstration. Users just keep pressing the space bar key to speed through the demonstration. Some users find this guided approach attractive; others are put off by the restrictive sequencing that prevents errors and exploration. Automated tutorials can be created using Autodemo® and Show Me How Videos™. Autodemo has arrangements with several worldwide companies and provides specific instructions on navigating the various websites. Additional research into online tutorials is ongoing, with excellent contributions from Autodesk™. Four emerging directions are gamification, community input into tutorials, leveraging transfer effects, and distribution of practice (Cockburn et al., 2014).

The opportunity for carrying out *practice tasks* during online tutorials is one of their greatest strengths. Getting users to be active is one of the key tenets of the minimal-manual approach, and it applies especially well to online tutorials.

Another attractive variant is the start-up tip: Each time users start the interface, they get a pop-up box displaying a brief explanation of a feature. Some systems monitor user behavior and show start-up tips only for features that are not used by this particular user. Of course, the user should always be given the option to turn off these tips at any time.

Creators of interactive tutorials must address the usual questions of instructional design and also the novelty of the computer environment. A library of common tasks for users to practice is a great help. Sample documents for word processors, slides for presentation software, and maps for geographic-information systems help users to experience the applications. Repeated testing and refinement are highly recommended for tutorials.

With the variety of tutorials available (from both high-quality companies and do-it-yourself selections), it is hard for the general user to assess the quality of the tutorial. There is limited information in the research literature about a set of quality metrics. A single rating does not seem to distinguish quality, so multiple criteria (learning, coolness, ease of following, enjoyment, writing style, error prediction, image helpfulness) need to be used (Lount and Bunt, 2014). Today tutorials are not just a single point of entry into learning the software. Most users will also go to the web and see what others say about the tutorial. Having a set of TaggedComments (Fig. 14.12), where users can tag their comments and pin them to particular locations in the tutorial, is helpful (Bunt et al., 2014). Another approach is to provide for community-enhanced tutorials, which continuously improve and evolve as more users work with them (Lafreniere et al., 2013).

Using some automated tools can improve the quality of tutorials. MixT automatically generates step-by-step tutorials from user demonstrations and actions (Chi et al., 2012). Another product is DemoCut, a video-editing system that improves the quality of amateur instructional videos for physical tasks (Chi et al., 2013). EverTutor provides simplified tutorial creation by generating interactive tutorials on smartphones based on user demonstrations (Wang et al., 2014). Designers need work to continue to improve the quality of tutorials (Wakkary et al., 2015).

14.5.2 Animated demonstrations and multimedia

Animated demonstrations have become a modern high-tech art form (see Section 12.4). Manufacturers originally designed them mostly to attract potential users of software or hardware by showing off system features using the best animations, color graphics, sound, and information presentation that advertising agencies could produce. Those demonstrations focused on building a positive product image. Demonstrations and videos have become a standard technique to train users as they work. These have been enhanced by the use of augmented reality (Mohr et al., 2015 and Section 7.6). The focus is

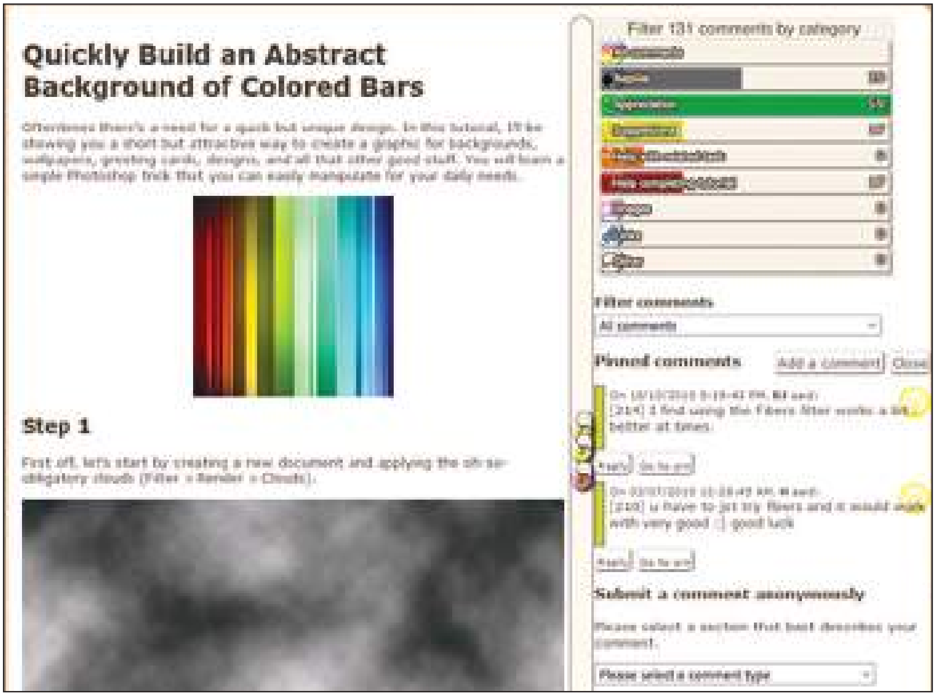


FIGURE 14.12

This is a specialized interface called TaggedComments. The comments section can be seen to the right of the images. In the upper right corner, the user can filter the comments by category. Color coding is used to distinguish between the categories. There is a place to *pin* a comment and a comment can be posted anonymously. (Bunt et al., 2014).

on demonstrating step-by-step procedures and explaining the results of the actions (Woolf, 2008). Automatic pacing and manual control satisfy hands-off and hands-on users, respectively. Use of standard playback controls allows users to stop, replay, or skip parts and add to their acceptability.

An animated demonstration can be prepared as a slide show, a screen-capture animation, or a video recording of a person using the device. A slide show might be appropriate for form fill-in or menu-based interfaces, but animation is preferable to demonstrate direct-manipulation interactions such as drag-and-drop operations, zoom boxes, or gestures. A screen-capture animation is easy to produce with standard tools such as Camtasia Studio® and Flash. These recordings can then be saved, possibly annotated or narrated, and replayed automatically by users. In our own explorations, we found that users appreciated recorded voice explanations, which make the demonstrations livelier and lead to more compact demonstrations; however, providing scripts and subtitles is necessary to address the needs of users with disabilities. Also, a video of a person using the interface can help clarify how special hardware is to be used—for example, to demonstrate the two-handed operation of a drawing system or the unfolding of a telephone keyboard accessory.

Novice users are sometimes overwhelmed by the complexity of today's interfaces. Supplying excellent documentation can aid their understanding, but sometimes the interface needs to be simplified. Using a *multi-layered interface design* that can unfold and further challenge and encourage users as they become more accomplished is a good approach. Experts can jump to the deeper layers quickly, but novices initially encounter a simplified interface (Hwang and Yu, 2011). Computer-game designers deserve credit for advancing the art of the animated demonstration, with lively introductions and introductory trailers that show samples of how the games are played. Demonstrations and previews have to explain the game and make it seem appealing and challenging, all within 30 seconds. Gamification including competition can be used in designing tutorials (Li et al., 2014).

Autodesk and Adobe both produce products that can create animations. There are other products as well including Cinema 4D, Toon Boom Studio, and Blender. The market is often changing with some products disappearing and new ones appearing. Specific uses often control the best choice.

14.6 Online Communities and Other Avenues for User Support

Instead of natural language conversations with computers to get help, interaction with other people online is proving to be effective and popular. This communal approach may employ e-mail, chat, or instant messaging for question asking and responses (Novick et al., 2007). Questions can be sent to a designated help desk or staff member or posted on a discussion board. Responses can be received in seconds or, more typically, minutes or hours. Applications (e.g., Oracle's RightNow) exist that can aggregate information from e-mails, chat, texts, user forums, and so on, into information that can be shared with customers to resolve product issues and answer product questions, letting users solve their problems with the help of other users.

The existence and proliferation of online communities (see Chapter 11) are increasingly appealing because of the minimized cost to software-maintenance organizations and help-desk staff. Many respondents get a sense of satisfaction from being able to help others and demonstrate their abilities. Some are motivated to achieve prominence within a community in the hope of gaining consulting contracts. Of course, the downside of broadcasting appeals for assistance is that users must publicly expose their lack of knowledge and risk getting incorrect advice. The upside is that a specific problematic case may be solved by an expert user.

Many websites now provide e-mail contact information or chat facilities (see Chapter 5) as opposed to written addresses or phone numbers. Also, to prevent basic questions from tying up staff resources, managers of help desks often record common questions and answers into files of FAQs. This enables newcomers to browse typical problems that have been discussed in the past. These files are often searchable and organized by type of issue or some other hierarchical scheme. Other alternatives for advice can be personal sharing activities. (Fig. 14.13).

Today, going to the Web for information of all types is considered standard practice. The simple interface of the ubiquitous Google search box is commonly used. The online choices out there are many and varied, but users need to be aware that not all information is correct and valid (see Chapter 15).

Although companies may provide a host of online facilities for information and advice, people are still very comfortable with asking the office "guru" for help (Novick and Ward, 2006b). Human-to-human communication removes some of the barriers found with more traditional documentation: Lack of understanding

FIGURE 14.13

People like face-to-face help as evidenced by frequent occasions when you find several users working together on a shared device.

can be handled quickly and without consequences of errors, and the human-to-human interface offers interactivity and other cues that enhance understanding.

14.7 The Development Process

Recognizing the difference between good and bad user documentation, regardless of format, is necessary for producing successful documentation on time and within a reasonable budget. Production of any documentation, like any project, must be managed properly, handled by suitable personnel, and monitored with appropriate milestones (see Box 14.3).

Getting started early is invaluable. Including the technical writers in creating the product functional specifications provides several benefits including the following: early involvement of people who represent the user, better written functional specification, and reuse of the functional specification content into product documentation. If the documentation-writing process begins before the interface is built, there also will be adequate time for review, testing, and refinement. Furthermore, the user documentation can act as a more complete and comprehensible alternative to the formal specification for the software. Implementers may miss or misunderstand some of the design requirements when reading a formal specification; well-written user documentation and other supporting material may clarify the design. The content writer becomes an effective critic, reviewer, and question asker who can stimulate the implementation team. In the time before the software is completed, the documentation may be the best way to convey the designers' intentions to potential customers and users as well as to implementers and project managers. Over the course of development, a

BOX 14.3

Development process guidelines.

- Seek professional content writers and copywriters.
- Prepare user documentation early (before implementation).
- Set up guidelines documents and coordinate and integrate across all involved departments.
- Review drafts thoroughly.
- Field-test early editions.
- Provide feedback mechanisms for readers.
- Revise to reflect changes regularly.

large number of documents may be generated. It may be worthwhile to assign a document specialist to a large project (van Loggem, 2013).

Informal walkthroughs with users are usually an enlightening experience for software designers and writers. Potential users are asked to read through the documentation and to describe aloud what they are seeing and learning as well as what they think might be missing. Field trials with moderate numbers of users constitute a further process for identifying problems with the user documentation and the software. Field trials can range in scope from half an hour with a half-dozen people to several months with thousands of users. One effective and simple strategy is for field-trial users to mark up the documentation while they are using it, allowing them to rapidly indicate typos, misleading information, and confusing sections. The use of collaborative review tools also provides a history trail and encourages all stakeholders to participate in the review process.

Software and its accompanying documentation are rarely truly completed. Rather, they undergo a continuous process of evolutionary refinement. Each version eliminates known errors, adds refinements, and extends the functionality. If the users can communicate with the writers, there is a greater chance of rapid improvement. When possible, keeping logs of the use of help materials and help-desk calls will determine which parts of the system need modification.

Quite often, this development work is performed by different groups of people who may be geographically dispersed in the same company, or some of the work may be "contracted out." However, it is important that the user sees a smooth and integrated view. This means attention to common colors, logos, terminology, and style. Standardized guidelines must be established and adhered to (Section 3.2), and the documentation, the associated software, and all the packaging must represent an integrated system. It is important to know the target audience for the documentation. For example, content writers would want to know if they were writing documentation for developers or for the users. Knowing if the user has any special needs that should be considered is important. This can include age, reading level, language proficiency, cross-cultural issues, or any other special characteristics. It is important while developing any documentation to be aware of lessons that can be learned from usability testing (Redish, 2010 and Chapter 5).

Practitioner's Summary

Sufficient personnel, money, and time should be allocated to producing support materials, whether they be online or on paper. Excellent content is crucial to customer success, and organizations should consider moving this effort from a cost-center allocation to part of the development activities. Documentation and online help should be developed before the implementation to help the development team define the interface and allow adequate time for testing. All documentation and online help should be tailored to the specific user communities where possible. Instructional examples should be realistic, encourage active exploration with exercises, use consistent terminology, and support error recognition and recovery. Animated demonstrations should be used when possible. Online guidance can lend a human touch if it contains supporting information by real users. Social media activities (including crowd-sourcing activities) through newsgroups, listservs, online communities, e-mail, chat, blogs, and instant messaging provide powerful low-cost support and easy access mechanisms and are widely accepted. Where possible, find a content or documentation specialist to create the documentation.

Researcher's Agenda

The main advantage of online materials is the potential for rapid retrieval and traversal, but little is known about how to offer this advantage conveniently without overwhelming novice users. Cognitive models of how animated, integrated demos facilitate learning require better understanding to guide designers. Users' navigation of online help systems should be recorded and studied to gain a better understanding of what characterizes effective help strategies. Better strategies for integrating help directly in the user interface are needed. Multi-layered designs in which users can select their level of expertise seem helpful, but further testing and refinement are necessary. Better understanding of reading patterns using electronic documents is also needed, as are further research and understanding regarding special populations and their specific design requirements.

Most of today's documentation advice is based on research from early interactions with computers. Computers did not have sophisticated interfaces, and today they are used by a variety of users with differing backgrounds (including some non-technical). Developing modern documentation guidelines for today's software is needed. Software today is not always used in a prescriptive sequential method, creating the need for innovative approaches to software documentation development. Metrics to rate the quality of the documentation are needed. This is a qualitative change moving from micro-HCI to macro-HCI designs. More work is needed to design new reading devices that work with and assist our cognitive resources so that reading does not become a shallow activity. Attention also needs to be given to low-literacy users.

WORLD WIDE WEB RESOURCES

www.pearsonglobaleditions.com/shneiderman

- ACM SIGDOC: http://sigdoc.acm.org/
- Communication Design Quarterly: http://writethedocs.org
  http://www.agilemodeling.com/essays/agileDocumentationBestPractices.htm
  http://www.smashingmagazine.com/2012/07/writing-effective-wordpress-documentation/
- IEEE Professional Communication Society: http://pcs.ieee.org/
- Society for Technical Communication: http://www.stc.org/
- Usability.gov: http://www.usability.gov/how-to-and-tools/methods/writing-for-the-web.html
- Shirley Kaiser: http://websitetips.com/webcontent/
- Leslie O'Flahavan: http://ewriteonline.com/

Discussion Questions

1. Your documentation team has developed an online help support for instant messaging client. You are hiring a usability testing firm to evaluate the online help and recommend improvements. Prepare a contract that specifies how you want the usability test to be performed and what reports you would like to receive. Your schedule gives them one week to prepare the study, one week to run it, plus one week to write up the final report. In your contract, specify the details of your requirements, including such information as the number of subjects required, test plans, and the types of reports.
2. What are the advantages of online documentation?
3. Defend the use of online communities for user assistance. Ensure your argument is well-rounded by pointing out both the weaknesses and strengths.
4. Describe the role user documentation and online help play in the lifecycle of a piece of software. Decide when these help materials should be created in the software design cycle.
5. What is meant by the "inverted pyramid" style of writing? Why is this style used in writing for the web?

References

Baker, Mark, *Every Page Is Page One: Topic-Based Writing for Technical Communication and the Web*, XML Press (2013).

Baron, Naomi, *Words on Screen: The Fate of Reading in a Digital World*, Oxford University Press (2015).

Bernstein, Michael S., Little, Greg, Miller, Robert C., Hartmann, Björn, Ackerman, Mark S., Karger, David R., Crowell, David, and Panovich, Katrina, Soylent: A word processor with a crowd inside, *Communications of the ACM 58*, 8 (2015), 85–94.

Bunt, Andrea, Dubois, Patrick, Lafreniere, Ben, Terry, Michael, and Cormack, David, TaggedComments: Promoting and integrating user comments in online application tutorials, *Proceedings of the ACM Conference on Human Factors in Computing Systems*, ACM Press (2014), 4037–4046.

Chen, Chih-Ming, and Chen, Fang-Ya, Enhancing digital reading performance with a collaborative reading annotation system, *Computers & Education 77* (2014), 67–81.

Chi, Pei-Yu (Peggy), Liu, Joyce, Linder, Jason, Dontcheva, Mira, Li, Wilmot, and Hartmann, Björn, DemoCut: Generating concise instructional videos for physical demonstrations, *Proceedings of the Symposium on User Interface Software & Technology (UIST)*, ACM Press (2013), 141–150.

Chi, Pei-Yu (Peggy), Ahn, Sally, Ren, Amanda, Dontcheva, Mira, Li, Wilmot, and Hartmann, Björn, MixT: Automatic generation of step-by-step mixed media tutorials, *Proceedings of the Symposium on User Interface Software & Technology (UIST)*, ACM Press (2012), 93–102.

Cockburn, Andy, Gutwin, Carl, Scarr, Joey, and Malacria, Sylvain, Supporting novice to expert transitions in user interfaces, *ACM Computing Surveys 47*, 2, article 31 (November 2014).

Earle, Ralph H., Rosso, Mark A., and Alexander, Kathryn, User preferences of software documentation genres, *Proceedings International Conference on Documentation*, ACM Press (2015).

Frampton, Beth, Use as directed: Developing effective operations and maintenance manuals, *Intercom*, STC (June 2008), 6–9.

Gentle, Anne, *Conversation and Community: The Social Web for Documentation*, 2nd Edition, XML Press (2012).

Hackos, JoAnn, Changing times, changing skills, *Communication Design Quarterly 3*, 2 (February 2015).

Hailey, David, *ReaderCentric Writing for Digital Media: Theory and Practice*, Baywood Publishing Company (2015).

Handley, Ann, *Everybody Writes: Your Go-To Guide to Creating Ridiculously Good Content*, Wiley (2014).

Hwang, T. K. Philip, and Yu, Horng-Yi, Accommodating both expert users and novice users in one interface by utilizing multi-layer interface in complex function products, in Rau, P. L. P. (Editor), *Internationalization, Design, HCII, LNCS* 6777, Springer-Verlag (2011), 159–165.

Johnson, Jeff, *Designing with the Mind in Mind*, 2nd Edition, Morgan Kaufmann (2014).

Johnson-Eilola, Johndan, and Selber, Stuart A. (Editors), *Solving Problems in Technical Communication*, University of Chicago Press (2013).

Katz, Joel, *Designing Information: Human Factors and Common Sense in Information Design*, Wiley (2012).

Kohl, John R., *The Global English Style Guide: Writing Clear, Translatable Documentation for a Global Market*, SAS Institute (2008).

Konnikova, Maria, Being a better online reader, *The New Yorker* (July 16, 2014).

Lafreniere, Ben, Grossman, Tovi, and Fitzmaurice, George, Community enhanced tutorials: Improving tutorials with multiple demonstrations, *Proceedings of the ACM Conference on Human Factors in Computing Systems*, ACM Press (2013), 1779–1788.

Li, Wei, Grossman, Tovi, and Fitzmaurice, George, CADament: A gamified multiplayer software tutorial system, *Proceedings of the ACM Conference on Human Factors in Computing Systems*, ACM Press (2014).

Lount, Matthe, and Bunt, Andrea, Characterizing web-based tutorials: Exploring quality, community, and showcasing strategies, *Proceedings International Conference on Documentation*, ACM Press (2014).

Mangen, Anne, Walgermo, Bente R., and Brønnick, Kolbjørn, Reading linear texts on paper versus computer screen? Effects on reading comprehension, *International Journal of Education Research 58* (2013), 61–68.

Matejka, Justin, Grossman, Tovi, and Fitzmaurice, George, Ambient help, *Proceedings of the ACM Conference on Human Factors in Computing Systems*, ACM Press (2011), 2751–2760.

Mehlenbacher, Brad, Multidisciplinary and 21st century communication design, *Proceedings International Conference on Documentation*, ACM Press (2009), 59–65.

Mohr, Peter, Kerble, Bernhard, Donoser, Michael, Schmalstieg, Dieter, and Kalkofen, Denis, Retargeting technical documentation to augmented reality, *Proceedings of the ACM Conference on Human Factors in Computing Systems*, ACM Press (2015), 3337–3346.

Nielsen, Jakob, F-shaped pattern for reading web content, *Jakob Neilsen's Alertbox* (April 17, 2006). Available at http://www.useit.com/alertbox/reading_pattern.html.

Novick, David G., Elizalde, Edith, and Bean, Nathaniel, Toward a more accurate view of when and how people seek help with computer applications, *Proceedings International Conference on Documentation*, ACM Press, New York (2007), 95–102.

Novick, David G., and Ward, Karen, What users say they want in documentation, *Proceedings International Conference on Documentation*, ACM Press, New York (2006a), 84–91.

Novick, David G., and Ward, Karen, Why don't people read the manual, *Proceedings International Conference on Documentation*, ACM Press, New York (2006b), 11–18.

Pearson, Jennifer, Buchanan, George, and Thimbleby, Harold, *Designing for Digital Reading*, Synthesis Lectures on Information Concepts, Retrieval, and Services, #29, Morgan & Claypool (2014).

Pew Research Center, *E-Reading Rises as Device Ownership Jumps* (January 2014). Available at http://pewinternet.org/Reports/2014/E-Reading-Update.asp.

Pinker, Steven, *The Sense of Style: The Thinking Person's Guide to Writing in the 21st Century*, Viking/Penguin Group (2014).

Redish, J. C., Technical communication and usability: Intertwined strands and mutual influences, *IEEE Transactions on Professional Communication 53,* 3 (September 2010), 191–201.

Redish, Janice (Ginny), *Letting Go of the Words: Writing Web Content That Works*, 2nd Edition, Morgan Kaufmann (2012).

Sherwin, Katie, Pop-ups and adaptive help get a refresh, *Nielsen/Norman Group* (March 15, 2015). Available at http://www.nngroup.com/articles/pop-up-adaptive-help/.

Stone, Robert W., and Baker-Eveleth, Lori, Students' expectation, confirmation, and continuance intention to use electronic textbooks, *Computers in Human Behavior 29* (2013), 984–990.

Strunk, Jr., William, White, E. B., and Angell, Roger, *The Elements of Style*, 4th Edition, Allyn & Bacon, New York (2000).

Tebeaux, Elizabeth, and Dragga, Sam, *The Essentials of Technical Communication*, 3rd Edition, Oxford University Press (2014).

van Loggem, Brigit, User documentation: The Cinderella of information systems, in Rocha, A., et al. (Editors), *Advances in Information Systems and Technologies*, Springer-Verlag (2013), 167–177.

Wakkary, Ron, Schilling, Markus Lorenz, Dalton, Matthew A., Hauser, Sabrina, Desjardins, Audrey, Zhang, Xiao, and Lin, Henry W. J., Tutorial authorship and hybrid designers: The joy (and frustration) of DIY tutorials, *Proceedings of the ACM Conference on Human Factors in Computing Systems*, ACM Press (2015), 609–618.

Wang, Cheng-Yao, Chu, Wei-Chen, Chen, Hou-Ren, Hsu, Chun-Yen, and Chen, Mike Y., EverTutor: Automatically creating interactive guided tutorials on smartphones by user demonstration, *Proceedings of the ACM Conference on Human Factors in Computing Systems*, ACM Press (2014), 4027–4036.

Wolf, MaryAnne, *Proust and the Squid: The Story and Science of the Reading Brain*, Harper (2007).

Woolf, Beverly, *Building Intelligent Interactive Tutors: Student-Centered Strategies for Revolutionizing E-Learning*, Morgan Kaufmann, San Francisco, CA (2008).

Yoon, Dongwook, Hinckley, Ken, Benko, Hrvoje, Guimbretière, François, Irani, Pourang, Pahud, Michel, Gavriliu, Marcel, Sensing Tablet Grasp + Micro-mobility for Active Reading. *Proceedings of the 28th Annual ACM Symposium on User Interface Software & Technology* (2015), 477–487.

Zinsser, William, *On Writing Well, Thirtieth Anniversary Edition*, Harper Collins, New York (2006).

This page intentionally left blank


Advanced Search
Search
User interface
Keyword Anywhere (GKEY)
AND OR NOT
Keyword Anywhere (GKEY)
OR NOT
Anywhere (GKEY)
From

CHAPTER 15

Information Search

> The gods of the earth and sea
> Sought through nature to find this tree.
> But their search was all in vain:
> There grows one in the human brain.
>
> **William Blake**
> *Songs of Innocence and Experience*, 1789

CHAPTER OUTLINE

15.1 Introduction

The way users conduct searches has undergone dramatic changes over the past two decades (Hearst, 2009, 2011; Nudelman, 2011; Wilson, 2011; Russell-Rose and Tate, 2013). Searching vast archives is now feasible for a broad spectrum of users, ranging from children preparing school reports to researchers looking for up-to-date results or experts to consult. This chapter focuses mainly on web or database search of text and multimedia collections, but many principles described here also apply to searching within a document or an address book.

The terminology swirl in this domain is especially colorful. The older terms *information retrieval* (often applied to bibliographic and textual document systems) and *database management* (often applied to more structured relational database systems with orderly attributes and sort keys) are being pushed aside by newer notions of *information seeking*, *filtering*, *collaborative filtering*, *sensemaking*, and *visual analytics*. Users often alternate these several strategies for finding information, a behavior that has been called *berry picking*. Computer scientists now focus on the huge volumes of available data and talk about *data mining* or *deep learning*. Information seeking differs from re-finding something that has been seen before as users might prefer navigation over search (e.g., following links, navigating a file structure or history), especially when users do not remember specific keywords to use in a search (Chapter 8).

This chapter reviews interfaces appropriate for first-time or intermittent versus frequent computer users and also for task novices versus experts. Providing powerful search capabilities without overwhelming novice users remains a challenge, usually addressed by providing a simple search interface with a link to the advanced search interface (Figs. 15.1 and 15.2). The simple interface consists of a single field in which to enter terms and a button to start the search. As users gain experience with the interface, they may request additional features by adjusting control panels. Knowledgeable and frequent users may need a wide range of search tools with many options that allow them to compose, save, replay, and revise increasingly elaborate query plans as they continue their information gathering over hours or days.

See also:

Chapter 8, Fluid Navigation

Chapter 9, Expressive Human and Command Languages

Chapter 16, Data Visualization

Many figures showing search interfaces in other chapters

FIGURE 15.1

The home page of the U.S. Library of Congress Online Catalog (https://catalog.loc.gov) shows the simple search box prominently placed at the top of the page and provides alternative means of finding items of interest in the diverse collections. Advanced search interfaces are provided to accommodate experienced searchers.

Web search engines have greatly improved their performance by making use of statistical rankings, the information latent in the web's link structure, and user's interactions (Liu, 2009). Thanks to the redundancy of information on the web, results almost always return some relevant documents, and they allow users to find answers directly or by following links. For example, to find an expert on information retrieval, users might first find papers on that topic, leading to identifying a major journal publication, the editors of the journal, and their personal webpages. Database searches are widespread as the general public turns to the web to reserve travel packages, shop for groceries, search digital libraries of children's books, and more. Specialized databases also help lawyers find relevant court cases and scientists locate the scientific data they need. Continuing evaluation will lead to further progress (Schuth et al., 2015).

To facilitate discussion, a few terms need to be introduced. Objects of interest, such as movies to rent, airline flights to reserve, flowers or books to purchase, or reviews to read, are stored in structured relational databases, textual document libraries, or multimedia document libraries or mentioned in unstructured web documents. A *structured relational database* consists of *relations*, which have *items* (also called *records*), and each item has multiple *attributes* (often called *fields*), which each have *attribute values*.

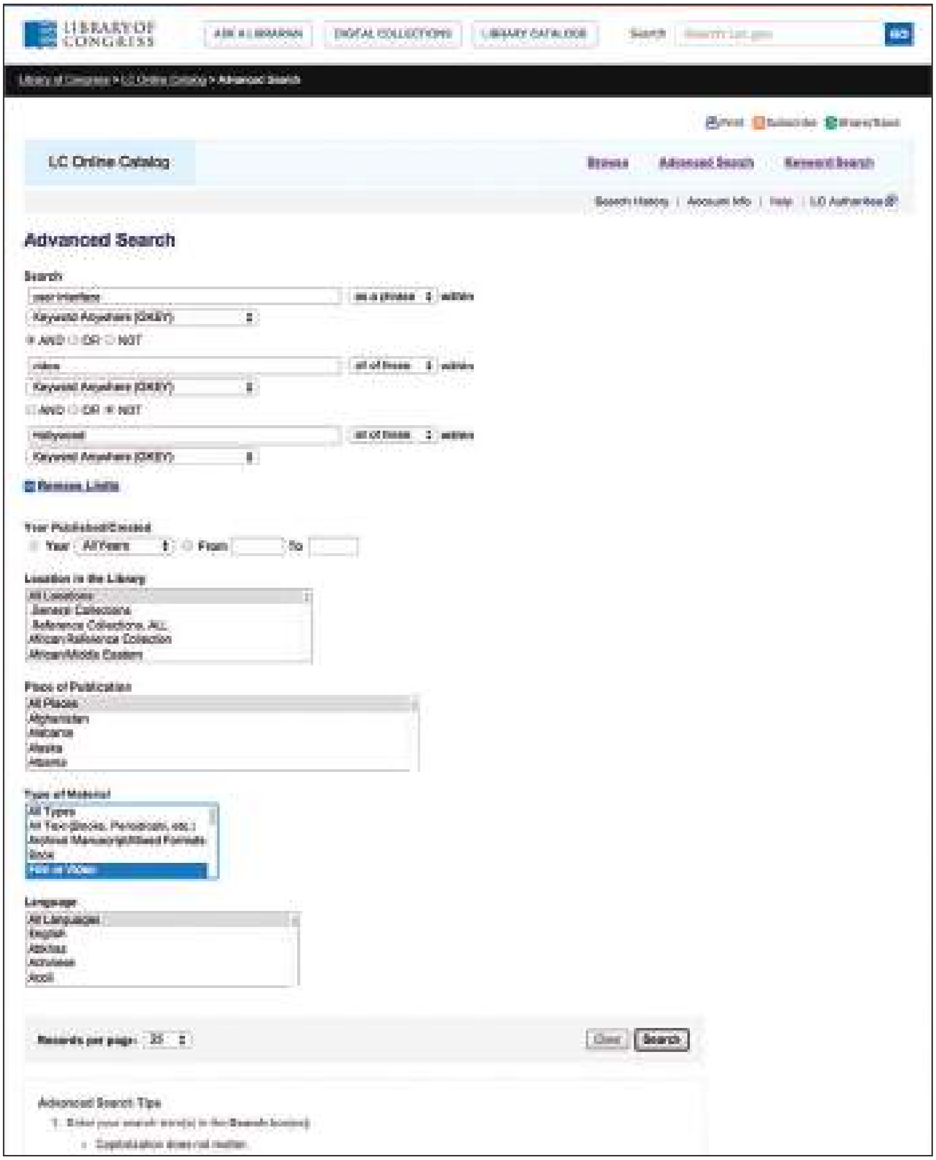


FIGURE 15.2

The advanced search interface of the U.S. Library of Congress Online Catalog (https://catalog.loc.gov). The entire page is now dedicated to search controls and tips. Using check boxes, text fields, and menus, users can compose Boolean queries, restrict the search scope to a subset of the collections, and apply filters based on metadata. Regular users sign up for an account to save results and keep a search history to facilitate re-finding.

A *library* consists of a set of *collections* (typically up to a few hundred collections per library) plus some *descriptive attributes* or *metadata* about the library (for example, name, location, owner). Each collection also has a name and some other descriptive attributes (for example, location, media type, curator, donor, dates, and geographic coverage) and a set of items (typically 10 to 100,000 items per collection). Items in a collection may vary greatly, but usually a superset of attributes exists that covers all the items. Attributes may be blank, have single values, have multiple values, or be lengthy texts. A *multimedia library* consists of collections of items that can contain images, scanned documents, sound, video, animations, datasets, and so on. A collection is typically owned by a single library, and an item typically belongs to a single collection. *Digital libraries* are generally sets of carefully selected and cataloged collections, while *digital archives* tend to be more loosely organized. *Directories* hold metadata about the items in a library and point users to the appropriate locations (for example, the NASA Global Change Master Directory helps scientists locate datasets in NASA's many archives). Items in *unstructured collections* (like the web) have no or very few attributes. These may include only format or date created. Tools are appearing that extract features automatically, but as this dynamically created metadata becomes available to interface designers, accuracy is often an issue.

Tasks can be decomposed into *browsing* or *searching* (Russell-Rose and Tate, 2013). Tasks can range from specific fact finding (or *known-item search*), where there is a single readily identifiable outcome, to more extended fact finding with uncertain but replicable outcomes. Relatively unstructured tasks include exploration of the availability of information on a topic, open-ended browsing of known collections, or complex analysis of problems and are also referred to as *exploratory searches*.

Here are some examples of tasks:

- **Specific fact finding (known-item search)**

  What are the houses available for sale near Annapolis, Maryland, with three bedrooms?
  How to change a flat tire?

- **Extended fact finding**

  What videos are by the author of the book *Peace Is Every Step*?
  How do Maryland and Virginia counties compare on the unemployment rate since the last election?

- **Exploration of availability**

  What genealogical information is available in the National Archives?
  Are there recent survey papers on voice recognition in the ACM digital library?

- **Open-ended browsing and problem analysis**

  Does the Mathew Brady Civil War photo collection show the role of women in that war?
  Are there promising new treatments for fibromyalgia that might help my patient?

Once users have clarified their information needs, the first step toward satisfying those needs is deciding where to search. The conversion of information needs (stated in task-domain terminology) to interface actions is a large cognitive step. Once this is done, users can express these actions in a query language or via a series of selections.

Supplemental *finding aids* such as description of the content of collections, tables of contents, and explanations of the indexes and subject classifications can help users pursue their information needs. Careful understanding of the task and of previous and potential future search requests helps provide hot-topic lists and useful classification schemes. The U.S. Congressional Research Service maintains a list of approximately 80 hot topics covering current bills before Congress and has 5,000 terms in its Legislative Indexing Vocabulary. The National Library of Medicine maintains the Medical Subject Headings (MeSH), with more than 27,000 items in a 12-level hierarchy, and the Gene Ontology Database has more than 15,000 genes organized in a 19-level hierarchy, with many genes appearing at multiple nodes.

Section 15.2 presents a five-stage search framework to guide search interface design with many examples. Section 15.3 focuses on dynamic queries and faceted search. Section 15.4 reviews search in multimedia documents and other specialized cases, and Section 15.5 covers the social aspects of search.

15.2 Five-Stage Search Framework

In designing the advanced search interface, a five-stage search framework may help to coordinate design practices and satisfy the needs of first-time, intermittent, and frequent users. The five stages of action, illustrated more fully in Box 15.1, are:

1. *Formulation:* Expressing the search
2. *Initiation of action:* Launching the search
3. *Review of results:* Reading messages and outcomes
4. *Refinement:* Formulating the next step
5. *Use:* Compiling or disseminating insight

Information seeking is an iterative process, so the five-stages can be repeated many times until users' needs are met. Users may not see all the components of the five stages, but if they are unsatisfied with the results, they should be able to have additional options and change their queries easily.

BOX 15.1

Five-stage framework to clarify search user interfaces.

1. **Formulation**
   - Use simple and advanced search.
   - Limit the search using structured fields such as year, media, or location.
   - Recognize *phrases* to allow entry of names, such as "George Washington."
   - Permit *variants* to allow relaxation of search constraints (e.g., phonetic variations).
   - Control the size of the initial result set.
   - Use scoping of source carefully.
   - Provide suggestions, hints, and common sources.
2. **Initiation of action**
   - Explicit actions are initiated by buttons with consistent labels (such as "Search").
   - Implicit actions are initiated by changes to a parameter and update results immediately.
   - Guide users to successful or past queries with auto-complete.
3. **Review of results**
   - Keep search terms and constraints visible.
   - Provide an overview of the results (e.g., total number).
   - Categorize results using metadata (by attribute value, topics, etc.).
   - Provide descriptive previews of each result item.
   - Highlight search terms in results.
   - Allow examination of selected items.
   - Provide visualizations when appropriate (e.g., maps or timelines).
   - Allow adjustment of the size of the result set and which fields are displayed.
   - Allow change of sequencing (alphabetical, chronological, relevance ranked, etc.).
4. **Refinement**
   - Guide users in progressive refinement with meaningful messages.
   - Make changing of search parameters convenient.
   - Provide related searches.
   - Provide suggestions for error correction (without forcing correction).
5. **Use**
   - Imbed actions in results when possible.
   - Allow queries, settings, and results to be saved, annotated, and sent to other applications.
   - Explore collecting explicit feedback (ratings, reviews, like, etc.).

15.2.1 Formulation

The formulation stage includes identifying the *source* of the information (i.e., where to search). Users may want to limit their searches for flights to certain travel sites, or limit a text search to help documents and not the entire web (Fig. 15.3), or search only women's clothing on a shopping website (e.g., women's shirts). Also called *scoping*, this limitation of the source can lead to better results but also lead to failures when the constraint remains active and users forget about it. Clearly displaying the source of information is important.

Even if it is technically feasible, searching all libraries or the entire web may not always be the best approach. Users often prefer to limit the search to a specific library or collection in a library (e.g., within the manuscript collections of the Library of Congress or within Wikipedia). *Keywords* or *phrases* can be specified, and structured *fields* such as year of publication, volume number, or language can be used to further limit the search scope. A text box and a few menus may be enough in most cases, but *form fill-in* (Section 8.6) allows users to specify more detailed searches in databases (e.g., search for nonstop flights between three local airports and New Orleans across a range of possible dates).

In database searches (e.g., Fig. 15.2), users often seek items that contain meaningful short phrases ("Civil War," "Environmental Protection Agency," "carbon monoxide"), and multiple-entry fields can be provided to allow for multiple phrases. Searches on phrases have proven to be more accurate than searches on individual words. Phrases also facilitate searching for names (for example, a search on "George Washington" should not turn up "George Bush" or "Washington, DC"). If Boolean operations, proximity restrictions, or other combining strategies

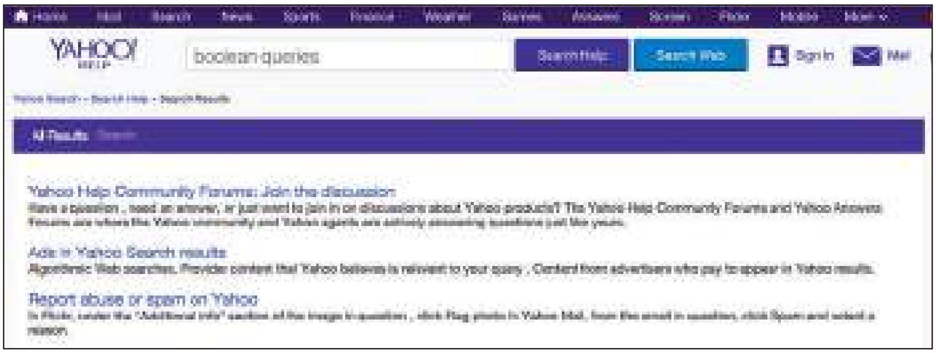


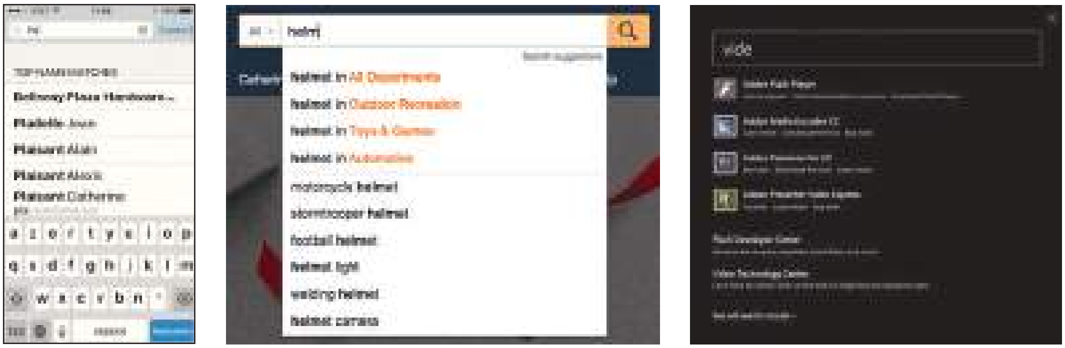
FIGURE 15.3

The Yahoo! help search box has two buttons of different colors to search two different sources of information: purple for searching the help information and blue for searching the web. Pressing the purple button "scopes" the results to the help information only and shows results below a purple banner. Searching the web jumps to a different page (the normal search) that reuses the blue button color, helping users keep track of which source of information they are searching.

are specifiable, users should also be able to express them. Users or service providers should additionally have control over *stop lists* (which typically filter out from the search terms common words, single letters, and obscenities).

When users are unsure of the exact value of the field (the terms to be searched for or the spelling or capitalization of a name), the search constraints can be relaxed by allowing variants to be accepted. In a textual-document search, advanced search interfaces may give users control over variant capitalization (case sensitivity), stemmed versions (the keyword “teach” retrieves words with variant suffixes such as “teacher,” “teaching,” and “teaches”), partial matches (the keyword “biology” retrieves “sociobiology” and “astrobiology”), phonetic variants from soundex methods (the keyword “Johnson” retrieves “Jonson,” “Jansen,” and “Johnsson”), synonyms (the keyword “cancer” retrieves “malignant neoplasm”), abbreviations (the keyword “IBM” retrieves “International Business Machines,” and vice versa), and broader or narrower terms from a thesaurus (the keyphrase “New England” retrieves “Vermont,” “Maine,” “Rhode Island,” “New Hampshire,” “Massachusetts,” and “Connecticut”).

When searching in a simple list of items (e.g., searching a name in a list of contacts), the result list can be displayed as users type. The list shrinks rapidly and users can select the wanted item without finishing typing the name. When the collection is large, *auto-completion* can be applied to the search term instead, revealing most common search phrases that match the text already typed (Fig. 15.4). The



(a) (b) (c)

FIGURE 15.4

Auto-complete suggestions can speed data entry and guide users toward successful queries.

(a) In a mobile phone address book, typing one character filters the list to all names that contain that character, and the list is updated continuously as users type.

(b) Typing “helm” in Amazon’s search box shows suggestions for “helmet light” or “welding helmet” but also suggestions to narrow the scope of the search to relevant departments.

(c) On the Adobe website, suggestions include products (e.g., typing the beginning of the word “video” suggests several video-editing tools).

auto-complete list is updated as users continue typing, which helps them recall terms of interest, limits misspelling, and speeds up the query initiation process.

Mobile applications may use context information such as location to narrow down the auto-completion suggestions. For example, searching on a map can narrow the history of previous searches to the ones relevant to the current location. This may allows users to find the exact location of a doctor's office for a follow-up visit even if they don't remember the name or exact address of the doctor—an impressive combined use of location, history and auto-completion.

For regular users who want additional control, advanced command languages can be offered to search databases (e.g., SQL; see Section 15.4). On the other hand, new users will benefit from the reading of typical phrases (which can be placed next to the search box), direct links to often searched items (e.g., sales or popular topics), and carefully designed tips. The seamless integration of search with navigation and browsing will allow users to switch to menus of choices when they are not able to come up with search phrases or to review sample materials to better understand what is available even before they start composing their query (see also Section 15.3 and Fig. 12.1 of NASA's Earthdata search interface).

15.2.2 Initiation of action

The second stage is the *initiation of action*, which may be explicit or implicit. Many systems have a search button for *explicit initiation*. A magnifier glass has become the de facto standard icon for search when space is limited, but pressing the Enter key on a keyboard or pausing during spoken interaction may be the only thing needed to initiate the search.

An appealing alternative is *implicit initiation*, in which each change to any component of the formulation stage immediately produces a new set of search results. *Dynamic queries* in which users adjust query widgets to produce continuous updates (Shneiderman, 1994) led the way in demonstrating the benefits of implicit initiation, and they have been widely adopted in *faceted browsing* (Section 15.3).

15.2.3 Review of results

The third stage is the *review of results*, in which users review results in textual lists (Fig. 15.5) or on geographical maps (Fig. 15.6), timelines (see HIPMUNK in Fig. 1.7), or other specialized visual overviews of results. If no items are found, that failure should be indicated clearly. When messages are worded carefully (see error messages in Section 12.8) and useful suggestions are provided, users are less likely to abandon their search (e.g., leave a shopping website to never come back).

When results are presented in a list, it is common practice to return only about 20 results, but larger initial sets are preferable for those with high bandwidth and large displays. Previews consisting of carefully selected text samples (or snippets; see Fig. 15.5), human-generated abstracts, photos, or automatically generated summaries help users select a subset of the results for use and can

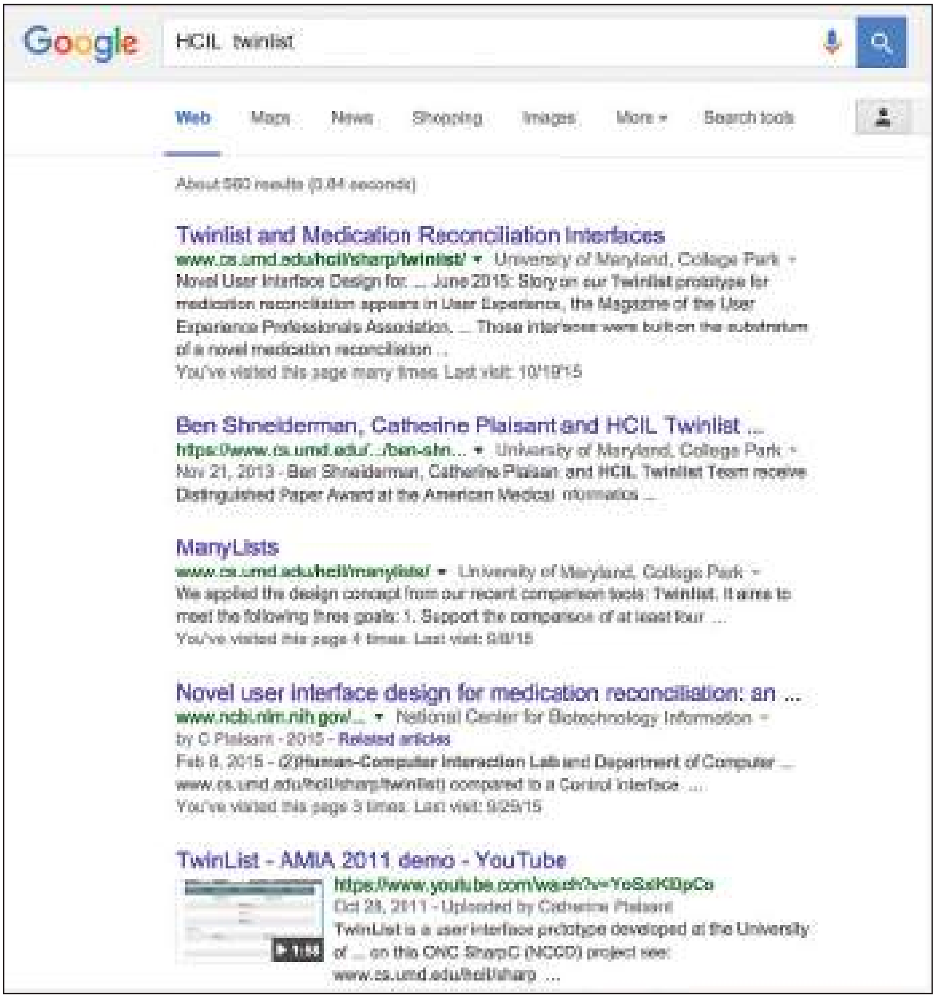


FIGURE 15.5

A Google Search result list. A summary is provided at the top (the total number of results). Each result includes preview information (or a snippet). Search terms are highlighted, including “Human-Computer Interaction Lab,” which is the expanded variant of the search term “HCIL.” The name of the top-level organization was added (here “National Center for Biotechnology Information”) to help users judge the trustiness of the information.

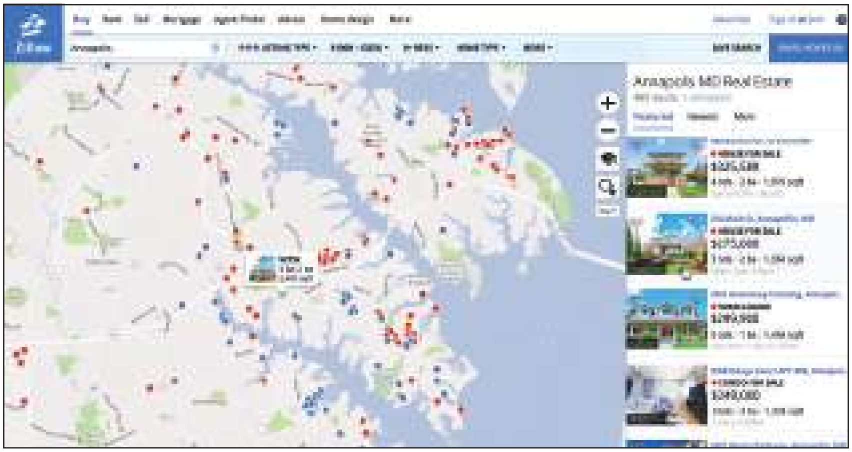


FIGURE 15.6

Searching for Annapolis on the real estate website Zillow returns a list of houses and dots displayed on a map. The two windows are coordinated; when the cursor hovers over a house in the result list, the location of the house is indicated on the map. A click on the house would bring all the details displayed in an overlapping window.

help them define more productive queries as they learn about the contents of the items (Greene et al., 2000). Translations may also be proposed. Allowing users to control how results are sequenced (e.g., alphabetical, chronological, relevance ranked, or by popularity) also contributes to more effective outcomes. If users have control over the result set size and which fields are displayed, they can better accommodate their information-seeking needs.

Highlighting the search terms in the snippet or other preview helps users gauge the value of results. Previous visits are noted. For websites, the URL is partially visible, and the name of the organization is provided—when available—helping users gauge the trustiness of the information. In database search, the preview information might indicate which collection the item belongs to or include a photo and important attributes (Fig. 15.6).

Additional *preview and overview surrogates* for items and collections can be created to facilitate browsing of results. Graphical overviews can indicate scope, size, or structure and help gauge the relevance of collections (e.g., with maps, timelines, or diagrams). Previews consisting of samples from collections entice users and help them define productive queries (Greene et al., 2000).

When the number of results is very large and metadata is available, a useful strategy is to provide an overview of the number of items in available categories (see also faceted search in Section 15.3). For example, when searching a library catalog, the number of books, journal articles, or news articles can be indicated (Fig. 15.7) and allow users to filter the results. When no metadata is available, one strategy is to automatically cluster the results based on content analysis (see, for

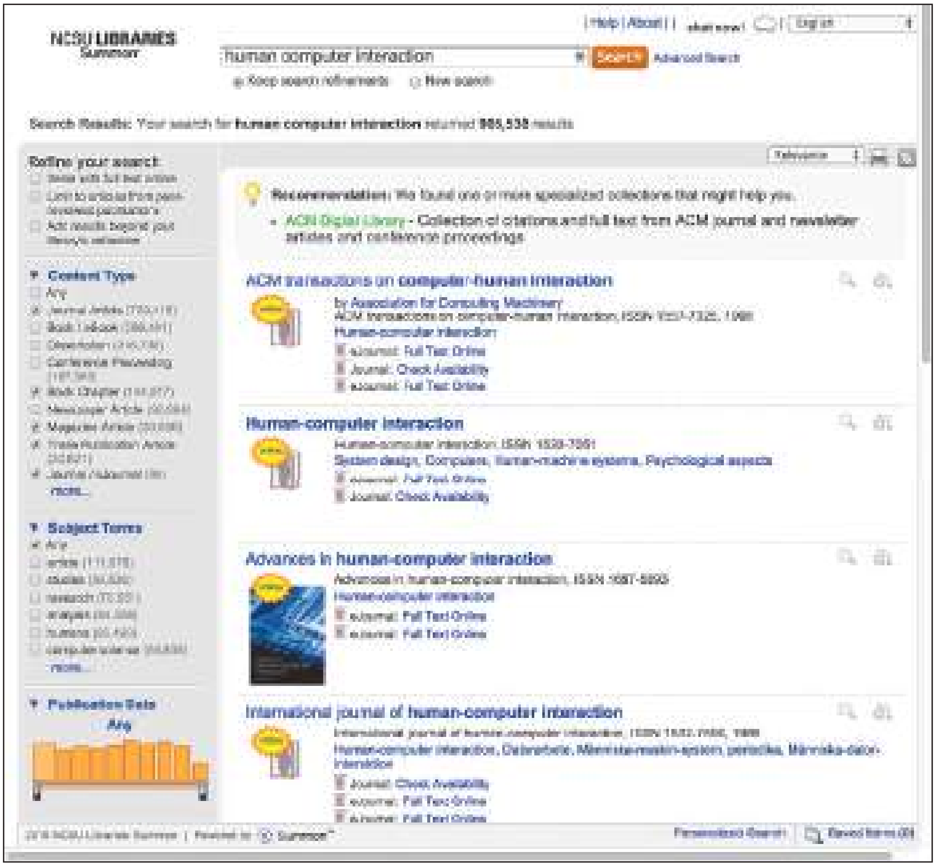


FIGURE 15.7

A search for "human computer interaction" powered by Summon™ for a university library catalog returns a very large number of results. On the left, users can see the number of results for categories organized by content type, subject terms, or publication date. The box provides an overview of the results, reveals how the search was done (e.g., here the default search does not returns dissertations), and facilitates further refinement of the search. The menu at the upper right allows users to sort results by relevance or by date. Help is available with a "Chat now" button, which allows users to chat with a librarian (http://www.lib.ncsu.edu).

example, Yippy at http://www.yippy.com). This allows users to navigate a tree of hierarchically organized topics, but the quality and appropriate labeling of the clusters are often problematic, so this technique is losing proponents.

To help users identify items of interest, access to the full document is usually necessary, with highlighting of the terms used in the search. For large documents, automatic scrolling to the first occurrence of the keyword is helpful, as are markers placed along the scrollbar to indicate the locations of other occurrences (see Fig. 9.8). A common issue when reviewing results within a document is to have the search box hide the location of the terms found in a document.

15.2.4 Refinement

The fourth stage is *refinement*. Search interfaces can provide meaningful messages to explain search outcomes and to support progressive refinement. For example, corrections can be proposed, such as asking, "Did you mean fibromyalgia?" when a term is misspelled. If multiple phrases were used, items containing all phrases should be shown first and identified, followed by items containing subsets. Progressive refinement, in which the results of a search are refined by changing the search parameters (e.g., search phrases but also time range, location, etc.), should be made convenient by leaving the search terms active—along with an easy way to clear them—instead of asking users to start from scratch every time.

15.2.5 Use

The final stage, *use* of the results, is where the payoff comes. Results may be merged and saved, disseminated by e-mail, or shared in social media. Users may want to feed the results to a bibliographic tool or be notified when new results become available. Sometimes direct answers or actions can be embedded directly in the result lists (Fig. 15.8), but most often search is only one of many components of a more complex analysis tool. For example, powerful environments are available for lawyers to review previous lawsuits and assemble supporting materials for their cases. Intelligence analysts might use tools such as nSPace from Uncharted Software (Fig. 15.9) to prepare evidence-based reports. Multiple searches can be specified at once, and names, dates, places, and organizations are

FIGURE 15.8

When possible (and important), provide information or simple actions without requiring users to leave the search results page. On the left, Google Search users get the answer to their safety-critical question at the top of the result list. On the right, Peapod shoppers looking for groceries can specify quantity and buy directly from the list of results after a search on "grapes."

FIGURE 15.9
nSpace TRIST from Uncharted Software is used by analysts to produce evidence-based reports. For this (fictitious) criminal investigation, a user is reviewing a collection of documents (shown as icons at the top). The search history on the left shows three searches. Names, places, and organizations have been automatically extracted. Here the search term “Laboratory” and the person “John Panni” are selected. Snippets are shown on the bottom left. Analysts use TRIST to define dimensions of interest and to quickly identify documents of interest for use in a Sandbox (Fig. 15.10) (http://uncharted.software/nspace; Chien et al., 2008).

extracted automatically. Analysts review documents and export information into a Sandbox (Fig. 15.10) to organize the evidence they found, mark it as supporting or refuting hypotheses, and then generate reports.

Many searches are related to searches done in the past. Users often need to find the same information again or to continue a search started the previous day. A search history, bookmarking, tagging, and indication of past visits in the results (e.g., “You have visited this page 2 times, Last visit 1/6/2016”) all contribute to helping users re-find information. Keeping the visited links visually distinct reminds users of where they have been.

Designers can apply the five-stage framework to make the search process more visible, comprehensible, and controllable by users. The five stages are often repeated many times until users needs are met.

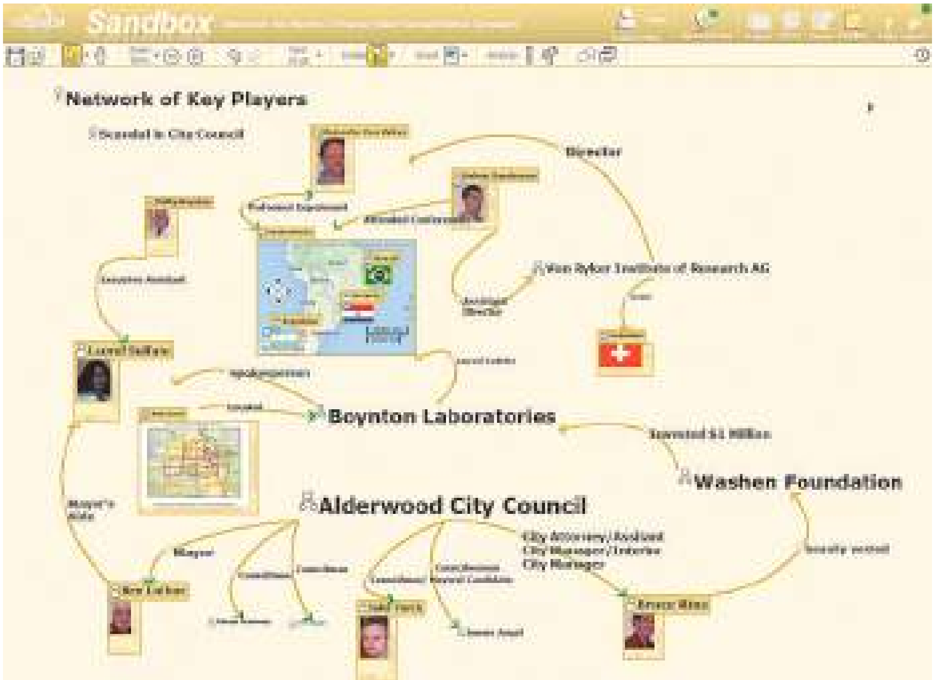


FIGURE 15.10

nSpace Sandbox® from Uncharted Software™ allows multiple analysts to organize and present the evidence gathered from research. A variety of tools such as node and link diagramming, automatic source attribution, recursive evidence marshaling, and timeline construction provide support for analysis and reporting.

15.3 Dynamic Queries and Faceted Search

While form fill-in interfaces can lead users to waste time specifying queries that have zero-hit or mega-hit result sets, early work on *dynamic queries* and *query previews* demonstrated the benefit of applying direct-manipulation principles to queries (Shneiderman, 1994; Greene et al., 2000). When metadata is available, dynamic query interfaces provide (1) a visual representation of the possible actions (e.g., menus, sliders, or buttons to represent choices for each field), (2) a visual representation of the objects being queried (e.g., a list of items, a map or any other visual overview), and (3) rapid, incremental, and reversible actions and immediate feedback. The dynamic query approach is appealing as it prevents errors and encourages exploration (Fig. 15.11). Early work on *query previews* used bar charts to show the distribution of attribute values for each field. It eliminated zero-hit queries as users could only select values leading to some

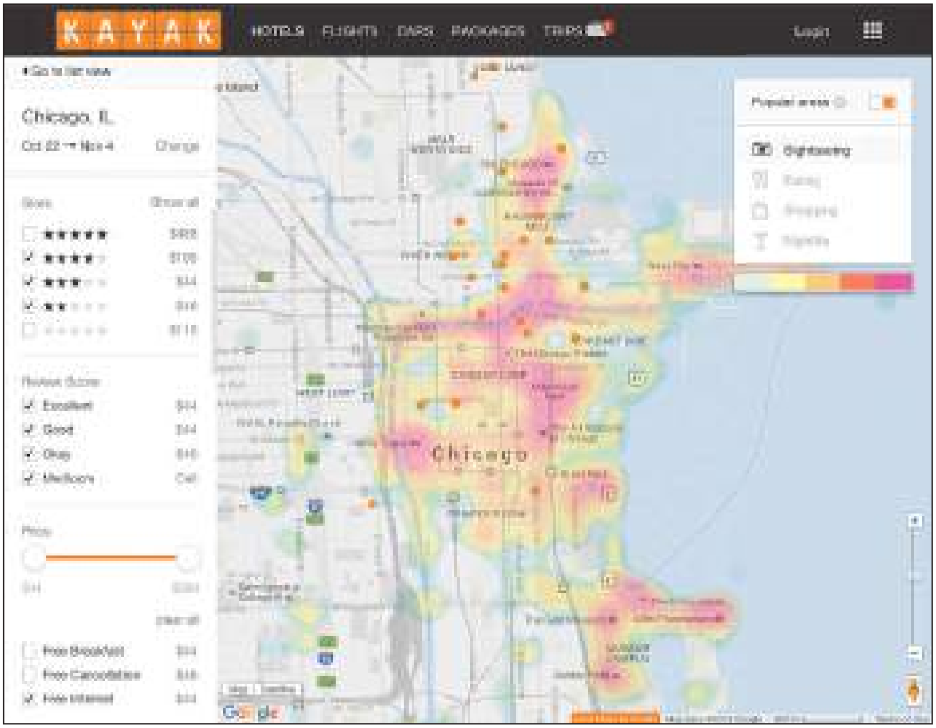


FIGURE 15.11

The hotel search interface of the Kayak travel website. After a form fill-in is used to provide the location (Chicago) and dates, results are displayed in a traditional list or on a map. The map provides an overview of the location of the hotels and can be zoomed to narrow the results. It is also augmented with a visualization of the popular sightseeing areas. On the left, menus are available to narrow down the categorical values, and sliders are provided for numerical values. Price is important, so the average price is provided for each category value.

results and also laid the foundation for faceted browsing (that typically uses the more compact numeric counts instead of bar charts to provide preview information about availability). Preview information does not necessarily need to be limited to the total count of items; for example, Fig. 15.12 shows an example where the preview is the average price, but users can still tell whether items are available.

The specification of fields' values can sometimes be simplified by using visual representations of the possible values. For example, selecting dates on calendars or using an airplane layout to select among available seats is useful. For house hunters seeking sale information near the city of Annapolis in Maryland, a location search box is needed to get started, but when a map of the area is displayed,

FIGURE 15.12

A preview of the price of available flights guides users to narrow down the time range for takeoff. The preview eliminates empty result sets and avoids high expenses (http://www.kayak.com).

it allows the selection of neighboring cities without having to know their names. Visual-search interfaces provide context and help users refine their needs by providing information about categorical or numerical values of attributes of the results. They are attractive and can reduce error messages such as "data out of range" while providing information about data availability and a feeling of thoroughness to users.

Faceted search was first demonstrated in the tool Flamenco (Yee et al., 2003.) It tightly integrates category browsing with keyword searching (i.e., navigation and search). Faceted search makes use of hierarchical faceted metadata presented as simultaneous menus (Section 8.2.4) and dynamically updated counts as a preview of the results. It allows users to navigate explicitly along multiple conceptual dimensions that describe the items and to progressively narrow or expand the scope of the query while browsing. It shows the structure as a starting point, organizes results in a recognizable structure, and gives control and flexibility over the order of metadata use and over when to navigate and when to search. Counts are updated dynamically, so empty result sets are avoided. Terms can be added to the search at any time. Items can belong to multiple categories, but all numerical values are binned into a small number of categories.

In the library example of Fig. 15.7, users can search for "human computer interaction" starting with books only and then expanding to books and dissertations or narrowing down to the most recent items. In the shopping example of Fig. 15.13, users can search for "REI tents" and review the results. They then can

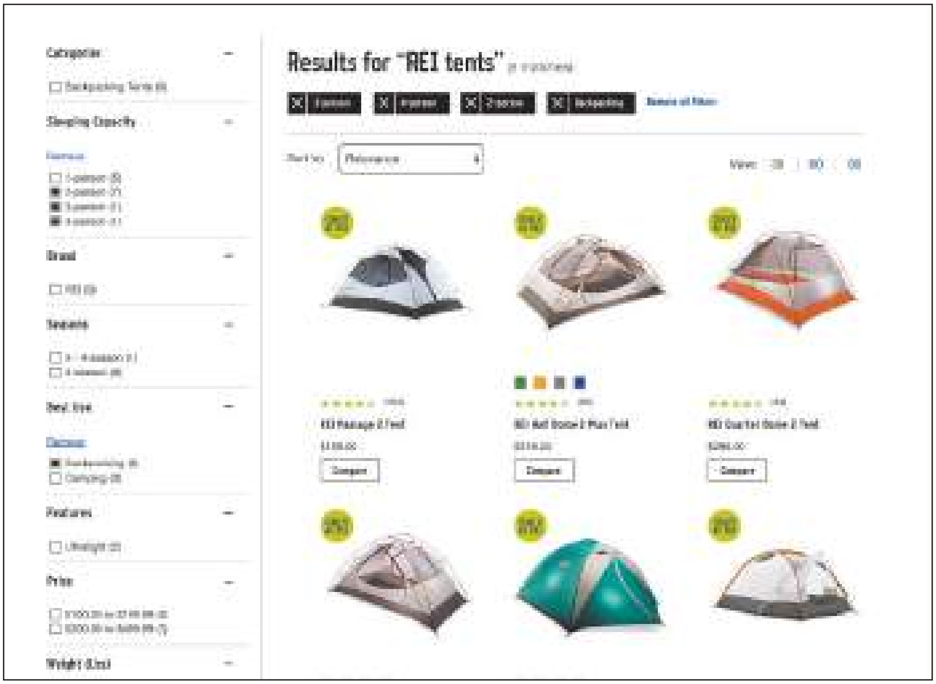


FIGURE 15.13

Faceted search interface of REI. Here users searched for “REI tents” and then browsed different tents by selecting values for multiple categories. The selected filters are clearly indicated at the top with a black background, making it easy for users to review the constraints and remove them.

narrow on tents for three people, narrow further by choosing backpacking, then widen the query to show three- and four-person tents because only one tent for three is available, and then switch brands, all the while staying in the flow and focusing their attention on the images of the tents.

In some interfaces, only a subset of the characteristics of faceted browsing is included. For example, some searches allow refinements in only one menu at a time (e.g., the International Children’s Digital Library, http://www.icdlbooks.org) as opposed to more than one menu simultaneously, which is possible with most shopping websites. Some interfaces reset all filters when the search terms change in order to adapt the categories to the items being returned. Even the counts might disappear when space is very limited (e.g., on small mobile devices). Fig. 15.14 illustrates how a mobile search interface can resemble faceted browsing by including a menu of filters that slides open and partially overlaps the result list. When users select a category value (e.g., condition being new), they can see the result list underneath being updated.

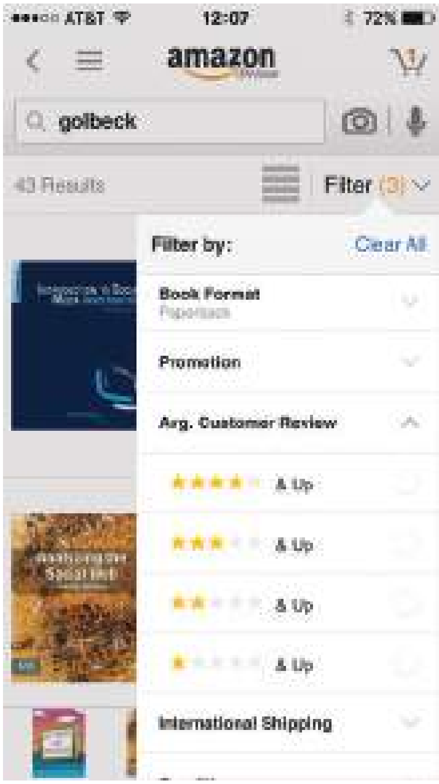


FIGURE 15.14

After searching for "Golbeck" in the Amazon app, users can scroll through the results or use the filter menu, which slides to the left and partially overlaps the result list. "Filter (3)" indicates that three filters have applied (e.g., "Paperback" as book format), reducing the results to 43.

The lack of consistency between search interfaces means that users have to discover how to search each time they use a different application. An analogy to the evolution of automobile user interfaces might clarify the need for standardization of search interfaces. Early competitors offered a profusion of controls, and each manufacturer had a distinct design. Some designs—such as having a brake pedal that was far from the gas pedal—were dangerous. Furthermore, if you were accustomed to driving a car with the brake to the left of the gas pedal and your neighbor's car had the reverse design, it might be risky to trade cars. It took half a century to achieve good design and appropriate consistency in automobiles; let's hope we can make the transition faster for search user interfaces.

15.4 Command Languages and “Natural” Language Queries

Form filling, dynamic queries, and faceted search allow users to specify fairly complex queries. A subset of Boolean queries is possible (ORs between attribute values and ANDs between attributes), but some users may want even more control over their queries. *Regular expressions* allow users to specify patterns of allowed variants (e.g., typing “*terror*” to return documents with “terrorist,” “terrorism,” or “anti-terrorism”) (see also Fig. 9.8). The *Structured Query Language* (SQL) remains a widespread standard for searching such structured relational database systems and often is the underlying query mechanism hidden under a more accessible front end. Using SQL, expert users can write queries that specify matches on attribute values, such as date of publication, language, or publisher. For example, an SQL-like command might be:

```
SELECT DOCUMENT#
FROM JOURNAL-DB
WHERE  (DATE >= 2004 AND DATE <= 2008)
   AND (LANGUAGE = ENGLISH OR FRENCH)
   AND (PUBLISHER = ASIST OR HFES OR ACM)
```

SQL has powerful features, but using it requires days of training to reach proficiency and even then users make frequent errors for many classes of queries.

Filtering with complex Boolean queries is made possible in commercial information-retrieval systems such as ProQuest Dialog™® and OCLC FirstSearch®. They permit complex Boolean expressions with parentheses, but their widespread adoption has been inhibited by their difficulty of use. Numerous proposals have been put forward to reduce the burden of specifying complex Boolean expressions, but a great part of the confusion stems from informal English usage. For example, a query such as “List all employees who live in New York and Boston” would typically result in an empty list because the “and” would be interpreted as an intersection; only employees who live in both cities would qualify! In English, “and” usually expands the options; in Boolean expressions, AND is used to narrow a set to the intersection of two others. Similarly, in the English expression “I’d like Russian or Italian salad dressing,” the “or” is exclusive, indicating that you want one or the other but not both; in Boolean expressions, however, an OR is inclusive and is used to expand a set.

Web search with “natural” language queries (for example, “How do I fix a flat?”) is appealing to users, particularly when using spoken interaction or searching the web (see more discussion of human language technology in Chapter 9). The computer’s capacity for understanding such queries is limited, but the availability of extremely large corpora enables search engines to find answers not by understanding the meaning of the question but by using query expansion, tracking large numbers of user interactions, and using statistical methods. Most often the semblance of a natural language query is achieved simply because other users have already asked the same question and the answers provided by humans can be retrieved (see more discussion in Section 9.4). For example, thousands of web users have asked, "How do I fix a flat?" and the answer can be located easily. Even if a user does not have the vocabulary to form the query adequately, it is likely that a long "natural" language query will provide useful answers. Hearst (2011) provides a nice example: "If a searcher needs a device to connect both a Wii and a DVD player to a TV but does not know what that device is called, a keyword query could fail. But the query 'How do I connect wii and dvd to my tv' turns up a nearly perfect match on a question-answering site, with the solution being a product called either 'video selector' or 'two-way A/V switcher.'"

Specialized corpora and systems restricted to narrow application domains can also lead to acceptable results. For example, searching in legal documents with a query such as "Find cases of tenants who have sued landlords unsuccessfully for lack of heat," the system can parse the text grammatically, provide synonyms from a thesaurus ("renters" for "tenants"), deal with singulars versus plurals, and handle other problems such as misspellings or foreign terms. Then the analyzer separates the query into standard components—such as plaintiff, defendant, and cause—and finds all meaningfully related legal citations.

See Chapter 9 for more discussion of human language technology.

15.5 Multimedia Document Search and Other Specialized Search

Interfaces to structured databases and text collections have greatly improved and allow users to achieve great feats, but search interfaces in multimedia-document collections are only starting to become more successful. To locate items such as images, videos, sound files, or animations, most systems depend primarily on text searches in descriptive documents or searches on keywords, tags, and metadata. For example, searches in photo libraries can easily be done by date, photographer, medium, location, or text in captions, but without captioning or tagging, finding a photo of a particular ribbon-cutting ceremony or a specific flower remains very difficult. Collaborative tagging of multimedia documents is dramatically changing how users search for photos, videos, maps, and webpages, but many important collections remain untagged. Automatically generated metadata is not as accurate as human-generated data but is often preferable to no metadata at all, as it can be useful to have computers perform initial filtering of results. Multimedia-document search interfaces that integrate powerful annotation and indexing tools, search algorithms to filter the collections, and media-specific browsing techniques for viewing the results lead to successful outcomes for users. Types of searches might include the following:

15.5.1 Image search

Finding images of things such as the Statue of Liberty might be simplified when they have already been tagged but very difficult to accomplish based solely on the pixels of the photo. Image-analysis researchers describe this task as query by image content, or QBIC (Datta et al., 2008; Heesch, 2008). Lady Liberty's distinctive profile might be identifiable if the orientation, lens focal length, and lighting were held constant, but the general problem is difficult in large and diverse collections of photos. Promising approaches are searching for distinctive local features, such as the torch or the seven spikes in the crown, or for distinctive textures or colors, such as red, white, and blue to locate an American flag. Of course, separating out the British, French, and other similarly colored flags is not simple, either.

More success is attainable with searches based on similarity, where users provide an image and retrieve items with similar features (e.g., Google Images) or even a sketch (e.g., Retrievr, http://labs.systemone.at/retrievr/). Results are often mixed, but errors might also help users broaden the scope of their search. When a photo matches closely a famous painting or a unique mosaic, the results may seem quite satisfying. Working with limited collections such as images of glass vases or blood cells leads to good results with specialized search algorithms. Even using screenshots of error message dialog boxes may be useful to find help (Yeh et al., 2011).

Photo tagging first appeared in commercial tools such as Adobe Photoshop Album but is now widely used in online tools such as Facebook, Flickr, and Google's Picasa™. Cameras now provide location, and automatic tagging of general image categories is becoming practical (e.g., Fig. 15.15), as is face recognition. The analysis of very large repositories of photos plus text data can also help make hypotheses about the content of photos (e.g., if thousands of people mention "Statue of Liberty" in social media or webpages next to very similar photos, it becomes possible to tag the photos with that phrase). Exploring how and why photos are shot, shared, and used can further aid tagging (Sandhaus, 2011). Overall, automatic tagging with human confirmation—and manual tagging when automation fails—can lead to successful applications. Rapid browsing of the results and smooth zooming are important.

15.5.2 Video search

Video content has increased dramatically, fueled by the ease of recording video with cellphones and the availability of video-sharing services such as YouTube and Vimeo. Many videos are short and have a narrow focus, so searching using the text of the title is often effective, but identifying videos that include objects,

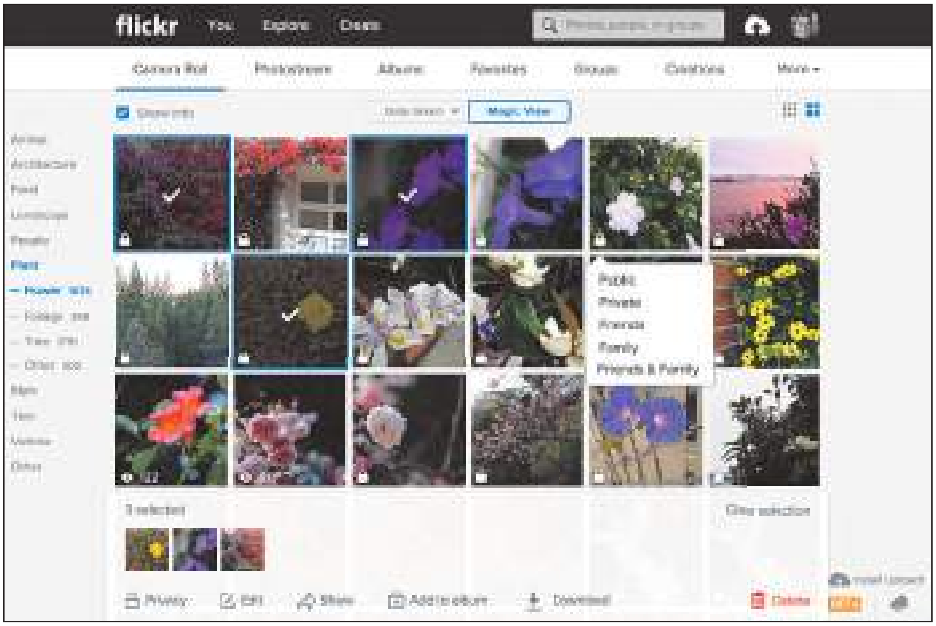


FIGURE 15.15

The "Magic View" of flickr automatically generates topic tags for each photo. Here users selected the photos with flowers. Three photos are selected and ready to be shared. The privacy setting is visible and can be changed with a menu.

actions, or events of interest and analyzing them remains a challenge (Snoek, 2008; Schoeffman, 2015). Video analysis builds on the advances in image analysis but adds the challenge of tracking a person or object between scenes and recognition based on information found in multiple frames. Analysis of the text in the scenes and speech-to-text transcripts help make large volumes of digital video more searchable. Finally, once results are available, automatic textual summaries describing the features found in the video might be useful (Xu et al., 2015), but most likely users will need to quickly review the video itself. Longer videos can be segmented into scenes or cuts to allow scene skipping. Researchers are exploring novel ways to browse result lists by video similarity as well (Fig. 15.16).

15.5.3 Audio search

Music-information retrieval systems can now use audio input, where users can query with musical content (Schedl, 2014). Users can sing or play a theme or hook from the desired piece of music, and the system returns the most similar items (e.g., with Shazam or Soundhound; Fig. 8.7). It is becoming possible to recognize individual performers, such as "find Madonna." Finding a spoken word or phrase in databases of telephone conversations is still difficult, but it is becoming possible, and speaker identification (also called voice biometrics) is

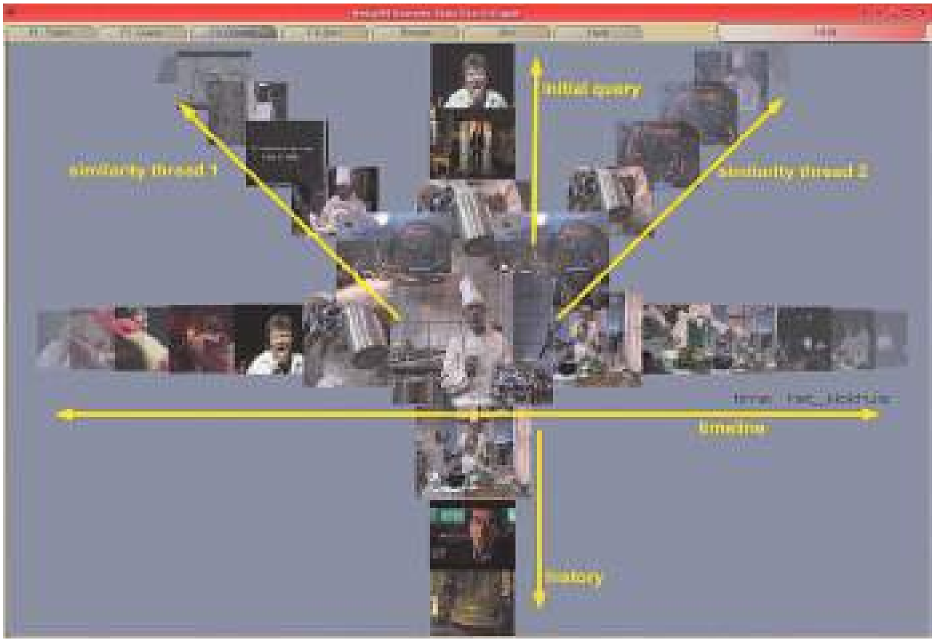


FIGURE 15.16

The ForkBrowser of the MediaMill semantic video search engine (de Rooij, 2008), which allows the user to browse the video collection along various dimensions to explore different characteristics of the collection.

starting to emerge with systems provided by Nuance or LexisNexis (see also Chapter 9 on speech recognition and human language technology).

15.5.4 Geographic information search

Geographic information is increasingly used to inform search (De Sabbata et al., 2015). Sensors on the ground or onboard vehicles provide the information for queries such as finding all businesses within 10 minutes of an airport. Mobile devices and digital personal assistants can use current location and direction of travel to inform searches such as "Where is the closest gas station?" User interfaces that provide map displays allow users to consider results in the context of additional geographic information and knowledge, as a nearby outdoor restaurant may not be the best option if it is too close to a noisy highway or accessible only via a toll road. Using *gazetteers* to deal with names that change over time and taking into consideration spatial synonyms are important (Samet et al., 2014), but user-generated geographic information (such as the terms used to describe places in collections of photos or tweets) is now making spatial search more "natural" by establishing the terms people use to describe places that do not have precise boundaries: downtown, the West End, and so on. Geographic information is

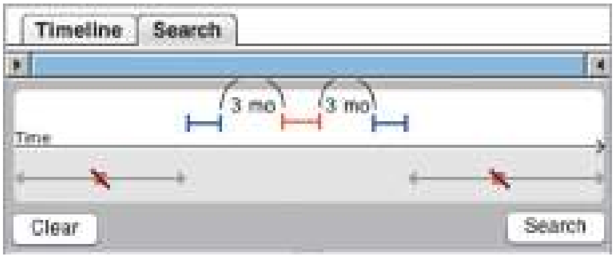


FIGURE 15.17
The EventFlow graphical search interface allows users to specify a sequence of events by placing point events or intervals icons on a timeline. Icons are explained in a separate color legend. They can specify the absence of events (which are shown crossed out) or add temporal constraints (http://www.cs.umd.edu/hcil/eventflow).

complex, and many challenges need to be addressed: deciding what to show on the map, designing dynamic legends that could summarize results (Dykes et al., 2010), and improving interaction with maps (Willett et al., 2015).

15.5.5 Multilingual searches

In some cases, users want to be able to search multilingual collections (Oard, 2009). Current web search engines merely provide translation tools, but prototype systems allow users to search multilingual collections of speech and/or printed documents in languages they do not know and provide specialized browsers for browsing results, (e.g., http://www.2lingual.com). The goal of translation systems may be to identify documents that justify the cost of high-quality professional translation.

15.5.6 Other specialized searches

Many other search interfaces are being designed to tackle specialized data types such as event sequences, graphs, structure document layouts, engineering diagrams, and so on. The graphical search box shown in Fig. 15.17 can be used by analysts to find patient records or student records that contain sequences of events of interest (Monroe et al., 2013).

15.6 The Social Aspects of Search

Evans and Chi (2010) define *social search* as "an umbrella term describing search acts that make use of social interactions with others. These interactions may be explicit or implicit, co-located or remote, synchronous or asynchronous." Users might *explicitly search* restaurant reviews in Yelp, filter results by ratings provided by thousands of reviewers, or ask for direct advice from friends and colleagues using social media sites like Facebook or Twitter (e.g., "I'm in Chicago, what jazz venue do you recommend?"). The aggregated use by prior users can also be presented in the form of explicit filters; for example, buttons for "Most viewed" and "Most shared" are helpful and comprehensible. *Social bookmarking and ranking* allows websites such as Diggs, Reddit, or Delicious to collect recommendations from thousands of people (i.e., the *wisdom of the crowd*) and explicitly rank items by popularity.

Examples of implicit use of signals left by others include the use of page rank, time spent on a page, mouse trails, or even social media connections by algorithms that select results or make similarity-based suggestions without the user necessarily realizing how the social aspects of search are coming into play. Early forms of *personalized search* relied on users having to create a profile with a set of search terms automatically applied to streaming information, such as e-mail messages, newspaper stories, or scientific journal articles. In contrast, current personalization relies on automatically building users' profiles based on their shopping, posting, or interaction history (often shared between websites), which can be aggregated and compared among groups of users to make personalized suggestions. This is a form of *implicit search* as results are presented without any query being specified. Corporate advertising in blogs or search portals might be seen by some as a form of personalized implicit search, as well as news spread by robots in social media (Lokot and Diakopoulos, 2016). The danger of such filtering results may be the creation of a *filter bubble* that closes off new ideas, subjects, products, and important information (Pariser, 2012). In general, users are more satisfied if they can tell what information was used to arrive at the recommendations, making recommendations from friends in social media sites more acceptable.

Collaborative filtering and recommender systems allow groups of users to combine their ratings to help one another find interesting items in large collections (Ekstrand et al., 2011). Each user rates items in terms of interest. The system can then recommend new items based on similarity. For example, if Joe rates six movies highly, the algorithms match him with other people who rated the same six movies highly and recommends other movies they liked (e.g., in Netflix). A slightly different social recommendation method is to follow the purchase of a product with the recommendation of another product based on the fact that other users have bought both products (e.g., in Amazon). Those strategies have an inherent appeal, and their mechanism is understood by most users, with the help of short explanatory sentences, so they are used extensively.

Music recommendation systems such as Pandora or Last.fm illustrate the appeal of combining personalization and recommenders. For example, Last.fm (Fig. 15.18) builds a profile of each user's taste by recording the musical tracks users listen to. User-generated tags categorize artists and tracks and help the site generate recommendations of similar artists or tracks. While the

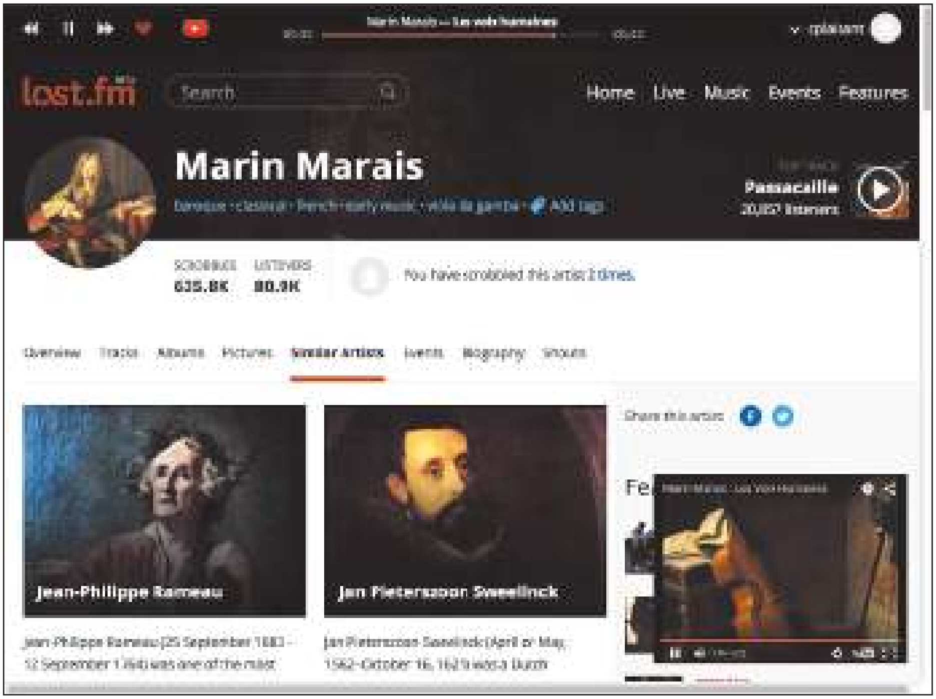


FIGURE 15.18

Last.fm is an example of online radio using playlists created automatically. The process starts by users selecting a start point (e.g., a song or artist they like); then users provide feedback on the suggestions by clicking on the heart or skipping the track.

newly suggested tracks are playing, users provide feedback explicitly (with likes) or implicitly (by skipping the track). By combining personal preferences, which serve as starting points (such as a song or an artist), and collaborative filtering to suggest tracks, the list quickly grows, and music can be generated all day. The classic search form interface vanishes in favor of a more fluid approach to search. Challenges remain when recommenders appear as a black box, leaving users sometimes puzzled by the recommendations, and researchers are exploring means to let users better guide the recommendation process (Harper et al., 2015).

Personalized search and recommendations are coming together in mobile personal assistants (see Chapter 9) as they try to answer common questions before users even ask them and trigger alerts such as "It is time to leave for your next appointment in Baltimore."

Human-powered question answering, such as in Yahoo! Answers or Ask.fm, allows users to enter questions and thousands of other users to propose answers. Voting on the quality of the answers allows the best answers to bubble up to the top.

Collaborative search corresponds to situations were users work together to conduct a search task. For example, remote family members might collaborate to plan a vacation. Collaborative search is an active research area but has not reached commercial success yet, so users are typically searching in groups without the benefit of specialized tools for collaboration. Shared documents (e.g., Google Docs) might facilitate collecting and organizing results and can be useful when complemented by e-mail, texting, or social media communication. Researchers are exploring ways to manage the division of labor (e.g., one person might be doing general search and triage, the second reviewing results and organizing leads or evidence, and a third researching leads). Improving users' awareness of each other's progress and making results, insight, and search histories persistent across sessions and among collaborators are a challenge (Morris, 2013; Shah, 2014). Future research is needed to assist teams of analysts in tracking criminals or understanding food poisoning outbreaks. As teams grow to address larger tasks, the social discovery framework (Shneiderman, 2011) may be useful to suggest how to pool the collective efforts to build better thesauri or indexes, to tag documents or objects, and to combine multiple search result suggestions.

Practitioner's Summary

While social media is changing how information reaches users, search interfaces remain a crucial component of many applications. Improved user interfaces to digital libraries and multimedia databases have spawned appealing new products. Flexible queries against complex text, sound, graphics, image, and video databases are emerging, while collaborative tagging and recommenders can eliminate the need for a search box. Faceted search and direct-manipulation approaches to query formulation effectively combine search and browsing. Very large repositories of questions and answers are making "natural" language queries a more effective specification method. Search and recommendation are evolving toward each other with personal digital assistants. Advanced search interfaces provide additional controls and even powerful command languages for users who learn to master them.

Researcher's Agenda

Although the computer contributes to the information explosion, it is potentially also the magic lens for finding, sorting, filtering, and presenting the relevant items. The need to search in structured documents, image libraries, and sound or video files still presents grand opportunities for improved user interfaces. Better understanding of the benefits and limitations of social and personalized search is needed. Automatic generation of metadata is improving, but solutions that facilitate human confirmation and manual tagging when automation fails will likely lead to more successful applications. Finally, providing collaborative search interfaces will allow teams to produce richer results and generate better insights.

WORLD WIDE WEB RESOURCES

www.pearsonglobaleditions.com/shneiderman

- Library of Congress Catalog: https://catalog.loc.gov; example collection: www.loc.gov/pictures
- Dan Russell's blog on search research: http://searchresearch1.blogspot.com
- Marti Hearst's book on search interfaces: http://searchuserinterfaces.com
- Newsstand for news search with map query interface: http://newsstand .umiacs.umd.edu/
- Multilingual search example: http://www.2lingual.com

Discussion Questions

1. Describe the challenges first-time users face when using an information-exploration system. Propose how these challenges can be overcome.
2. Argue whether textual search interfaces should keep details of how the search is performed hidden from the users. Decide which approach will allow the user to get more accurate results.
3. Describe a framework, of three to seven phases, that will help to coordinate design practices to satisfy the needs of first-time, intermittent, and frequent users who are accessing a variety of textual and multimedia libraries.
4. Explain the strategy commonly used for searching multimedia archives. List the limitations associated with this strategy.

References

Chien, L., Tat, A., Proulx, P., Khamisa, A., and Wright, W., Grand Challenge Award 2008: Support for diverse analytic techniques, nSpace2 and GeoTime visual analytics, *IEEE Visual Analytics Science and Technology Conference* (2008), 19–24.

Datta, R., Joshi, D., Li, J., and Wang, J. Z., Image retrieval: Ideas, influences, and trends of the new age, *ACM Computing Survey 40*, 2 (2008), 1–60.

Dykes, J., Wood, J., and Slingsby, A., Rethinking map legends with visualization, *IEEE Transactions on Visualization and Computer Graphics 16*, 6 (2010), 890–899.

Ekstrand, M., D., John T. Riedl, J. T., and Konstan, J. A., Collaborative filtering recommender systems, *Foundations and Trends in Human-Computer Interaction 4*, 2 (2011), 81–173.

Evans, B. M., and Chi, E. H., An elaborated model of social search, *Information Processing & Management* (2010), 31 pages.

Greene, S., Marchionini, G., Plaisant, C., and Shneiderman, B., Previews and overviews in digital libraries: Designing surrogates to support visual information-seeking, *Journal of the American Society for Information Science 51*, 3 (2000), 380–393.

Harper, F. M., Xu, F., Kaur, H., Condiff, K., Chang, S., and Terveen, L., Putting users in control of their recommendations, *Proceedings of the 9th ACM Conference on Recommender Systems*, ACM (2015), 3–10.

Hearst, M. A., *Search User Interfaces*, Cambridge University Press, New York (2009).

Hearst, M. A., "Natural" search user interfaces, *Communications of the ACM 54*, 11 (2011), 60–67.

Heesch, D., A survey of browsing models for content based image retrieval, *Multimedia Tools and Applications 40* (2008), 261–284.

Liu, T.-Y., Learning to rank for information retrieval, *Foundations and Trends® in Information Retrieval 3*, 3, Now Publishers (2009), 225–331.

Lokot, T., and Diakopoulos, N., News bots: Automating news and information dissemination on Twitter, *Digital Journalism* (2016).

Monroe, M., Lan, R., Morales del Olmo, J., Plaisant, C., Shneiderman, B., and Millstein, J., The challenges of specifying intervals and absences in temporal queries: A graphical language approach, *Proceedings of the ACM Conference on Human Factors in Computing Systems*, ACM Press, New York (2013), 2349–2358.

Morris, M. R., Collaborative search revisited, *Proceedings of the 2013 ACM Conference on Computer Supported Cooperative Work* (2013), 1181–1192.

Nudelman, G., *Designing Search: UX Strategies for eCommerce Success*, Willey (2011).

Oard, D. W., Multilingual information access, in Bates, M., and Maack, M. N. (Editors), *Encyclopedia of Library and Information Sciences*, 3rd Edition, Taylor & Francis (2009).

Pariser, E., *The Filter Bubble, What the Internet is Hiding from You*, The Penguin Press (2012).

de Rooij, O., Snoek, C., and Worring, M., Balancing thread based navigation for targeted video search, *Proceedings ACM International Conf. Image and Video Retrieval*, ACM Press, New York (2008), 485–494.

Russell-Rose, and Tate, T., *Designing the Search Experience: The Information Architecture of Discovery*, Morgan Kaufmann (2013).

De Sabbata, S., Mizzaro, S., and Reichenbacher, T., Geographic dimensions of relevance, *Journal of Documentation 71*, 4 (2015), 650–666.

Samet, H., Sankaranarayanan, J., Lieberman, M., Adelfio, M. D., Fruin, B., C., Lotkowski, J. M., Panozzo, D., Sperling, J., and Teitler, B. E., Reading news with maps by exploiting spatial synonyms, *Communications of the ACM 57*, 10 (2014), 64–77.

Sandhaus, P., and Boll, S., Semantic analysis and retrieval in personal and social photo collections, *Multimedia Tools and Applications 51*, 1 (2011), 5–33.

Schedl, M., Gómez, E., and Urbano, J., Music information retrieval: Recent developments and applications, *Foundations and Trends® in Information Retrieval, 8*, 2–3, Now Publishers (2014), 127–261.

Schoeffmann, K., Hudelist, M., and Huber, J., Video interaction tools: A survey of recent work, *ACM Computing Surveys 48*, 1 (2015), Article 14.

Schuth, A., Hofmann, K., and Radlinski, F., Predicting search satisfaction metrics with interleaved comparisons, *Proceedings of ACM Conference on Research and Development in Information Retrieval*, ACM Press, New York (2015), 463–472.

Shah, C., Capra, R., and Hansen, P., Special issue on collaborative information seeking, *IEEE Computer 47*, 3 (2014).

Shneiderman, B., Dynamic queries for visual information seeking, *IEEE Software 11*, 6 (1994), 70–77.

Shneiderman, B., Social discovery in an information abundant world: Designing to create capacity and seek solutions, *Information Services & Use 31* (2011), 3–13.

Snoek, C. G. M., and Worring, M., Concept-based video retrieval, *Foundations and Trends® in Information Retrieval 2*, 4, Now Publishers (2008), 215–322.

Willett, W., Jenny, B., Isenberg, T., and Dragicevic, P., Lightweight relief shearing for enhanced terrain perception on interactive maps, *Proceedings of the ACM Conference on Human Factors in Computing Systems*, ACM Press, New York (2015), 3563–3572.

Wilson, M., *Search User Interface Design*, Morgan & Claypool Publishers (2011).

Xu, S., Li, H., Chang, X., Yu, S., Du, X., Li, X., Jiang, L., Mao, Z., Lan, Z., Burger, S., and Hauptmann, A., Incremental multimodal query construction for video search, *Proceedings of the 5th ACM on International Conference on Multimedia Retrieval* (2015), 675–678.

Yee, K.-P., Swearingen, K., Li, K., and Hearst, M., Faceted metadata for image search and browsing, *Proceedings of the ACM Conference on Human Factors in Computing Systems*, ACM Press, New York (2003), 401–408.

Yeh, T., White, B., Pedro, J. S., Katz, B., and Davis, L. S., A case for query by image and text content: searching computer help using screenshots and keywords, *Proceedings of the 20th International Conference on World Wide Web*, ACM, Hyderabad, India (2011), 775–784.

CHAPTER 16

Data Visualization

> The real voyage of discovery consists not in seeking new landscapes but in having new eyes.
>
> **Marcel Proust**

CHAPTER OUTLINE

16.1 Introduction

Today's users are routinely engaging with larger and more complex volumes of data than ever before—and not just for professional situations as part of their jobs but also for personal and recreation purposes. For example, while it is not surprising that a business analyst has to process millions of sales records to determine a valid marketing strategy, even casual users at home need to navigate thousands of movies to find the perfect entertainment for a night in, browse hundreds of social media updates daily to keep abreast of their circle of friends, or scan through thousands of product reviews to find the right toaster to buy. Regardless of application, the medium chosen to represent the data governs the ease with which a person can perform a specific task using the information. This means that successful designers should adapt the data presentation based on what the user needs to do.

The best medium for many tasks and types of data is a *visual representation*—after all, a picture is supposedly worth a thousand words! For example, a building blueprint, a geographic map, and a digital photograph are generally best presented as 2-D pictures on the computer screen and not as a list of coordinates, colors, and shapes. Similarly, while text is the optimal presentation to convey a single number (such as the cost of a product, the distance to the supermarket, or an approval percentage), a visual presentation such as a bar chart, a line graph, or a scatterplot is often a better choice when conveying multiple related points in a dataset, such as average reviews for multiple products, stock values over time, or the relation between income and years of experience in a job. This idea of data-driven pictures is called *visualization* and is defined as the graphical representation of data to amplify cognition (Card, 2012; Ware, 2013). Visualization draws upon the massive bandwidth of our visual system to essentially allow people to "use vision to think" and dates as far back as William Playfair's line graphs and bar charts from 1786, Charles Minard's flow maps from 1869, Florence Nightingale's rose diagrams from 1857, and John Snow's cholera outbreak maps from 1854 (Fig. 16.6) (Tufte, 2001; Friendly, 2006).

In terms of Norman's gulfs of action, a macro-HCI theory describing the difference between a user's mental model and an interactive system's state (Chapter 3), visualization minimizes the gulf of evaluation because a well-designed graphical representation is optimized for many perceptual tasks. Based on this concept, Section 16.2 first presents the typical tasks that people tend to conduct using visual analysis methods. Section 16.3 then reviews typical data types and examples of common visualization techniques designed for them. This example-based framework is necessary since the visualization discipline is young and still lacks specific macro-HCI theories for selecting the optimal visual representation

See also:

Chapter 7, Direct Manipulation and Immersive Environments

Chapter 8, Fluid Navigation

Chapter 10, Devices

Chapter 11, Communication and Collaboration

Chapter 15, Information Search

that will minimize the gulf of evaluation for a dataset and task. Instead, visualization design is often empirical in nature.

Compared to the static visualizations of old, computer-based visualization has the added benefit of being interactive, which opens up vast opportunities beyond the static representations printed on paper. Similar to the above discussion, an effective interaction method for a visualization minimizes the gulf of execution—the difference between user intention and system actions—by enabling the user to easily carry out the task. However, interaction for visualization differs in many ways from typical interfaces and user applications.

Finally, much has happened in the more than two decades since the visualization field was established at the end of the last century: Computers have become faster and evolved into new forms ranging from smartphones and tablets to wall displays and tabletops, our society is awash in a deluge of data drawn from every discipline and domain, and a new generation of mobile and ubiquitous computing is turning our world into one where computing has disappeared into the fabric of everyday life (Dourish and Bell, 2011). This means that many of the foundational principles that visualization researchers traditionally have held to be true no longer are. Section 16.4 reviews the challenges facing both researchers and practitioners in data visualization.

16.2 Tasks in Data Visualization

Why do people want to interact with data? A pragmatic designer will start with the tasks that users want to perform in order to decide how to support those using interactive visual representations. Determining a standard set of such data analysis tasks has been an active area of research in the visualization community for the past two decades. One of the formative efforts in this venture was Shneiderman's *visual information-seeking mantra* from 1996: "overview first, zoom and filter, then details on demand," which still accurately captures the high-level sensemaking process (Klein, 2006) that users engage in when interacting with data. Amar et al. (2005)

approached the problem from the other direction (i.e. from the bottom up instead of from the top down), deriving 10 low-level analytic tasks that people commonly perform: retrieve value, filter, compute derived value, find extremum, sort, determine range, characterize distribution, find anomalies, cluster, and correlate. Munzner (2014) since filled in the gap between high-level sensemaking and low-level analytic tasks using a typology of abstract visualization tasks, which focuses on the why, what, and how of engaging with data at all abstraction levels.

While these efforts lay the necessary theoretical foundation for how users engage with data, they do not provide concrete guidance for designers looking to build novel visualization tools. To achieve this, this section presents a taxonomy of interactive dynamics that combine the analysis task with the practical operations that users need in their visualization tools (Heer and Shneiderman, 2012). The taxonomy consists of 12 task types grouped into three high-level categories, as shown in Box 16.1: (1) data and view specification (visualize, filter, sort, and derive); (2) view manipulation (select, navigate, coordinate, and organize); and (3) process and provenance (record, annotate, share, and guide). These three categories incorporate the critical tasks that enable iterative visual

BOX 16.1

Twelve task types for visualization organized into three high-level categories (adapted from Heer and Shneiderman [2012]).

Task Categories	***Task Types***
Data and view specification	**Visualize** data by choosing visual encodings
	Filter out data to focus on relevant items
	Sort items to expose patterns
	Derive values of models from source data
View manipulation	**Select** items to highlight, filter, or manipulate
	Navigate to examine high-level patterns and low-level detail
	Coordinate views for linked exploration
	Organize multiple windows and workspaces
Process and provenance	**Record** analysis histories for revisitation, review, and sharing
	Annotate patterns to document findings
	Share views and annotations to enable collaboration
	Guide users through analysis tasks or stories

analysis, including visualization creation, interactive querying, multi-view coordination, history, and collaboration.

For each of the 12 task types described here, examples are given that showcase the idea using real-world and predominantly commercial visualization tools. While this is by no means an exhaustive survey, these examples give practical and operational evidence of how designers can model their interfaces to support sensemaking of large-scale and complex data.

16.2.1 Data and view specification

Core functionality of any data visualization tool includes basic operations to *visualize* data using a visual representation, to *filter* out unrelated information, and to *sort* information to expose patterns. Users also need to *derive* new data from the input data, such as normalized values, statistical summaries, and aggregates. These four task types can be explained as follows:

- *Visualize data by choosing visual encodings.* Not surprisingly, selecting a visual encoding for a particular dataset is the most fundamental operation for a visualization tool. A common approach in practical visualization tools is to simply provide a palette of available charts, allowing users to easily pick the chart most appropriate for their data (Fig. 16.1). Microsoft Excel and Tableau

FIGURE 16.1

Visualization palette (upper right) in the Tableau Desktop application for a dataset of shark attacks. The "show me" feature in the tool (Mackinlay et al., 2007) will automatically highlight the suitable charts that can be used for the selected data.

both provide such palettes; in addition, Tableau also has a novel feature called "show me" that automatically selects the most appropriate visualization given the structure of the data (Mackinlay et al., 2007).

- *Filter out data to focus on relevant items.* While the overview of a dataset is often important in orienting the user in a visualization, eliminating unrelated information from the view is critical as the user starts to investigate the data in detail. Several methods for filtering exist, such as directly lassoing important objects (Choi et al., 2015) or selecting intervals and values on data dimensions using *dynamic queries.* Fig. 15.10 shows a hotel search interface on the Kayak travel website with an integrated filter interface. The interface allows for dynamically querying the hotels that match filtering criteria by changing range sliders for price intervals and selecting features by checking boxes (review scores, free breakfast, free internet, etc.). The results update dynamically as the filters are changed.
- *Sort items to expose patterns.* Ordering data items according to some dimension, such as age, income, or price, is vital in exposing hidden patterns in the data. Sorting a list of items is often easily performed by clicking the header category; toggling reverses the order.
- *Derive values of models from source data.* Original datasets can often be augmented with data computed from the original, such as statistics (e.g., mean, median), transformations, and even powerful data mining methods. In fact, calculating derived data as part of an interactive system with a user in the loop is a nascent but growing research area called *visual analytics* (Keim et al., 2008), where computational methods work in synergy with the user.

16.2.2 View manipulation

Much of the value of visualization comes from being able to manipulate the view on the screen, including the ability to *select* items or regions, to *navigate* the viewport's position on a large visualization, to *coordinate* multiple views so that data can be seen from multiple perspectives, and to *organize* the resulting dashboards and workspaces.

- *Select items to highlight, filter, or manipulate.* Pointing to an item or region of interest is common in everyday communication because it indicates the subject of conversation and action. In a visualization tool, common forms of selection include clicks (by mouse or by touch), mouse hover, and region selections (e.g., rectangular and elliptical regions or free-form lassos) (Fig. 16.2).
- *Navigate to examine high-level patterns and low-level detail.* Visualizations often contain more information than can be comfortably shown on screen, either

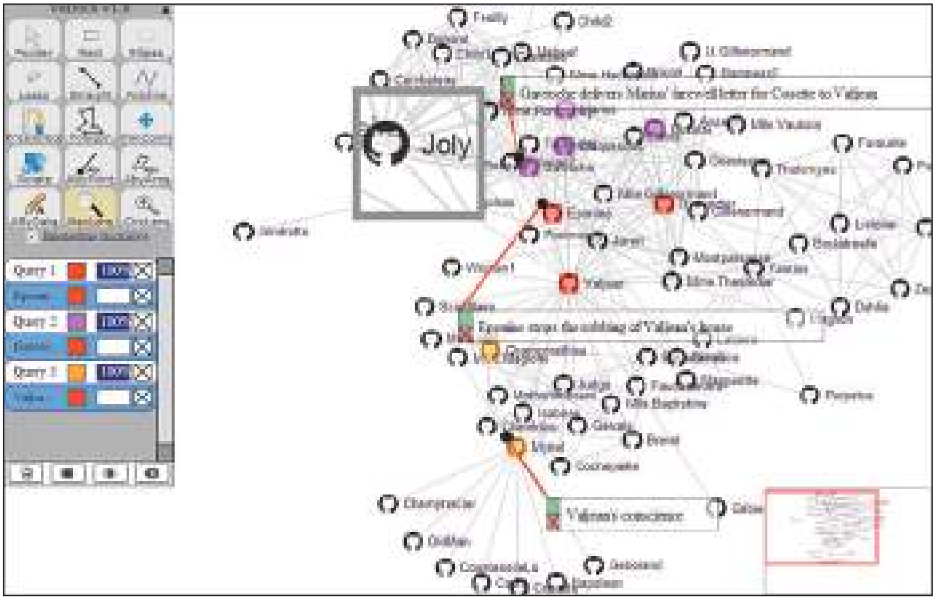


FIGURE 16.2

Selection tools and data-aware annotations in an interactive node-link diagram representation of the social network for all of the characters in Victor Hugo's *Les Misérables*. Characters are linked together if they appear in the same chapter in the book. The textual annotations are connected to nodes using red lines and stay connected as the graph layout changes. The toolbar on the upper left is part of the VisDock toolkit and provides tools for annotation, navigation, and selection (Choi et al., 2015).

due to the sheer number of pixels or just due to high visual clutter. For such dense information spaces, navigation operations such as pan and zoom allow the user to control the size and position of the viewport on the visualization (see the navigation tools in the tool dock in Fig. 16.2 or the map controls in Fig. 15.11). Not surprisingly, zooming and panning operations have now become common in many conventional user applications, such as Google Maps, Adobe Photoshop, and Microsoft Word.

- *Coordinate views for linked exploration.* Since each visualization technique has its own strengths and weaknesses, practical visualization tools often include multiple views of the same dataset so that each view illustrates a specific aspect of the data. When using multiple views in this way, such as in a visualization dashboard (Few, 2013), it is customary to coordinate the views so

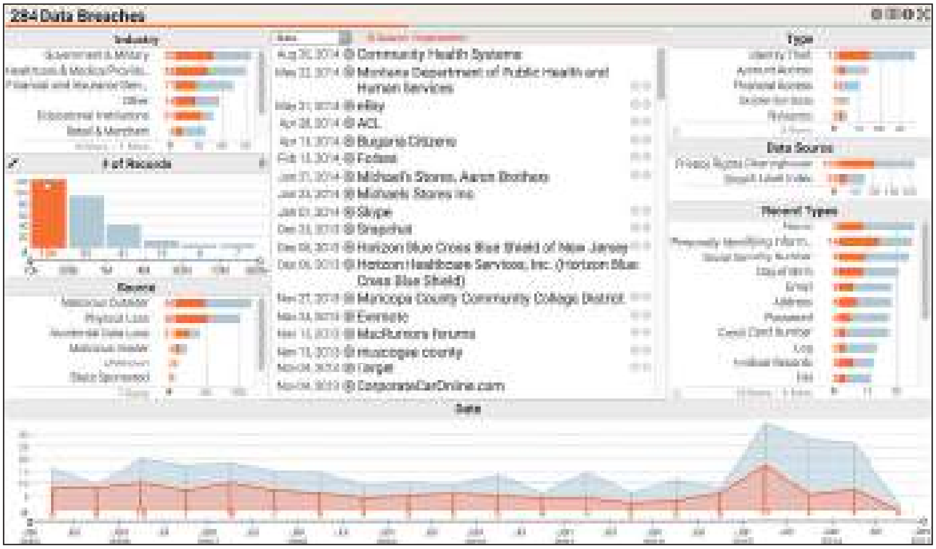


FIGURE 16.3

Exploring 284 data breaches in the United States using Keshif (http://keshif.me/), a multi-view visualization tool that shows different aspects of the data in separate views (Yalcin, 2016). Selecting items in one view highlights them in others; for example, the user is currently hovering over the bar for "70k–300k" in the view titled "# of Records," which causes those 124 breaches to also be highlighted in orange in other views, including in the timeline at the bottom.

that selecting items in one view highlights the item (or related items) in other views (Fig. 16.3). Also see Section 12.3.1 for more on this.

- *Organize multiple windows and workspaces.* While involving multiple views of a dataset allows users to explore complex data using straightforward and familiar visualizations, this also introduces the need for users to organize and lay out the views to fit their needs. Many tools allow for dragging and dropping views to achieve this, such as the Keshif tool shown in Fig. 16.3.

16.2.3 Process and provenance

If the previous two categories of tasks deal with the mechanics of creating, manipulating, and viewing visualizations, the third category encompasses higher-level tasks for scaffolding, interpreting, and documenting the exploration process. More specifically, the tasks here involve the ability to *record* the analysis, to *annotate* regions of interest in a visualization, to *share* views with colleagues, and to *guide* others through presentations of the analysis outcome.

- *Record analysis histories for revisitation, review, and sharing.* Visualization tools do not only help users collect insights from their data, they should ideally also support mechanisms to record these insights as well as the path leading up to them. One approach that several tools provide is an automatically recorded history of interactions, allowing the user to review and revisit the exploration and even share it with others (Fig. 16.4).
- *Annotate patterns to document findings.* Most visualizations use data in a read-only fashion since the goal is to let the data inform the user's exploration, but some tools allow for adding metadata in the form of textual or graphical annotations associated with the visualization (Fig. 16.2). Textual annotations constitute labels, captions, or comments, whereas graphical annotations are sketches, highlights, or handwritten notes. To be truly useful, annotations should be *data-aware* so that they are associated with underlying data points and not just drawn as a transparent layer on top of the visualization (Heer and Shneiderman, 2012; Choi et al., 2015). Drawing annotations on such a transparent layer make them meaningless when the visualization is filtered or reorganized.
- *Share views and annotations to enable collaboration.* Analyzing data is often a social activity involving multiple users working together (Heer et al., 2009), either in de facto teams or in loose constellations of people on the

FIGURE 16.4

Graphical history interface using thumbnails of previous visualization states organized in a comic-strip layout (Heer et al., 2008). The labels describe the actions performed.

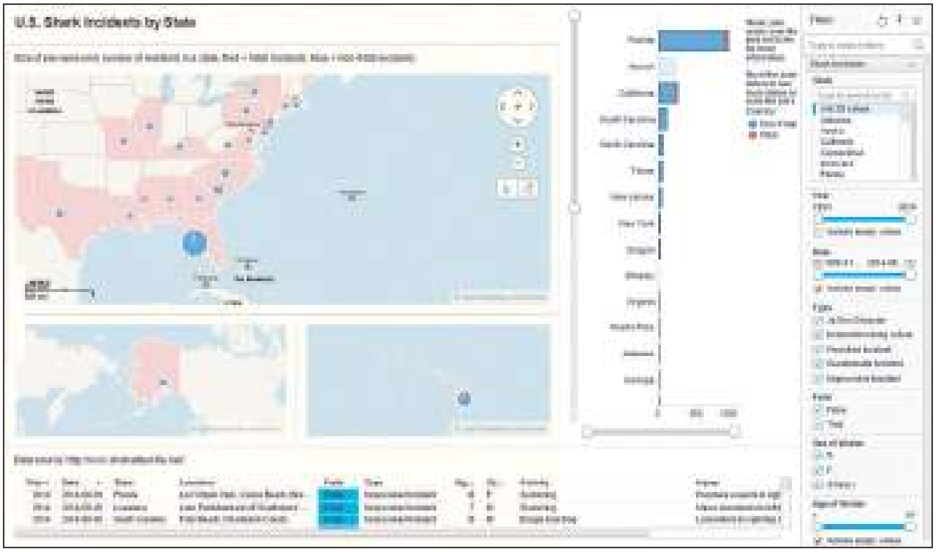


FIGURE 16.5

Spotfire visualization dashboard of shark attacks published on the web. Users can interact with the dashboard, causing views to update dynamically. The tool also allows for application bookmarking (storing the state for specific insights) as well as sharing the analysis on social media platforms such as Facebook, Twitter, and LinkedIn.

internet. The implication is clear: to support the analysis life cycle fully, visual analytics tools should support social interaction. This could include simple functionality to export shareable formats of charts (PDF, PNG, JPG, etc.) and datasets (CSV, JSON, XLS, etc.) from a visualization tool as well as more advanced sharing mechanisms such as application bookmarking and publishing visualizations on the web (Fig. 16.5).

- *Guide users through analysis tasks or stories.* As visualization tools become increasingly available to casual users looking to get insight into their own data—such as their social networks, personal finances, or local communities—there is also an increased need to guide these novices through appropriate approaches to analyze their data. Similarly, carefully crafted data stories can help explain even complex phenomena using a combination of visualizations, annotations, and textual descriptions (Fig. 16.6).

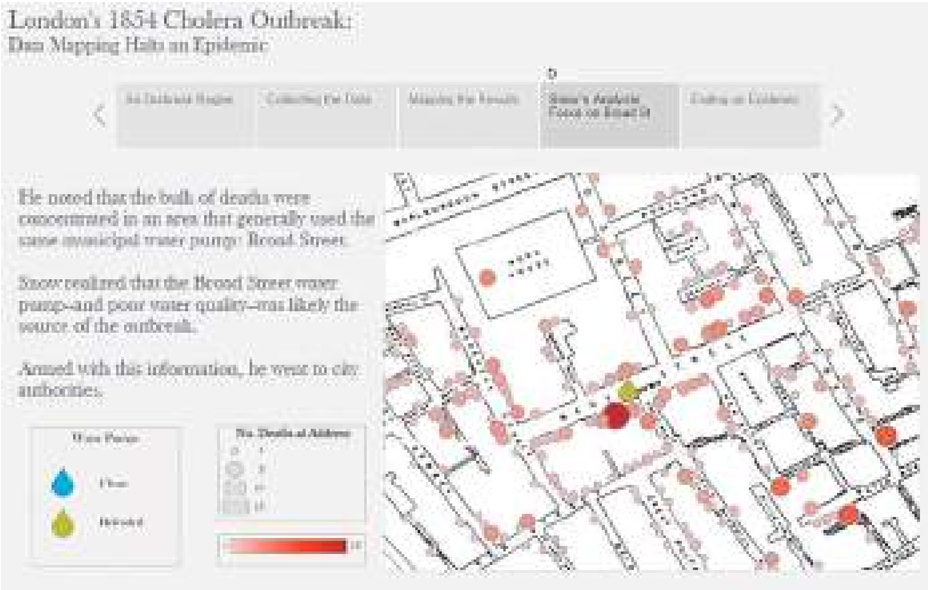


FIGURE 16.6

Web-based visualization of London's 1854 cholera outbreak showing physician John Snow's use of visualization to find its source. This visualization was created in Tableau using its Story Points feature, which allows users to build a narrative from data. The horizontal list of five boxes at the top of the display are the main points in the story, and viewers can be automatically guided through the story by moving to each point from left to right.

16.3 Visualization by Data Type

The visualization field currently lacks a unified theory that can recommend the optimal visualization technique given the data type to represent and the tasks that the user wants to perform. Furthermore, most data types do not have a straightforward mapping from symbolic to visual form; consider the complete works of William Shakespeare, a hundred years of temperature data for a thousand weather stations across the United States, or an organizational chart that changes over time as people move through the ranks—none of these datasets can be trivially rendered in graphical form. Finally, all visualization techniques have their strengths and weaknesses. The net result is that selecting the appropriate visualization technique for a dataset and task is still very much a *design problem*, similar to interaction design as a whole.

To help designers find appropriate visualizations, this section gives an overview of seven common data types and some representative visualization techniques for each type (Box 16.2). Similar to the section on tasks (Section 16.2), this section is not intended to be exhaustive, but rather to provide some concrete examples and guidelines on these design choices.

- *1-D linear data.* Linear data types are one dimensional—such as program source code, textual documents, dictionaries, and alphabetical lists of names—and can be organized in a sequential manner. Text, in particular, is a linear data type because it is designed to be read in sequence, so the challenge is to represent the data in such a way that not every word needs to be read and the parallel nature of visualization can be leveraged. Tag clouds and word clouds, which scale social tags versus words based on their frequency and arrange them in a 2-D space, originated on the social photo sharing website Flickr (Viégas and Wattenberg, 2008) and have quickly become the most common text visualization technique (Fig. 16.7). More advanced text visualizations exist that also make good use of position, phrases, and relations; the typographic maps created by Axis Maps (Fig. 16.9) and also later replicated by Afzal et al. (2012) are one example.
- *2-D space data.* Planar data include geographic maps, floor plans, and newspaper layouts. Each item in the collection covers some part of the total area and may or may not be rectangular. Each item has task-domain

BOX 16.2

Summary of data types and example visualization techniques associated with each.

Data Type	***Visualization Techniques and Systems***
1-D linear	Tag clouds, Wordle, PhraseNets, parallel tag clouds
2-D space	Geographic information systems (GIS), self-organizing maps
3-D volume	Volume rendering, medical visualization, molecule visualization
Multi-dimensional	Tableau, parallel coordinates, scatterplot matrices
Temporal	Google Finance, EventFlow, LifeLines, TimeSearcher
Tree	Treemaps, degree of interest trees, space trees
Network	Node-link diagrams, adjacency matrices, NodeXL, Cytoscape

FIGURE 16.7

While tag clouds summarize popular tags used in collaborative tagging applications, word clouds display statistics about word usage in a text collection. Here, a word cloud generated by the online generator at https://www.jasondavies.com/wordcloud/ shows the most frequent words in this chapter.

attributes, such as name, owner, and value, and interface-domain features, such as shape, size, color, and opacity. Many systems adopt a multiple-layer approach to dealing with map data, but each layer is 2-D. User tasks include finding adjacent items, regions containing certain items, and paths between items and performing the seven basic tasks. The canonical example is geographic information systems, such as Google Maps and Esri ArcGIS, but the John Snow example (Fig. 16.6), the *New York Times* 2012 electoral map (Fig. 16.8), and typographic maps (Fig. 16.9) are also forms of 2-D maps.

- *3-D volume data.* Real-world objects, such as molecules, the human body, and buildings, have volume and complex relationships with other items. Computer-assisted medical imaging, architectural drawing, mechanical design, chemical structure modeling, and scientific simulations are built to handle these complex 3-D relationships (Fig. 16.10). Users' tasks typically deal with continuous variables such as temperature or density. Results are often presented as volumes and surfaces, and users focus on relationships of left/right, above/below, and inside/outside. In 3-D applications, users must

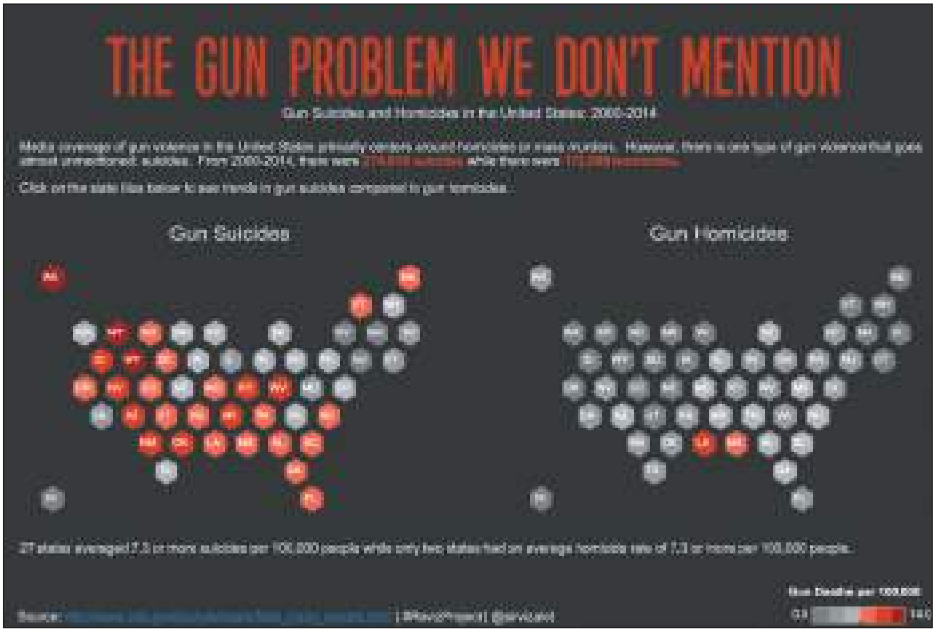


FIGURE 16.8

Geographic visualization of gun deaths (suicides versus homicides) in the United States from 2000 to 2014 using data from the Center for Disease Control and Prevention (CDC). Instead of using the actual geographic boundaries of the individual states, this map replaces states with uniform hexagons that have been color-coded using the color scale on the bottom right. The benefit of this representation is to prevent large states from dominating the visual appearance of the overall map. The hexagons have been placed so that they largely preserve the topology of the original map. (https://public.tableau.com/profile/matt.chambers#!/vizhome/TheGunProblemWeDontMention/TheGunProblemWeDontMention)

cope with their position and orientation when viewing the objects and must handle the potential problems of occlusion and navigation. Chapter 7 gives some more insight into the challenges and opportunities in 3-D immersive environments.

- *Multidimensional data.* When the number of data dimensions for a dataset exceeds the three dimensions that can be trivially rendered using 3-D graphics (Fig. 16.10), the dataset is said to be multidimensional. Such data are commonly found in relational databases as well as spreadsheets, where the columns become data dimensions and the rows become data points or items. Most tools for multidimensional visualization manage the large number of dimensions by using multiple views to show different aspects of the data. Microsoft Excel and Tableau (Fig. 16.1) are examples of such tools; in Tableau, multiple charts can also be connected (Section 16.2.2). Tableau also allows analysts to assemble multiple views into *visualization dashboards* (Few, 2013), which provide easily referenced insight into new data as they come in. A few

FIGURE 16.9

Typographic map of Washington, DC, created by Axis Maps. A typographic map consists entirely of text organized into shapes using colored labels of streets, parks, highways, shorelines, and neighborhoods. While this map took a skilled cartographer hundreds of painstaking hours to create, Afzal et al. (2012) later proposed an automatic approach taking mere minutes.

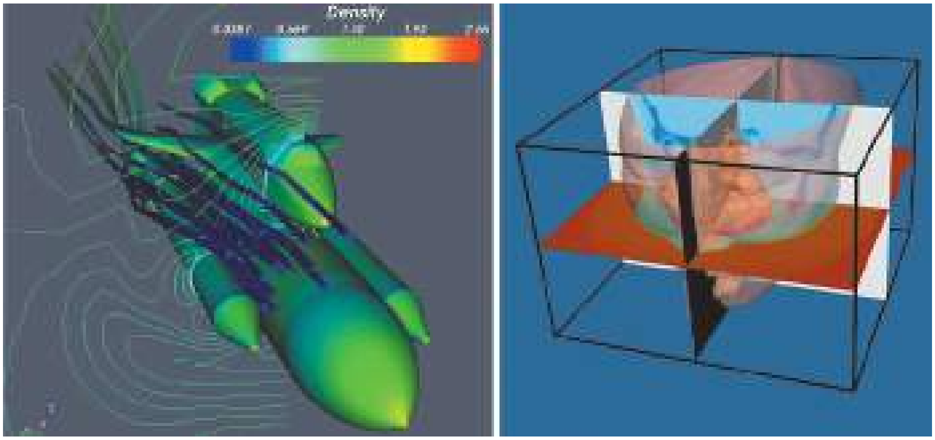


FIGURE 16.10

Two 3-D visualizations created using the Visualization Toolkit (VTK), a commercial software development library by Kitware, Inc. (http://www.kitware.com/). The left image shows flow density around the space shuttle using a rainbow color scale. The right image shows a CT scan of a human head with cross-sectional planes through the data.

visualization techniques display a large number of data dimensions in the same view, the most well-known being parallel coordinate plots (Inselberg, 2009). In a parallel coordinate plot, each parallel vertical axis represents a dimension, and each item becomes a line connecting values in each dimension (Fig. 16.11).

- *Temporal data.* Virtually all datasets can have a temporal component by collecting data points over time; examples include electrocardiograms, stock market prices, or weather data. Temporal data are separate from 1-D data in that they have different relations for time points versus time intervals and in that they are sometimes linear, sometimes cyclical, and sometimes branching (Aigner et al., 2008). Data can either be continuous (Fig. 16.12) or consist of discrete events (Fig. 16.13) that have a start and finish time and may overlap. Frequent tasks include finding all events before, after, or during some time period or moment and in some cases comparing periodical phenomena. The Gapminder tool (Fig. 12.9) visualizes time-changing data using animation and trails.
- *Tree data.* Hierarchies or trees are collections of items where each item (except the root) has a link to one parent item. Items and the links between parents and children can have multiple attributes. Interactions can be applied to items and links as well as to structural properties—for example, for

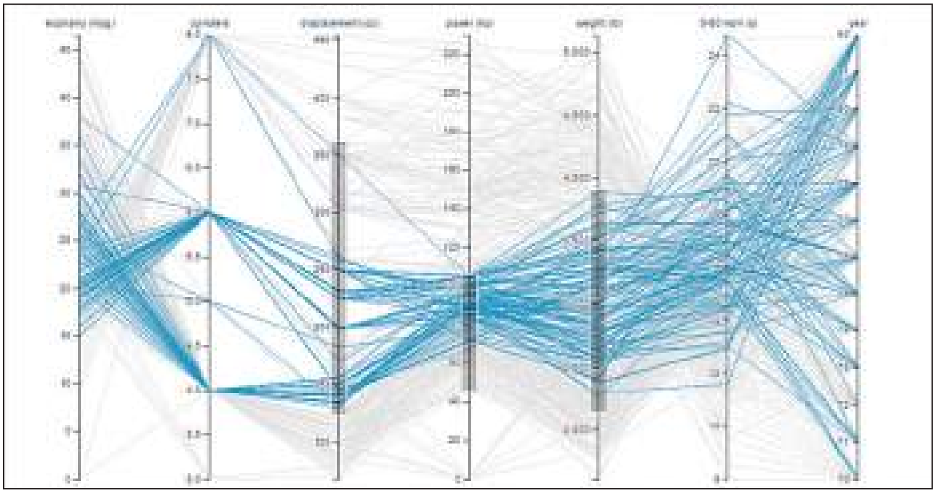


FIGURE 16.11

Parallel coordinate visualization of cars from the 1970s and 1980s created using the D3 library. This visualization supports axis filtering where selecting data ranges on the dimension axes filters out the cars that do not meet all of the criteria (gray lines).

FIGURE 16.12

Google Finance line graph showing the year-to-date performance of three stock market indices: the Dow Jones Industrial Average (.DJI, blue), the NASDAQ Composite (.IXIC, red), and the S&P 500 (.INX, yellow). The overview window at the bottom shows several years from 2011 to 2015; grabbing the window allows for panning and resizing the detail view (top).

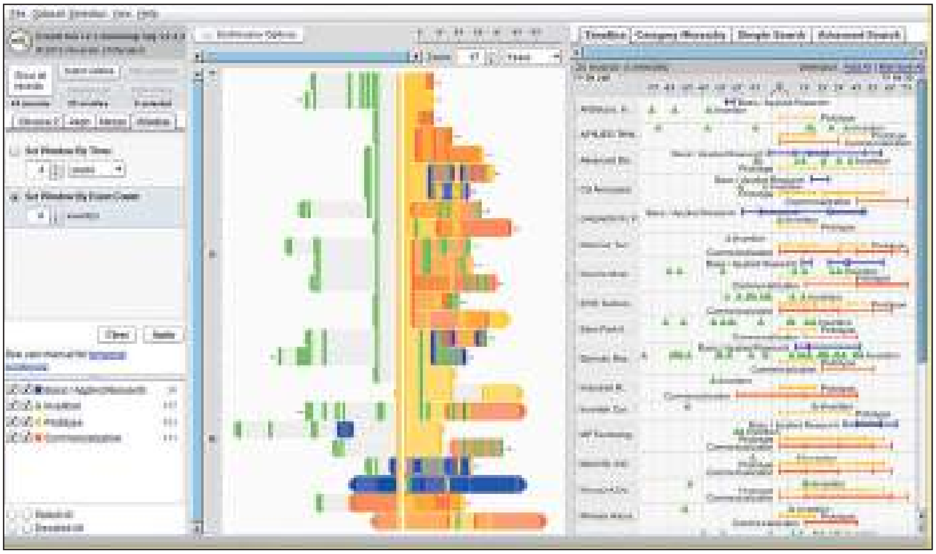


FIGURE 16.13

The EventFlow (http://www.cs.umd.edu/hcil/eventflow) temporal event visualization system used to visualize sequences of innovation activities by Illinois companies. Activity types include research, invention, prototyping and commercialization. The timeline (right panel) shows the sequence of activities for each company. The overview panel (center) summarizes all the records aligned by the first prototyping activity of the company. In most of the sequences shown here, the company's first prototype is preceded by two or more patents with a lag of about one year between the last patent application and the first prototype.

FIGURE 16.14
The S&P 500 Market Monitor by Visual Action (http://www.visualaction.com/), a web-based treemap visualization showing stock performance of the 500 large companies making up the S&P 500 index on the NYSE and NASDAQ stock markets. Each rectangle represents a company, sized according to its market capitalization, colored based on its one-day change, and organized into sectors.

a company organizational chart, is it a deep or shallow hierarchy, and how many employees does each manager supervise? Interface representations of trees can use an outline style of indented labels or a more graphical style such as a node-link diagram. Treemaps are a space-filling approach that shows arbitrary-sized trees in a fixed rectangular space (Shneiderman et al., 2012). Treemaps have been applied successfully to many applications, from U.S. budget proposals to stock market data to (Fig. 16.14).

- *Network data.* Unlike trees, networks have no single root but instead represent arbitrary relationships between items. In addition to tree tasks, network users often want to know about the shortest or least costly paths connecting two items or traversing the entire network. Node-link diagrams are one type of interface representation (Fig. 16.2), but layout algorithms are often so complex that user interaction is limited when large networks are shown, and filtering becomes important. Another option is to display an adjacency matrix, with each cell representing a potential link and its attribute values. Network visualization is an old but still imperfect art because of the complexity of relationships and user tasks. New interest in this topic has been spawned by visualization tools for social networks, such as NodeXL (Fig. 16.15 as well as Fig. 11.1).

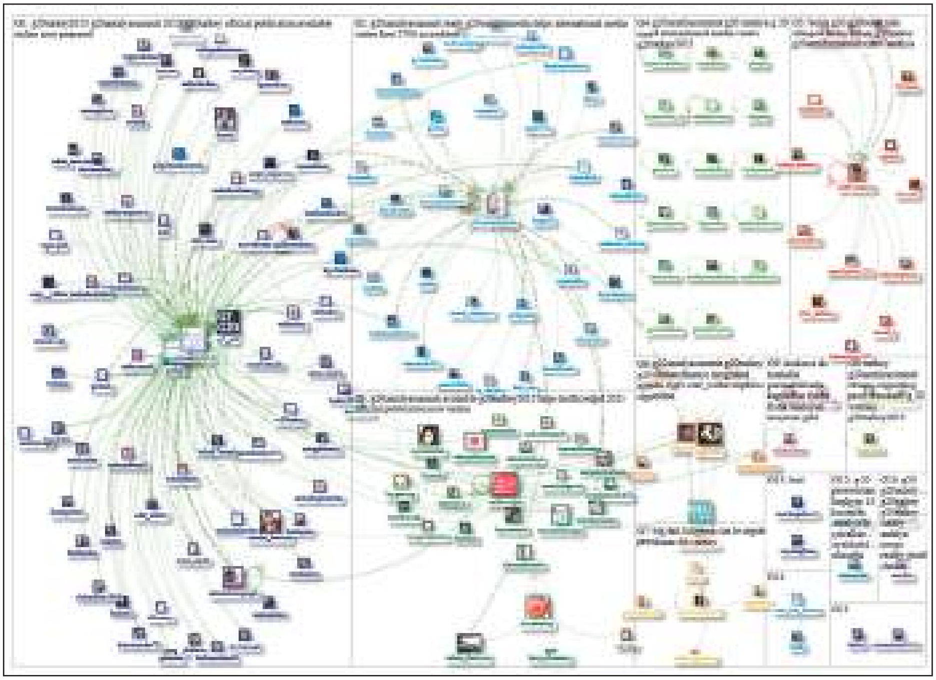


FIGURE 16.15

Social network visualization built using NodeXL (Hansen et al., 2010) of 191 Twitter users tweeting with the hashtag "#G20AntalyaSummit" on November 9, 2015. The hashtag refers to the 2015 G-20 summit held in Antalya, Turkey, on November 15–16, 2015. The users have been grouped and laid out in boxes based on the contents of the tweets. NodeXL (https://nodexl.codeplex.com/) allows social scientists to collect, analyze, and visualize network graphs using a familiar interface that plugs into Microsoft Excel.

16.4 Challenges for Data Visualization

The task and data type task taxonomies above help organize the field of data visualization. Commercial visualization tools are increasingly adopting many of these techniques. Furthermore, the number of visualization books—those that are oriented toward students and researchers, such as Munzner (2014) and Ware (2013), as well as those oriented toward designers and practitioners, such as Yau (2013) and Few (2013, 2015)—is increasing and will serve to increase the audience of visualization further. Nevertheless, there are still many challenges that researchers and practitioners alike need to face to create successful tools:

- *Importing and cleaning data.* Deciding how to organize input data to achieve a desired result often takes more thought and work than expected.

Getting data into the correct format, filtering out incorrect items, normalizing attribute values, and coping with missing data can also be burdensome tasks. This activity is also called *data wrangling* (Kandel et al., 2011) and has resulted in a commercial product called Trifacta Wrangler (https://www.trifacta.com/) as well as Datawatch Monarch (http://www.datawatch.com).

- *Integrating data mining.* Data visualization and data mining originated from two separate lines of research. Visualization researchers believe in the importance of involving users in the loop, while data mining researchers believe that statistical algorithms and machine learning can be relied on to find interesting patterns. Some consumer purchasing patterns, such as spikes in demand before snowstorms or correlations between beer and pretzel purchases, stand out when properly visualized. However, statistical tests can be helpful in finding more subtle trends in consumer desires or demographic linkages for product purchases. Increasingly, researchers are combining the two approaches to create visual tools where users are aided by powerful computational algorithms; this research area has often been called *visual analytics* and keeps the human in the loop but augments human capabilities with the computer in a synergistic way.
- *Viewing big data.* A general challenge to data visualization is the management of large volumes of data. Many tools—even commercial ones—can deal with only a few thousand or tens of thousands of items while maintaining real-time interactivity. Even if a tool is designed to manage larger datasets, there are two additional limitations that quickly come into play: the number of available pixels to show data on the screen and the number of individual points that can be perceived in practice by the user. Large displays can remedy the former limitation (Chapter 10), at least to a point, but the human perceptual limits are more difficult to circumvent. Instead, crafty visualization designers and researchers must turn to *data abstraction* (Shneiderman, 2008; Elmqvist and Fekete, 2010), where individual data points are partitioned, clustered, or sampled into smaller and more manageable numbers. Recent advances in interactive big data analytics (Fisher et al., 2012) are investigating solutions to these problems, including partial queries, incremental visualization, and streaming data.
- *Achieving universal usability.* Making visualization tools accessible to diverse users regardless of their backgrounds, technical disadvantages, or personal disabilities is necessary when the tools are to be used by the public, but it remains a huge challenge for designers (Plaisant, 2005). For example, visually impaired users may need to use text-based alternatives to the visual display. *Sonification* uses non-speech audio to convey data and can be used for graphs, scatterplots, and tables as well as potentially more complex data representations. Tactile displays (Figs. 10.25 and 10.26) can also be used to

convey data but are currently not widely available (even if 3-D printing is making this kind of "physical" visualization (Jansen et al., 2013) increasingly plausible). Users with color deficiencies can be provided with alternative palettes or tools to customize the display colors. For example, the popular red/green palette of colors can be complemented by an alternative blue/yellow palette. ColorBrewer (Fig. 12.10) and VisCheck (http://vischeck.com/) offer guidelines on color schemes that work for those with color vision impairment.

- *Supporting casual users.* The original audience for visualization was scientists, doctors, and engineers, and to this day, many tools are targeted to professionals within science, engineering, medicine, business, and journalism. However, with the rising tide of public data and the advent of powerful web-based visualization toolkits such as D3 (Bostock et al., 2011), visualization software is leaving the exclusive domain of the office and is entering the kitchens, living rooms, and, indeed, bedrooms of millions of people worldwide. This development has been dubbed *casual visualization* (Pousman et al., 2007) in that it encompasses non-expert users exploring data with personal rather than work-motivated relevance for different purposes than typical professional usage, such as for awareness, reflection, and social insight. Accepting this more inclusive definition of the audience for visualization also means that the potential for lasting impact increases manifold. In fact, as visualization gains traction for casual users, a new user group can be discerned: those who are using the visualization tools for professional purposes and have expert-level skill in their own domain but who have little training, inclination, or resources to achieve expertise in the use of visualization. Such *data enthusiasts* or *causal experts*, for lack of a better term, represent an additional opportunity for visualization to achieve more widespread adoption in the future.
- *Dissemination and storytelling.* With the radically larger potential audience for today's visualization comes the need to better convey the findings from a visual analysis without resorting to overly complex visual representations. This focus on dissemination has caused both visualization researchers and practitioners to adopt the notion of *storytelling* to build narrative visualizations that explain their findings using locations, characters, and plot (Segel and Heer, 2010; Kosara and Mackinlay, 2013). In fact, Tableau released its new Story Points mode (Fig. 16.6) in 2013 to help people create these data stories themselves.
- *Adapting to any device.* Better performance, advanced graphics, touch displays, mobile computing, and natural interfaces—needless to say, much has happened in the field of computing during the 25-odd years since visualization was introduced. However, visualization has been curiously resistant to challenging the status quo of the personal computer (Lee et al., 2012). Embracing

this revolutionary leap in technology would enable an equally transformative leap for visualization in several ways. First of all, many of these new computing platforms encompass large, multi-user displays (e.g., Figs. 10.19, 10.20, 10.21, and 10.22), which will facilitate collaborative and social visualization (Isenberg et al., 2011). Second, harnessing mobile devices would enable us to apply ubiquitous computing to anytime and anywhere sensemaking of data (Elmqvist and Irani, 2013). Third, this new generation of pen-, touch-, or gesture-based—almost “natural”—interaction may yield increased fluidity and flexibility, better freedom of expression, and reduced indirection between the person, the technology, and the data (Lee et al., 2012). In fact, the notion of a visualization may even transcend digital devices and take physical form (left side of Fig. 10.25).

- *Evaluation.* Data visualization systems can be very complex. The analysis is rarely an isolated short-term process, the tasks are high-level, and users may need to look at the same data from different perspectives over a long period of time (Carpendale, 2008). They may also be able to formulate and answer questions they didn’t anticipate having before looking at the visualization (making it difficult to use typical empirical studies techniques, where subjects are recruited for a short time to work on imposed tasks). Finally, while discoveries can have a huge impact, they occur very rarely and are unlikely to be observed during a study. Insight-based studies, as described by Saraiya, North, and Duca (2005), are one first step. Case studies report on users in their natural environments doing real tasks. They can describe discoveries, collaborations among users, frustrations of data cleansing, and the excitement of data exploration, and they can report on frequency of use and benefits gained (Perer and Shneiderman, 2008). The disadvantage of case studies is that they are very time-consuming and may not be replicable or applicable to other domains.

Practitioner’s Summary

Data visualization is moving out of research laboratories with a growing number of commercial products now available, such as Tableau Software®, TIBCO Spotfire, Trifacta Wrangler, Datawatch, IBM Cognos®, Visual Action, and Macrofocus. Meanwhile, the web has become a prime platform for visualization, with the D3 toolkit now almost having been elevated to a standard for such web-based visualizations, and commercial tools routinely providing mechanisms to publish, share, and discuss visualizations using the web. While practical tools are increasingly providing interactions and visual representations almost as soon as they are proposed by the scientific community, successful designers need to familiarize themselves with both standard data-driven tasks as well as visualization techniques to be able to navigate and select the most suitable one.

Researcher's Agenda

As data visualization is becoming mainstream, the increased exposure and impact are also calling for more fundamental, streamlined, and usable research that can be quickly adapted in commercial tools as well as included in infographics and web-based visualizations on the internet. Specific future challenges for data visualization include improving the wrangling of data prior to being able to visualize a dataset, continued integration of automatic algorithms with humans in the loop to facilitate analytical reasoning, and a renewed emphasis on big data to tackle the truly wicked problems of our society and world. This broadening appeal also means that several social factors of visualization are becoming more important than ever, including universal usability, casual users, multiple device platforms, and the focus on dissemination through storytelling to lower entry barriers. Finally, as with all of HCI, improvement is only possible in the presence of measurement, so evaluation remains a challenge for visualization researchers and practitioners alike.

WORLD WIDE WEB RESOURCES

www.pearsonglobaleditions.com/shneiderman

- Crossfilter, a JavaScript library for multivariate filtering: http://square.github.io/crossfilter/
- Cytoscape, a graph visualization platform: http://www.cytoscape.org/
- D3, a web-based visualization toolkit: http://d3js.org/
- FlowingData, a showcase of effective visualization and analysis: http://flowingdata.com/
- Gephi, a graph visualization platform: http://gephi.github.io/
- ggplot2, an R implementation of the grammar of graphics: http://ggplot2.org/
- Keshif, a multi-view data exploration tool for multi-dimensional data: http://keshif.me/
- Lyra, a visualization design platform with no programming requirements: http://idl.cs.washington.edu/projects/lyra/
- NodeXL, an Excel plugin for graph visualization: https://nodexl.codeplex.com/
- Polymaps, a JavaScript library for creating dynamic maps: http://polymaps.org/
- Processing.js, a JavaScript port of the Processing library: http://processingjs.org/
- Raphaël, a JavaScript library for vector graphics: http://raphaeljs.com/
- SHIVA, a web application for online visualization: http://www.viseyes.org/
- Vega, a declarative grammar for visualization: https://vega.github.io/
- VisDock, a JavaScript library for interaction in visualization: https://goo.gl/4l4pu1

Discussion Questions

1. Produce a definition of data visualization. Explain how it caters to the perceptual abilities of humans.
2. Describe a taxonomy of interactive dynamics that combine the analysis task with the practical operations that users need in their visualization tools of interactive dynamics that results in task types for data visualization (Heer and Shneiderman, 2012).
3. Differentiate between tree data and network data.
4. Describe three challenges data visualization researchers face when trying to build an interface. Suggest solutions to conquer these problems.

References

Afzal, Shehzad, Maciejewski, Ross, Jang, Yun, Elmqvist, Niklas, and Ebert, David S., Spatial text visualization using automatic typographic maps, *IEEE Transactions on Visualization and Computer Graphics 18,* 12 (2012), 2556–2564.

Aigner, Wolfgang, Miksch, Silvia, Müller, Wolfgang, Schumann, Heidrun, and Tominski, Christian, Visual methods for analyzing time-oriented data, *IEEE Transactions on Visualization and Computer Graphics 14,* 1 (2008), 47–60.

Amar, Robert, Eagan, James, and Stasko, John T., Low-level components of analytic activity in information visualization, *Proceedings of the IEEE Symposium on Information Visualization*, IEEE Computer Society, Washington, DC (2005), 111–117.

Bostock, Michael, Ogievetsky, Vadim, and Heer, Jeffrey, D3: Data-driven documents, *IEEE Transactions on Visualization and Computer Graphics 17,* 12 (2011), 2301–2309.

Card, Stuart, Information visualization, in Jacko, Julie (Editor), *The Human-Computer Interaction Handbook*, 3rd Edition, CRC Press, Boca Raton, FL (2012), 515–548.

Carpendale, Sheelagh, Evaluating information visualizations, in Kerren, Andreas, Stasko, John T., Fekete, Jean-Daniel, and North, Chris (Editors), *Information Visualization: Human-Centered Issues and Perspectives, Lecture Notes in Computer Science* 4950, Springer, Berlin (2008), 19–45.

Choi, Jungu, Park, Deok Gun, Wong, Yuet Ling, Fisher, Eli, and Elmqvist, Niklas, VisDock: A toolkit for cross-cutting interactions in visualization, *IEEE Transactions on Visualization and Computer Graphics 21*, 9 (2015), 1087–1100.

Dourish, Paul, and Bell, Genevieve, *Divining a Digital Future: Mess and Mythology in Ubiquitous Computing*, MIT Press (2011).

Elmqvist, Niklas, and Fekete, Jean-Daniel, Hierarchical aggregation for information visualization: Overview, techniques, and design guidelines, *IEEE Transactions on Visualization and Computer Graphics 16,* 3 (2010), 439–454.

Elmqvist, Niklas, and Irani, Pourang, Ubiquitous analytics: Interacting with big data anywhere, anytime, *IEEE Computer 46*, 4 (2013), 86–89.

Few, Stephen, *Information Dashboard Design: Displaying Data for At-a-Glance Monitoring*, 2nd Edition, Analytics Press, Burlingame, CA (2013).

Few, Stephen, *Signal: Understanding What Matters in a World of Noise*, Analytics Press, Burlingame, CA (2015).

Fisher, Danyel, DeLine, Rob, Czerwinski, Mary, and Drucker, Steven, Interacting with big data analytics, *ACM Interactions 19,* 3 (2012), 50–59.

Friendly, Michael, A brief history of data visualization, in Chen, Chun-houh, Härdle, Wolfgang, and Unwin, Antony (Editors), *Handbook of Data Visualization*, Springer Verlag, Berlin (2006), 15–56.

Hansen, Derek, Shneiderman, Ben, and Smith, Marc A., *Analyzing Social Media Networks with NodeXL: Insights from a Connected World*, Morgan Kaufmann Publishers, Burlington, MA (2010).

Heer, Jeffrey, Mackinlay, Jock D., Stolte, Chris, and Agrawala, Maneesh, Graphical histories for visualization: Supporting analysis, communication, and evaluation, *IEEE Transactions on Visualization and Computer Graphics 14*, 6 (2008), 1189–1196.

Heer, Jeffrey, and Shneiderman, Ben, Interactive dynamics for visual analysis, *Communications of the ACM 55*, 4 (April 2012), 45–54.

Heer, Jeffrey, Viégas, Fernanda, and Wattenberg, Martin, Voyagers and voyeurs: Supporting asynchronous collaborative information visualization, *Communications of the ACM 52*, 1 (2009), 87–97.

Inselberg, Alfred, *Parallel Coordinates: Visual Multidimensional Geometry and Its Applications*, Springer-Verlag, New York (2009).

Isenberg, Petra, Elmqvist, Niklas, Scholtz, Jean, Cernea, Daniel, Ma, Kwan-Liu, and Hagen, Hans, Collaborative visualization: Definition, challenges, and research agenda, *Information Visualization 10*, 4 (2011), 310–326.

Jansen, Yvonne, Dragicevic, Pierre, and Fekete, Jean-Daniel, Evaluating the efficiency of physical visualizations, *Proceedings of the ACM Conference on Human Factors in Computing Systems*, ACM Press, New York (2013), 2593–2602.

Kandel, Sean, Heer, Jeffrey, Plaisant, Catherine, Kennedy, Jessie, van Ham, Frank, Henry Riche, Nathalie, Weaver, Chris, Lee, Bongshin, Brodbeck, Dominique, and Buono, Paolo, Research directions for data wrangling: Visualizations and transformations for usable and credible data, *Information Visualization 10*, 4 (2011), 271–288.

Keim, Daniel, Andrienko, Gennady, Fekete, Jean-Daniel, Görg, Carsten, Kohlhammer, Jörn, and Melançon, Guy, Visual analytics: Definition, process and challenges, in Kerren, Andreas, Stasko, John T., Fekete, Jean-Daniel, and North, Chris (Editors), *Information Visualization: Human-Centered Issues and Perspectives, Lecture Notes in Computer Science* 4950, Springer, Berlin (2008), 154–175.

Klein, Gary, Moon, Brian, and Hoffman, Robert R., Making sense of sensemaking 2: A macrocognitive model, *IEEE Intelligent Systems 21*, 5 (2006), 88–92.

Kosara, Robert, and Mackinlay, Jock D., Storytelling: The next step for visualization, *IEEE Computer 46*, 5 (2013), 44–50.

Lee, Bongshin, Isenberg, Petra, Riche, Nathalie, and Carpendale, Sheelagh, Beyond mouse and keyboard: Expanding design considerations for information visualization interactions, *IEEE Transactions on Visualization and Computer Graphics 18*, 12 (2012), 2689–2698.

Mackinlay, Jock D., Hanrahan, Pat, and Stolte, Chris, Show me: Automatic presentation for visual analysis, *IEEE Transactions on Visualization and Computer Graphics 13*, 6 (2007), 1137–1144.

Munzner, Tamara, *Visualization Analysis and Design*, CRC Press, Boca Raton, FL (2014).

Perer, Adam, and Shneiderman, Ben, Integrating statistics and visualization: Case studies of gaining clarity during exploratory data analysis, *Proceedings of the ACM Conference on Human Factors in Computing Systems*, ACM Press, New York (2008), 265–274.

Plaisant, Catherine, Information visualization and the challenge of universal access, in Dykes, Jason, MacEachren, Alan M., and Kraak, Menno-Jan (Editors), *Exploring Geovisualization*, Elsevier, Amsterdam, Netherlands (2005).

Pousman, Zachary, Stasko, John T., and Mateas, Michael, Casual information visualization: Depictions of data in everyday life, *IEEE Transactions on Visualization and Computer Graphics 13*, 6 (2007), 1145–1152.

Saraiya, Purvi, North, Chris, and Duca, Karen, An insight-based methodology for evaluating bioinformatics visualization, *IEEE Transactions on Visualization and Computer Graphics 11,* 4 (2005), 443–456.

Segel, Edward, and Heer, Jeffrey, Narrative visualization: Telling stories with data, *IEEE Transactions on Visualization and Computer Graphics 16*, 6 (2010), 1139–1148.

Shneiderman, Ben, Extreme visualization: Squeezing a billion records into a million pixels, *Proceedings of the ACM SIGMOD Conference on the Management of Data*, ACM Press, New York (2008), 3–12.

Shneiderman, Ben, Dunne, Cody, Sharma, Puneet, and Wang, Ping, Innovation trajectories for information visualizations: Comparing treemaps, cone trees, and hyperbolic trees, *Information Visualization 11,* 2 (2012), 87–105.

Tufte, Edward R., *The Visual Display of Quantitative Information*, 2nd Edition, Graphics Press, Cheshire, CT (2001).

Viégas, Fernanda, and Wattenberg, Martin, Tag clouds and the case for vernacular visualization, *ACM Interactions 15,* 4 (2008), 49–52.

Ware, Colin, *Information Visualization: Perception for Design*, 3rd Edition, Morgan Kaufmann, Waltham, MA (2013).

Yalcin, M. Adil, Elmqvist, Niklas, and Bederson, Ben, AggreSet: Rich and scalable set exploration using visualizations of element aggregations, *IEEE Transactions on Visualization and Computer Graphics 22,* 1 (2016), 688–697.

Yau, Nathan, *Data Points: Visualization That Means Something*, John Wiley & Sons, Indianapolis, IN (2013).

AFTERWORD

Societal and Individual Impact of User Interfaces

> The machine itself makes no demands and holds out no promises: it is the human spirit that makes demands and keeps promises. In order to reconquer the machine and subdue it to human purposes, one must first understand it and assimilate it. So far we have embraced the machine without fully understanding it.
>
> **Lewis Mumford**
> *Technics and Civilization*, 1934

AFTERWORD OUTLINE

Over the past 40 years, human-computer interaction (HCI) researchers and user experience professionals have empowered billions of people with life-changing mobile devices, powerful laptop applications, remarkably diverse web tools, and vast computing cloud resources. These interfaces save lives, support families and communities, enable creative expression, expand commerce, and accelerate education. These interfaces also promote political participation, citizen science, sports engagement, shared entertainment experiences, and much more.

Of course, poor designs frustrate users, undermine privacy, and allow malicious actors to inflict widespread damage. The challenge for the future is to increase the positive outcomes while lessening the possibility of damage. Much has been made about the impact of Moore's Law, which describes the rapid increase of computer chip densities, yielding higher performance, increased storage, and dramatically reduced costs. However, the role of HCI design is as powerful as Moore's Law in bringing the power of the web and mobile devices to billions of users. This afterword considers future grand challenges for HCI researchers, designers and developers (Section A.1) and cautions about some potential dangers (Section A.2).

Internally, the field of human-computer interaction has developed an increasingly powerful set of theories, principles, and guidelines that have helped improve cell phones, touchscreen bank machines, reliable business tools, elaborate image and video editing, voting machines, and other potent technologies. Professional practitioners have developed a wide array of methods for collecting requirements, exploring designs, testing and refining prototypes, and then monitoring usage to support continuous improvement. To their credit, user-interface designers have made their products accessible for novices and experts, low- and high-literacy users, internationally diverse communities, and infants and elders as well as users with various disabilities. Much work remains to be done, but this admirable history provides a strong foundation for the next stages of discovery and innovation.

A.1 Future Interfaces and Grand Challenges

The natural question that journalists ask of HCI researchers is: What is the next big thing? One popular school of thought claims that future innovations emerge from advanced technology development, often based on Moore's Law (which described the rapid increase in chip densities, leading to the proliferation of faster and cheaper computers). Leaders of this view believe that advances will come by developing new devices, especially those that are ubiquitous and pervasive, suggesting that they can be everywhere, cheap, and small. A second theme is that these new devices will be wearable, mobile, personal, and portable, suggesting that they can be carried by users at all times. The third theme is that these new devices will be embedded, context-aware, and ambient, suggesting that they will be built into our surroundings (and thereby invisible) but available when needed and responsive to user needs. Finally, some of these new devices are labeled as perceptive and multi-modal, suggesting that they can perceive user needs and allow interaction by visual, aural, tactile, haptic, gestural, and other stimuli. The students of this school of thought have generated clever innovations, such as tiny medical sensors that monitor health, hidden detectors that protect from dangers, and entertainment devices that enrich experiences. Technology development is a fertile source of new ideas and generates rich media attention for the compelling scenarios.

The widespread dissemination of user interfaces continues, so it may be useful to lay out a set of grand challenges to steer the direction of future research, design, and commercial development. As human-computer interaction researchers, we are profoundly aware of the immense problems of our age: growing human populations consume natural resources, flourishing cities require housing and transportation, thriving families demand education and safety, and rising expectations put pressure on healthcare and social systems. National priorities include economic development, political participation, civil rights, and the delicate balance between appropriate security and excessive surveillance while pursuing open government policies that limit corruption and prevent waste.

While addressing these problems, researchers, designers, and developers who recognize the aspirations of individuals, their desire to control their own lives, and their hopes to contribute to their communities are more likely to produce constructive products and services. At the same time, these forward-looking innovators will manifest their sensitivity to individual and community norms, respect individual differences, and observe ethical codes of conduct.

With these considerations in mind, here are 16 grand challenges for human-computer interaction researchers, designers, and developers:

1. *Develop handbook of human needs.* Abraham Maslow's hierarchy of needs from the 1940s provides some guidance, establishing a foundation of survival and safety, embracing love and esteem, and supporting self-actualization that realizes personal potentials. However, a contemporary and detailed handbook of human needs would provide guidance and inspiration for designers in refining designs and inventing new tools or services.
2. *Shift from user experience to community experience.* User experience designers have cleverly invented interfaces and processes that support work, communication, and fun. Now the opportunity is to shift to community experience design, social media participation, game theoretic mechanisms, and motivational strategies to engage growing communities in constructive ways. Successful examples, such as Wikipedia or citizen science projects, show what is possible, but the common outcome of community experience

design is insufficient response, raising the question of how to make more consistently successful outcomes. This shift is mirrored in the theory shift from emphasis on micro-HCI to macro-HCI.

3. *Refine theories of persuasion.* Theories of persuasion could lead to more rapid progress in smoking cessation, obesity reduction, medication compliance, and cancer prevention. A periodic table of persuasion strategies would chart the micro-structure of motivation for designers who create applications for individuals, friends and family, colleagues and neighbors, and citizens and markets.
4. *Encourage resource conservation.* The growing needs of a growing population will have to be trimmed by efficient strategies for reducing use of water, energy, and natural resources while increasing production from renewable sources. User interfaces and community engagement will play key roles in providing feedback that encourages winning strategies.
5. *Shape the learning health system.* A grand opportunity for HCI researchers and designers is to help shape massive healthcare systems that support patients seeking wellness, clinicians delivering care, and providers eager to reduce costs while increasing the quality of care. Macro-HCI thinking and big data analytic tools could provide insights at every level, which could be shared with relevant stakeholders, but producing meaningful changes in such massive systems remains a challenge. Bottom-up strategies could propel patient and clinician participation, while top-down governance is needed to set policies, cope with malicious actors, and guide continuous improvement.
6. *Advance the design of medical devices.* Researchers have been rapidly developing medical devices that go far beyond the current hearing aids, pacemakers, body sensors, and data recording tools. As implanted insulin pumps, vision restoration systems, prosthetic limbs, brain-computer interfaces, and nano-devices mature, user interfaces to monitor performance, log activity, and enable appropriate controls will be required.
7. *Support successful aging strategies.* The growing population of older adults want to maintain their health and independence while aging in place. They could benefit from interfaces that collect data from sensors, encourage healthy diet and exercise, promote social connectedness, and enable balanced involvement from caregivers. How might the growing Internet of Things help older adults improve quality of life and maintain their independence longer?
8. *Promote life-long learning.* Traditional educational systems are expanding to include online learning (massive open online courses, or MOOCs), professional just-in-time training, learning through games, and social learning of many kinds. Developing best practices for a range of ages, motivations, and cultures will help to make these systems more reliably successful for large numbers of diverse users.

9. *Stimulate rapid interface learning.* Multi-layer user interfaces enable new users to become experts with basic features; then users can control their progress to advanced features as needed. Multi-layer user interfaces also simplify design for diverse users and users with disabilities.
10. *Engineer new business models.* As existing business models give way to new ones, user interfaces play a key role in ensuring success of peer-to-peer sales of products and services such as taxi rides, vacation home rentals, and task completion. Closer business-to-consumer connections and lower barriers to consumer-to-consumer collaboration also promise new possibilities if the user experience can be designed to promote trust, conflict resolution, and open feedback from consumer reviews.
11. *Design novel input and output devices.* As user input continues to shift from keyboards to gestures, speech, and body movement, users will need reliable mechanisms to express their intentions. Expansion of tactile and tangible environments provides fresh possibilities. Still and video cameras, 3-D scanners, and sensors will accelerate the capacity to record, analyze, and share rich data streams. Similarly, as output display diversity expands, tiny haptic feedback devices, ambient sound generators, projected displays, and large public displays will challenge designers to provide information rates and content appropriately adjusted to current tasks. Transparent glasses, immersive goggles, and ambient devices offer new possibilities along the spectrum of private to public presentation. 3-D printing and novel fabrication methods will make possible the production of physical items such as jewelry, foods, and larger objects such as chairs, cars, and building construction components.
12. *Accelerate analytic clarity.* The big data movement is generating a high volume and a variety of data whose analysis could lead to better understanding of invisible processes in business, community growth/decay, learning, or public health. Supported by well-integrated visual interfaces and statistical techniques, this better understanding could lead to more confident and bolder decisions that improve individual, community, and planetary welfare.
13. *Amplify empathy, compassion, and caring.* Human relationships flow more smoothly when empathy is expressed for others in appropriate situations. Similarly, compassionate and caring actions make life better for individuals, families, and communities. Understanding and encouraging such behaviors could make life more hope-filled and satisfying for many.
14. *Secure cyberspace.* Criminal activity and privacy violations threaten to undermine user participation in every form of transaction, participation, political engagement, and tool usage. Designing for usable privacy and security will help ensure that benefits are retained, intrusions minimized, and expectations of safety realized.

15. *Encourage reflection, calmness, and mindfulness.* Novel interfaces that encourage reflection on past experiences and intended actions with a calm and mindful attitude could enhance life experiences, creative processes, and self-awareness. Reflection about life's challenges, the needs of less fortunate people, end-of-life decisions, and the digital afterlife, while difficult, could lead to comforting clarity.
16. *Clarify responsibility and accountability.* Interfaces that clarify users' responsibility for their actions by making decisions and their outcomes visible and sometimes public could promote more appropriate behaviors with less overt bias or deception. While machine autonomy is seen as a goal by some designers, in many contexts the preferred approach may be to ensure human control while increasing the level of automation. Similarly, algorithmic accountability interfaces would allow users to better understand underlying computational processes in search, recommender, and other algorithms, giving users the potential to better control their actions.

Undoubtedly, other opportunities and unexpected developments in human-computer interaction research will occur. Conducting research on these complex sociotechnical systems requires fresh thinking as well. The traditional controlled experimental approaches associated with micro-HCI research will need to be complemented by rigorous and repeated in-depth case studies, which are part of macro-HCI research. Addressing these problems will require improved interdisciplinary methods that emerge from science, engineering, and design.

A.2 Ten Plagues of the Information Age

"The real question before us lies here: Do these instruments further life and enhance its values, or not?"

Lewis Mumford
Technics and Civilization, 1934

It would be naïve to assume that widespread use of user interfaces brings only benefits. There are legitimate reasons to worry that increased dissemination of information and communication technologies might lead to personal, organizational, political, or social oppressions. People who fear computers, robots, and other technologies have good reason for their concerns. The frustration of users who can't accomplish their tasks and the disruptions caused by network failures that shut down airline and other systems are legitimate causes of concern. Furthermore, unwanted e-mail (spam), malicious viruses, pornography, and other annoyances must be addressed so that users can benefit from advanced technologies.

User-interface designers have an opportunity and a responsibility to be alert to the dangers and to make thoughtful decisions about reducing them. The potential and real dangers from use of information and communication technologies include:

1. *Anxiety.* Many people avoid computers or mobile devices or use them with great anxiety; they experience *computer shock*, *web worry*, or *network neurosis*. Their anxieties include fear of breaking the machine, worry over losing control, trepidation about appearing foolish or incompetent ("computers make you feel so dumb"), or more general concern about facing something new. These anxieties are real, should be acknowledged rather than dismissed, and can often be overcome with positive experiences. Can we build improved user interfaces that will reduce the heightened level of anxiety experienced by many users?
2. *Alienation.* As people spend more time using computers and mobile devices, they may become less connected to other people. Computer users as a group are more introverted than others, and increased time with technology may increase their isolation. The dedicated game player who rarely communicates with another person is an extreme case, but what happens to the emotional relationships of a person who spends eight hours per day dealing with e-mail instead of chatting with colleagues or family members? Can we build user interfaces that encourage more constructive human social interaction?
3. *Information-poor minority.* Although some utopian visionaries believe that information and communication technologies will eliminate the distinctions between rich and poor or will right social injustices, often these tools are just another way in which the disadvantaged are disadvantaged. People who have weak computer skills may have a new reason for not succeeding in school or not getting a job. The well-documented differences in access by rich versus poor communities or nations can be overcome if we recognize them and make commitments to bridging the gap by offering appropriate access, training, support, and services. Can we build user interfaces that empower low-skilled workers to perform at the level of experts? Can we provide training and education for every member of society?
4. *Impotence of the individual.* Large organizations can become impersonal because the cost of handling special cases is great. Individuals who are frustrated in trying to receive personal treatment and attention may vent their anger at the organization, the personnel they encounter, or the technology that limits rather than enables. People who have tried to find out the current status of their Social Security accounts or have banks explain accounting discrepancies are aware of the problems, especially if they have language or hearing deficits or other physical or cognitive handicaps. How can we design so that individuals will feel more empowered and self-actualized?

5. *Bewildering complexity and speed.* The tax, welfare, and insurance regulations developed by computer-based bureaucracies are so complex and fast-changing that it is extremely difficult for individuals to make informed choices. Even knowledgeable technology users are often overwhelmed by the torrent of new software packages, mobile devices, and web services, each with hundreds of features and options. Simplicity is a simple, but too often ignored, principle. Stern adherence to basic principles of design may be the only path to a safer, more sane, simpler, and slower world where human concerns predominate.
6. *Organizational fragility.* As organizations come to depend on more complex technology, they can become fragile. When networks break down, security lapses, or virus attacks occur, they can propagate rapidly and halt the work of many people. With computer-based airline services, communications, or electricity grids, failures can mean rapid and widespread shutdowns of service. Since networks have many entry points, a small number of people can disrupt a large organization. Can developers anticipate the dangers and produce robust, fault-tolerant designs?
7. *Invasion of privacy.* The widely reported threat of invasion of privacy is worrisome because the concentration of information and the existence of powerful retrieval systems make it possible to violate the privacy of many people easily and rapidly. Of course, well-designed computer systems have the potential of becoming more secure than paper systems if managers are dedicated to privacy protection. Airline, telephone, bank, medical, legal, and employment records can reveal much about an individual if confidentiality is compromised. Can managers seek policies and systems that reduce privacy threats from cyber-criminals, governments, or companies?
8. *Unemployment and displacement.* As automation spreads, productivity and overall employment may increase, but some jobs may become less valued or even eliminated. Retraining can help some employees, but others will have difficulty changing lifetime patterns of work. Especially in recessionary times, displacement may happen to low-paid clerks or highly paid machine operators whose work is outsourced overseas or automated. Can employers develop labor policies that ensure retraining and guarantee jobs?
9. *Lack of professional responsibility.* Faceless organizations may respond impersonally to and deny responsibility for problems. The complexity of technology and organizations provides ample opportunities for employees to pass the blame on to others or to the computer: "Sorry, the computer won't let us give you a mortgage." Will designers and users of electronic medical systems, driverless cars, or defense-related user interfaces be able to escape responsibility for decisions? Will user interfaces become more trusted than a person's word or a professional's judgment? Complex and confusing user interfaces enable users and designers to blame the machine, but with improved designs, users and designers will give and accept credit and responsibility where they are due.

10. *Deteriorating image of people.* With the development of *intelligent interfaces*, *smart machines*, and *expert systems*, it seems that the machines have indeed *taken over* human abilities. These misleading phrases not only generate anxiety about computers and robots but also may undermine the image that we have of people and their abilities. Some behavioral psychologists suggest that we are little more than machines; some artificial intelligence workers believe that the automation of many human abilities is within reach. The unbounded creativity of humans, the deep trust and empathic relationship among people, and the imagination of each child seem lost or undervalued. Can robotic scenarios for medical services, elder care, and warfare be tempered with an appreciation for the role of human compassion and judgment?

Undoubtedly, more plagues and problems exist. Each situation is a small warning for designers. Each design is an opportunity to apply technology in positive, constructive ways that avoid these dangers. There is no sure vaccine for preventing the 10 plagues outlined here. Even well-intentioned designers can inadvertently spread them, but alert, dedicated designers whose consciousness is raised can reduce the dangers. Strategies for preventing the plagues and reducing their effects include:

- **Human-centered participatory design.** Concentrate attention on the users and the tasks they must accomplish. Make users the center of attention, include them in the design process, and build feelings of competence, mastery, clarity, and predictability. Construct well-organized menus, present specific and constructive instructions and messages, develop comprehensible displays, offer informative feedback, enable error prevention, and ensure appropriate response times.
- **Organizational support.** Beyond the interface design, the organization must also support the users. Apply human-centered design strategies and elicit frequent evaluations and feedback from users. Techniques include personal interviews, focus groups, online surveys, online community discussions, and accessible customer service.
- **Job design.** European labor unions have been active in setting rules for computer users to prevent the exhaustion, stress, or burnout. Rules might be set to limit hours of use, guarantee rest periods, facilitate job rotation, and support education. Similarly, negotiated measures of productivity or error rates can help reward exemplary workers and guide training. Monitoring or metering of work must be done cautiously, making both managers and employees beneficiaries of a thoughtful plan.
- **Education.** The complexity of modern life and user interfaces makes education critical. Schools and colleges, as well as employers, all play a role in training. Special attention should be paid to continuing education, on-the-job training, and teacher education.

- **Feedback, recognition, and rewards.** User communities have become actively engaged in providing user-generated content and participating in governance. They can help set community norms that promote respectful behavior, encourage design improvements by communicating with managers and designers, and help fellow users to learn what they need. Constructive contributions should be acknowledged by recognition such as the ACM Awards for professional contributions and the Webby Awards for effective design.
- **Public consciousness raising.** Informed consumers and users of information and communications technologies can benefit the entire community. Professional societies such as the ACM, IEEE, HFES, and UXPA and user groups can play a key role through public relations, consumer education, and professional standards of ethics.
- **Legislation.** Much progress has been made with legislation concerning privacy, right of access to information, and computer crime, but more work remains to be done. Cautious steps toward regulation, work rules, and standardization can be highly beneficial. Dangers of restrictive legislation do exist, but thoughtful legal protection will stimulate development while ensuring employee safety and health.
- **Advanced research.** Individuals, organizations, and governments can support research to develop novel ideas, minimize the dangers of technology, and spread the advantages of interactive systems. Improved theories of cognitive behavior, individual differences, community evolution, and organizational change would be helpful in guiding designers and implementers.

Practitioner's Summary

Successful interactive user interfaces will bring ample rewards to their designers, but widespread use of effective tools is only the means to reach higher goals. A user interface is more than a technological artifact; interactive systems, especially when linked by computer networks, create human sociotechnical systems. As Marshall McLuhan pointed out, "The medium is the message," and therefore each interactive user interface is a message from the designer to the user. That message has often been a harsh one with the underlying implication that the designer does not care about the user. Nasty error messages are obvious manifestations; complex menus, cluttered screens, and confusing dialog boxes can also contribute to user frustration.

Most designers want to send a more kind and caring message. Designers, implementers, and researchers are learning to send warmer greetings to users via effective and well-tested user interfaces. The message of quality is compelling to the recipients and can instill good feelings, appreciation for the designer, and the desire to excel in one's own work. More than ever, user-interface designers have a large challenge before them and a share in the responsibility for the development of human relationships. Let's use this opportunity well to create a better world.

Researcher's Agenda

Interface designers can work toward furthering such high-level goals as world peace, excellent healthcare, energy efficiency, adequate nutrition, and safe transportation. In addition to these admirable goals, designers should aspire to advance the causes of accessible education for all, improved communication, freedom of expression, support for creative exploration, and socially constructive entertainment. User interfaces can help users attain these high-level goals if designers clearly state measurable objectives, obtain the participation of professionals, and gather feedback from users. Design considerations include adequate attention to individual differences among users, support of social and organizational structures, design for reliability and safety, and provision for access by older adults and physically challenged and low-literacy users.

The expectation that new devices will be universally usable and support creativity could launch numerous ambitious research projects. Terror prevention, disaster response, international development, medical informatics, e-commerce, and government services are appealing candidates for research because the impact of changes could be so large. If designers are to provide novel services to diverse users, they will need effective theories and rigorous empirical research to achieve ease of learning, rapid performance, low error rates, and good retention over time while preserving high subjective satisfaction.

WORLD WIDE WEB RESOURCES

www.pearsonglobaleditions.com/shneiderman

- Many grand challenges await eager researchers:
  http://www.engineeringchallenges.org/challenges.aspx
  https://sustainabledevelopment.un.org/
- Organizations dealing with ethics, social impact, and public policy are doing their best to make computing and information services as helpful as possible:
  http://www.acm.org/about/code-of-ethics
  http://www.sigcas.org/
  http://www.ieee.org/about/corporate/governance/p7-8.html

This page intentionally left blank

Name Index

The name index includes only names appearing in the chapter text. References following the chapter text are not indexed. Therefore, some paper co-authors may not appear in the name index (e.g., in the case of a paper cited in the text as (Smith, J. et al., 2016) for legibility, only Smith will appear in the index, even though the co-authors' names appear in the reference section.

S

T

U

V

W

Subject Index

A

K

L

M

N

O

P

Q

R

S

T

U

V

Credits

Frontmatter
13 (top to bottom): Photo by BK Adams (bkadams@cs.umd.edu); Photo by Al Santos (asantos@umd.edu); Photo taken by NSU photography team, NSU Office of Publications; **14** (top to bottom): Photo taken by NAU photography team: (Marketing.Photography@nau.edu); Photo by Niklas Elmqvist (elm@umd.edu).; photo by Nicholas Diakopoulos (nad@umd.edu).

Part 1
22: Ratchanida Thippayos/123RF; Monty Rakusen/Cultura/Getty Images; Monkey Business/Fotolia; RooM the Agency/Alamy Stock Photo; PhotosIndia.com LLC/Alamy Stock Photo

Chapter 1
25: Excerpt from The Home Computer Revolution by Theodor Holm Nelson. Published by Nelson © 1977.; **27:** (1.1a) Screenshot of Google Search with Voice Activation. Copyright © by Google Inc. Used by permission of Google Inc.; (1.1b) Screenshot of Zombies, Run! Copyright © by Six to Start. Used by permission of Six to Start.; (1.1c) Screenshot of Twitter. Copyright © by Twitter Inc.; Bill Buxton; Ben Shneiderman; Catherine Plaisant; **28:** (1.2) John Klinovsky, Screenshot of "Apple® Mac OS X" reprinted with permission from Apple Inc., Screenshot of Picasa., Screenshot of Google Map. Copyright © by Google Inc. Used by permission of Google Inc.; (1.3) Catherine Plaisant; **29:** (1.4) Jennifer-Golbeck, Billy Tidwell, Sterling, Screenshot of Amazon Web Site Copyright © by Amazon.; **30:** (1.5) NASA Web Site; **31:** (1.6a) "BodyVis: A New Approach to Body Learning Through Wearable Sensing and Visualization" © 2015 Association for Computing Machinery, Inc. Reprinted by permission. (1.6b) "BodyVis: A New Approach to Body Learning Through Wearable Sensing and Visualization" © 2015 Association for Computing Machinery, Inc. Reprinted by permission. (1.7) Screenshot of Hipmunk. Copyright © by Hipmunk. Used by permission of Hipmunk; Icons of United Airlines in screenshot of Hipmunk.; Icons of Alaska Air Group in screenshot of Hipmunk; Icons of American Airlines in screenshot of Hipmunk; **36:** (1.8) Ted Soqui/Corbis; **37:** (1.9) Screenshot of Allscripts Wand™. Copyright © by Allscripts. Used by permission of Allscripts.; **38:** (1.10) Screenshot of Autodesk SketchBook Pro. Copyright © by Autodesk Inc. Used by permission of Autodesk Inc. **39:** (1.11a) Screenshot from *USENIX Journal of Election Technology and Systems* (JETS), Vol. 2, p. 2 (2014). (1.11b) Screenshots of Clear Ballot. Copyright © by Clear Ballot Group Inc. Used by permission of Clear Ballot Group Inc.

Chapter 2
57: Excerpt from "The Data on Diversity" by Beryl Nelson, *Communications of the ACM*, vol. 57(11), pp. 86–95. Published by ACM © 2014.; Quote from Reply to American Philosophical Society by Thomas Jefferson, 1808; **58:** (2.1) Courtesy of Raising the Floor - International Association; **65:** Excerpt from World Summit on the Information Society,

Outcome Documents Geneva 2003—Tunis 2005. Copyright © 2005 by International Telecommunication Union (ITU), Geneva. Used by permission of International Telecommunication Union (ITU).; (2.2a) Microsoft Research India; (2.2b) Microsoft Research India; **66:** (2.3) Raising The Floor; **69:** (2.4): Inria/Photo P.-O Gaumin; **70:** Excerpt from Human Factors Research Needs for an Aging Population edited by Sara J Czaja. Published by National Academy Press, © 1990; **72:** (2.5a) Reflective Thinking; (2.5b) Screenshot of King Ted in Digital Mysteries. Copyright © by Reflective Thinking. Used by permission of Reflective Thinking

Chapter 3
81: Quote from *Thoughts* by Horace Mann. Published by H B Fuller in 1867.; Quote from *Philosophy in the Twentieth Century* by Alfred Jules Ayer. Published by Weidenfeld and Nicolson, © 1982. **83:** (3.1) Copyright © 2016 Apple Inc. All rights reserved. **87:** (3.2) Screenshot of Cerner Design. Copyright © by Cerner Corporation. Used by permission of Cerner Corporation.; **92:** (3.3) Pearson; Education, Inc.; **97:** Quote by Sigmund Freud.; **103:** (3.4) Windows 10, Microsoft Corporation.

Part 2
122: Ratchanida Thippayos/123RF; Monty Rakusen/Cultura/Getty Images; Monkey Business/Fotolia; RooM the Agency/Alamy Stock Photo; PhotosIndia.com LLC/Alamy Stock Photo

Chapter 4
125: Quote from *Principles in Design* by W H Mayall. Published by Design Council, © 1979.; Quote from *Towards a New Architecture* by Le Corbusier. Published by Courier Corporation, © 1931.; **131:** (4.1) Pearson Education; **140:** (4.2) Brenna McNally; **141:** (4.3) John Consoli; **142:** (4.4) Pearson Education; Sketch by Senthil Kumaran Chandrasegaran. Copyright © by Senthil Kumaran Chandrasegaran. Used by permission of Senthil Kumaran Chandrasegaran. **143:** (4.5a) Sketch created by Joran Booth. Copyright © by Joran Booth. Used by permission of Joran Booth; (4.5b) Sketch created by Lauren Kretsch. Copyright © by Lauren Kretsch. Used by permission of Lauren Kretsch; **147:** (4.6) Pearson Education; **148:** (4.7) Pearson Education; **149:** (4.8) “LensMouse: augmenting the mouse with an interactive touch display” © 2010 Association for Computing Machinery, Inc. Reprinted by permission.

Chapter 5
167: Quote from *Gravity and Grace* by Simone Weil. Published by Routledge and Kegan Paul, © 1947. **172:** (Box 5.1) Excerpt from “Table 2 - Game heuristics” from “Heuristic Evaluation for Games: Usability Principles for Video Game Design” by David Pinelle, Nelson Wong and Tadeusz Stach in *Proceedings of ACM CHI 2008 Conference on Human Factors in Computing Systems*, April 2008, http://dl.acm.org/citation.cfm?doid=1357054.1357282, Copyright © 2008 by ACM, Inc. Used by permission of ACM, Inc. **174:** (Box 5.2) From “Making Usability Recommendations Useful and Usable” in *Journal of Usability Studies*. Published by ACM New York, © 2007; **176:** (5.1) Noldus Information Technology; **177:** (5.2) Excerpt from A Quick and Dirty Usability Scale in Usability Evaluation In Industry edited by Patrick W. Jordan, B. Thomas, Ian Lyall McClelland and Bernard Weerdmeester. Published by Taylor & Francis, © 1996.; User-Centered Design Process Map (Retitled as Usability.gov Step by Step Usability Guide), U.S. Department of Health & Human Services; **181:** (Box 5.3) From

Moderating Usability Tests: Principles and Practices for Interacting. Published by Morgan Kaufmann, © 2008. **182:** (5.6) Dotshock/Shutterstock; **189:** (Table 5.1) Table From About the QUIS, version 7.0. Copyright by University of Maryland. Used by Permission of University of Maryland. **196:** (5.8) LDprod/Shutterstock; **198:** (5.9) Figure from TechSmith. Copyright © by TechSmith Corporation. Used by permission of TechSmith Corporation. **201:** (Box 5.4) From Seven Pitfalls to Avoid When Running Controlled Experiments on the Web in KDD '09 *Proceedings of the 15th ACM SIGKDD International Conference on Knowledge Discovery and Data Mining*. Published by ACM New York, © 2009.

Chapter 6

211: Quote from "The Value of Prototyping" in *Prototyping: A Practitioner's Guide* by Todd Zaki Warfel. Published by Rosenfeld Media, © 2009; Quote by Mike Davidson. Copyright © by Mike Davidson. **214:** (6.1) Wavebreakmedia/Shutterstock; **217:** (6.2) Screenshot reprinted with permission from Apple Inc. Steven Jacobs; **218:** (6.3) Screenshot of iPhone Mail Screens reprinted with permission from Apple Inc.; Definition of Big Data from Google. Published by Google. **219:** (6.4) From "Volvo Single View of Vehicle: Building a Big Data Service from Scratch in the Automotive Industry" by P. Wozniak, R. Valton, and M. Fjeld in CHI EA '15 *Proceedings of the 33rd Annual ACM Conference Extended Abstracts on Human Factors in Computing Systems*, pp. 671–678, May 2015. Copyright © 2015 by the Association for Computing Machinery, Inc. Used by permission of Pawel Wozniak; **220:** (6.5) From "Volvo Single View of Vehicle: Building a Big Data Service from Scratch in the Automotive Industry" by P. Wozniak, R. Valton, and M. Fjeld in *CHI EA '15 Proceedings of the 33rd Annual ACM Conference Extended Abstracts on Human Factors in Computing Systems*, pp. 671–678, May 2015. Copyright © 2015 by the Association for Computing Machinery, Inc. Used by permission of Pawel Wozniak.

Part 3

226: Ratchanida Thippayos/123RF; Monty Rakusen/Cultura/Getty Images; Monkey Business/Fotolia; RooM the Agency/Alamy Stock Photo; PhotosIndia.com LLC/Alamy Stock Photo

Chapter 7

229: From "Leibniz" by Frederick Kreiling in Scientific American 218(5), May 1968. Published by Scientific American, © 1968. **234:** (7.1) Reflecting Thinking; **235:** (7.2) Art Olson Lab; **239:** (7.3) Maxine Cohen; **242:** (7.4) Niklas Elmqvist; **243:** (7.5) Alejik/Shutterstock; **246:** (7.6) Web-based 3d Interactive Model, by Floored, Inc.; **247:** (7.7) NASA; **249:** (7.8) National Science Foundation; **252:** (7.9) Phil Jones/U.S. Department of Veterans Affairs; **253:** (7.10) Master Video/Shutterstock; **255:** (7.11) Marcio Jose Sanchez/AP Images; (7.12) Keita Higuchi; **257:** (7.13) Polhemus; (7.14) OCULUS VR, LLC; **258:** (7.15) Figure 1 - Simplified Representation of a RV Continuum" from "Augmented Reality: A Class of Displays on the Reality-Virtuality Continuum" by Paul Milgram, Haruo Takemura, Akira Utsumi, and Fumio Kishino, 1994. Copyright © 1994 by Paul Milgram. Used by permission of Paul Milgram. Copyright © 1994 by SPIE. Used by permission of SPIE. **259:** (7.16) Illustration by Brian G. Payne; **260:** (7.17) Nokia City Lens; MicroSoft; (7.18) IKEA; **261:** (7.19) HITLab; **262:** (7.20) Image with permission from University of Southern California Institute for Creative Technologies; **263:** (7.21) Laura Downey

Chapter 8

271: From A Certain World: A Commonplace Book by W. H. Auden. Published by The Viking Press, © 1994. **274:** (8.1) Pearson Education; **275:** (8.2a) Carol Boston; (8.2b) Screenshot of Zombies, Run! Copyright © by Six to Start. Used by permission of Six to Start.; (8.3) Pearson Education; **276:** (8.4) Pearson Education; **277:** (8.5) Powerpoint, Microsoft Corporation; **278:** (8.6) © 2016 Recreational Equipment, Inc. All rights reserved. REI and the REI co-op logo are trademarks of Recreational Equipment, Inc; (8.7a-c) Screenshot of Soundhound. Copyright © by SoundHound Inc.; **280:** (8.8) Screenshot of Screenshot of Autodesk Fusion 360. Copyright © by Autodesk, Inc. Used by permission of Autodesk, Inc.; **283:** (8.9) Ben Shneiderman, Catherine Plaisant, Maxine Cohen, Steven Jacobs; (8.10) Screenshot of NASA; **284:** (8.11) Screenshot of craigslist. Copyright © by craigslist; **285:** (8.12) Screenshot of AWWWARDS ONLINE S.L. Copyright © by AWWWARDS ONLINE S.L. Used by permission of AWWWARDS ONLINE S.L.; **286:** (8.13) © 2016 Recreational Equipment, Inc. All rights reserved. REI and the REI co-op logo are trademarks of Recreational Equipment, Inc; **288:** (8.14a) Imaginechina/Corbis; (8.14b) Rodrigo Reyes Marin/AFLO/Nippon News/Corbis; (8.14c) Niklas Elmqvist; (8.14d) Niklas Elmqvist; **293:** (8.15) Word for Mac 2011, Office 2013. Microsoft Corporation; **297:** (8.16) Copyright © 2016 by IEEE. All rights reserved. Use of this Website signifies your agreement to the IEEE Terms and Conditions. Used by permission of IEEE. **302:** (8.17) Screenshot of Google CAPTCHA. Copyright © by Google Inc. Used by permission of Google Inc.; (8.18) Windows, Microsoft Corporation; **303:** (8.19) Reprinted with permission of The University of Texas Health Science Center at Houston, copyright 2013.

Chapter 9

311: From "On a Method of Expressing by Signs the Action of Machinery" by Charles Babbage in Philosophical Transactions of the Royal Society of London 116(1/3):250–265. Published by the Royal Society, 1826; **316:** (9.1) Cody Dunne; (9.1b) Screenshot from Mozilla. Copyright © by Mozilla.; **319:** (9.2a) Screenshot of Siri from IPhone reprinted with permission from Apple Inc.; (9.2b) Google and the Google logo are registered trademarks of Google Inc., used with permission; (9.2c) Cortona, Microsoft Corporation; (9.2d) Screenshot of Hound. Copyright © by SoundHound Inc.; **320:** (9.3) Screenshot of Correcting a word during dictation using Nuance Dragon. Copyright © by Nuance Communications, Inc. Used by permission of Nuance Dragon; Word, Microsoft Corporation; **321:** (9.4) Google and the Google logo are registered trademarks of Google Inc., used with permission; **323:** (9.5) Copyright © by Nuance Communications, Inc. Used by permission of Nuance Dragon; **328:** (9.6) US Navy; (9.7) Google and the Google logo are registered trademarks of Google Inc., used with permission; **330:** (9.8) Screenshot of Sublime Text 2. Copyright © by Sublime HQ PTY LTD. Used by permission of Sublime HQ PTY LTD.

Chapter 10

337: Quote from *The Medium is the Massage: An Inventory of Effects by Marshall McLuhan and Quentin Fiore.* Published by Bantam Books, © 1967; **339:** (10.1) Gurinder Osan/AP Images; (10.2) Piotr Adamowicz/Alamy Stock Photo; **340:** (10.3a) Screen shot(s) reprinted with permission from Apple Inc.; (10.3b) Lukas Wroblewski/Shutterstock; **342:** (10.4) Blue Orb, Inc.;

344: (10.5a) Image © 2016 BlackBerry Limited. All Rights Reserved. Used with permission; (10.5b) Scott Disanno; **345:** (10.6) Niklas Elmqvist; **351:** (10.7a) Screen shot(s) reprinted with permission from Apple Inc.; (10.7b) Niklas Elmqvist; **352:** (10.8) Olga Popova/Shutterstock; **353:** (10.9) Maxine Cohen; (10.10a) Scott Disanno; (10.10b) Scott Disanno; **354:** (10.11) Photo Courtesy of Wacom Technology Corporation; **357:** (10.12) Trace R & D Center, University of Wisconsin-Madison; **358:** (10.13a) Steve Landau; (10.13b) Steve Landau; **359:** (10.14) "Figure 3 - Paths Taken by Participants to Click on a 32 Pixel Target at a Distance of 256 Pixels" in "Differences in Pointing Task Performance Between Preschool Children and Adults Using Mice" by Juan Pablo Hourcade, Benjamin B. Bederson, Allison Druin, and Francois Guimbretiere from ACM Transactions on Computer-Human Interaction 11(4): 357–386, December 2004. Copyright © 2004 by Association for Computing Machinery, Inc. Used by permission of ACM, Inc.; **361:** (10.15) LiveScribe; **363:** (10.16) Leap Motion, Inc.; **366:** (10.17) Prof. Ben Shneiderman; **367:** (10.18) Tom Costello; **368:** (10.19) Stony Brook University; **369:** (10.20) Inria/Photo H. Raguet; **370:** (10.21) Denise Smith; **371:** (10.22) "Branch-explore-merge: facilitating real-time revision control in collaborative visual exploration" © 2012 Association for Computing Machinery, Inc. Reprinted by permission; **373:** (10.23a) Niklas Elmqvist; (10.23b) Niklas Elmqvist; **374:** (10.24) Pearson Education; **376:** (10.25a) Yvonne Jansen; (10.25b) Dr. Jason Alexander; (10.25c) "PaperPhone: understanding the use of bend gestures in mobile devices with flexible electronic paper displays" © 2011 Association for Computing Machinery, Inc. Reprinted by permission; (10.26) "inFORM: dynamic physical affordances and constraints through shape and object actuation" © 2013 Association for Computing Machinery, Inc. Reprinted by permission;

Chapter 11

385: Quote by Henry Ford. Copyright © by The Henry Ford; Quote from the movie, *The Social Network.* Published by Relativity Media, © 2015; **389:** (11.1) My2k Twitter NodeXL SNA Map and Report for Tuesday, 08 January 2013 at 01:25 UTC from NodeXL Graph Gallery. Copyright © 2013, by Marc Smith. used by permission of Marc Smith; **391:** (11.2) Pearson Education; **393:** (11.3) Screenshot of Google Docs. Copyright © by Google Inc. Used by permission of Google Inc.; **394:** (11.4) Screenshot of A Wikipedia Page After Editing, Wikipedia, NA; **396:** (11.5) Screenshot of Graphic with three badges from the game DUST, Copyright © by Brigham Young University. Used by permission of Brigham Young University; **397:** (11.6) Risa Chubinsky; **399:** (11.7) Screenshot of Fitbit App in iPhone. Copyright © by eBay Inc. Used by permission of eBay Inc.; **401:** (11.8a) Tom Bartindale; (11.8b) Tom Bartindale; **402:** (11.9) From Fleischmann's Vienna Model Bakery, New York Public Library; **406:** (11.10) Screenshot of Twitter. Copyright © by Twitter.com; (11.10) Tweet by Diakopoulos. Used by permission of Nick Diakopoulos; (11.10) Tweet by Charles Ornstein. Copyright © by ProPublica. Used by permission of ProPublica; **407:** (11.11) Ben Shneiderman, Catherine Plaisant, Maxine Cohen, Steven Jacobs; **408:** (11.12) Screenshot of GitHub, Inc. Copyright © by GitHub, Inc. Used by permission of GitHub, Inc.; **411:** (11.13) A Yelp user's profile page on Yelp.com. Copyright © by Yelp Inc. Used by permission of Yelp Inc.; Screenshot of Yelp Account of David Ayman Shamma, http://ayman.yelp.com/. Used by permission of David Ayman Shamma; **414:** (11.14) Catherine Plaisant

Part 4
422: Ratchanida Thippayos/123RF; Monty Rakusen/Cultura/Getty Images; Monkey Business/Fotolia; RooM the Agency/Alamy Stock Photo; PhotosIndia.com LLC/Alamy Stock Photo

Chapter 12
425: From On Shame and the Search for Identity by Helen Merrell Lynd. Published by Routledge Ltd, © 2013; **427:** (12.1) NASA; **432:** (12.2) US Government; **433:** (12.3) Copyright © by Association for Computing Machinery, Inc. Reprinted by permission; (12.4) Screenshot of Apple iTune reprinted with permission from Apple Inc. **435:** (12.5) Ben Shneiderman, Catherine Plaisant, Maxine Cohen, Steven Jacobs; **436:** (12.6) Visual Studio, Microsoft Corporation; **438:** (12.7) Ben Shneiderman, Catherine Plaisant, Maxine Cohen, Steven Jacobs; (12.8) Figure 2: A Fisheye View of the Graph in Figure 1 in "Graphical Fisheye Views" by Manojit Sarkar and Marc H Brown from *Communications of the ACM 37*(12): 73–83, Copyright © December 12, 994 by Association for Computing Machinery, Inc. Used by permission of ACM, Inc. **440:** (Box 12.2) Ben Shneiderman, Catherine Plaisant, Maxine Cohen, Steven Jacobs; **441:** (12.9) Free material from www.gapminder.org; **447:** (12.10) Screenshot of Color Brewer.org by Cynthia A. Brewer. Copyright © by Pennsylvania State University. Used by permission of Pennsylvania State University; **449:** (Box 12.4) From "Web-Based Presentation of Information: The Top Ten Mistakes and Why They Are Mistakes" by Thomas S. Tullis in *HCI International 2005 Conference*, Las Vegas, Nevada, USA. Published by HCI International, © 2005.

Chapter 13
465: Quote from *Principles of Psychology*, Vol. 01 by William James. Published by Henry Holt & Company in 1890; Quote from *Journal* by Henry David Thoreau. Published by Houghton Mifflin Harcourt in 1906; **467:** (13.1) NASA; **476:** (13.4) Screenshot of Picasa Upload Manager. Copyright © by Google Inc. Used by permission of Google Inc.

Chapter 14
485: From *Blink: The Power of Thinking Without Thinking* by Malcolm Timothy Gladwell. Published by Little Brown, © 2005; **488:** (14.1) OneNote 2016, Microsoft Corporation; **492:** (14.3) From F-Shaped Pattern For Reading Web Content by Jakob Nielsen, https://www.nngroup.com/articles/f-shaped-pattern-reading-web-content/, April 2006. Copyright © 1998–2016 Nielsen Norman Group, All Rights Reserved. Used by permission of the Nielsen Norman Group; **493:** (Box 14.1) From *Letting Go of the Words: Writing Web Content That Works* by Janice Redish. Published by Elsevier, © 2012; **496:** (14.4) Screenshot of FmPro Migrator. Copyright © by .com Solutions Inc. Used by permission of .com Solutions Inc.; **497:** (14.5) Word, Microsoft Corporation; **498:** (14.6) Screenshot from Saxopedia music. Published by Creative Commons; **499:** (14.7) Neirfy/Shutterstock; **499:** (14.8a-b) Google and the Google logo are registered trademarks of Google Inc., used with permission; **500:** (14.9) National Library of Medicine; **501:** (14.10) Rochelle and Joe Nappi; **502:** (14.11) Jaejeung Kim; Image © Ken Hinckley, 2015. Used by permission. All rights reserved; **505:** (14.12) Figure adapted from Tagged Comments: Promoting and Integrating User Comments on Online Application Tutorials by Andrea Bunt, et al. Copyright © 2014 Association for Computing Machinery, Inc. Reprinted by permission; **507:** (14.13) Monkey Business Images/Shutterstock

Chapter 15

517: "Song" from *Songs of Innocence and Experience* by William Blake. Published in 1789; **519:** (15.1) Library of Congress; **520:** (15.2) Library of Congress; **524:** (15.3) Screenshot of Yahoo; **525:** (15.4a) Screenshot of Apple Iphone Contact Manager reprinted with permission from Apple Inc; (15.4b) Screenshot of Amazon. Copyright © by Amazon.com, Inc; (15.4c) Adobe product screenshot reprinted with permission from Adobe Systems Incorporated; **527:** (15.5) Screenshot of Google Search. Copyright © by Google Inc. Used by permission of Google Inc.; **528:** (15.6) Screenshot of Zillow. Copyright © by Zillow, Inc.; **529:** (15.7) Screenshot of NCSU Libraries. Copyright © by NC State University. Used by permission of NC State University; **530:** (15.8a) Screenshot of Google Search. Copyright © by Google Inc. Used by permission of Google Inc.; (15.8b) Screenshot of Peapod. Copyright © by Peapod, LLC.; **531:** (15.9) Screenshot of Nspace TRUST. Copyright © 2012–2016 by Uncharted Software Inc. Used by permission of Uncharted Software Inc. See Reference: The Sand Box for Analysis - Concepts and Methods by W. Wright, D. Schroh, et al. in ACM CHI, 2006. **532:** (15.10) Screenshot of nSpace Sandbox. Copyright © 2012–2016 by Uncharted Software Inc. Used by permission of Uncharted Software Inc. See Reference: Information Triage Trist by D. Jonker, W. Wright, et al. in Proc. International Conference on Intelligence Analysis, 2005; **534:** (15.12) Screenshot of Kayak. Copyright © by Kayak.com. Used by permission of Kayak.com; **535:** (15.13) Screenshot of Amazon. Copyright © by Amazon.com, Inc.; **536:** (15.14) Screenshot of Yahoo; **540:** (15.15) Figure 1: We Demonstrate the MediaMill ForkBrowser on the TRECVID 2007 Video Collection from MediaMill: Fast and Effective Video Search Using the ForkBrowser by Ork de Rooij, Cees G.M. Snoek, and Marcel Worring. Copyright © 2008 by Informatics Institute. Used by permission of Informatics Institute; **541:** (15.16) "Figure 1" from MediaMill: Fast and Effective Video Search Using the ForkBrowser by Ork de Rooij, Cees G.M. Snoek, and Marcel Worring. Copyright © 2008 by Ork de Rooij, Cees G.M. Snoek, and Marcel Worring. Used by permission of Informatics Institute; **542:** (15.17) Screenshot of Youtube; **544:** (15.18) Screenshot of Last.fm. Copyright © by Last.fm Ltd. Used by permission of Last.fm Ltd.

Chapter 16

551: Quote by Marcel Proust; **554:** (Box 16.1) Excerpt from *Communications of the ACM 55*(4):45–54. Copyright © 2012 by Association for ComputingMachinery, Inc. Reprinted by permission; **555:** (16.1) Screenshot of Tableau Desktop Application. Copyright © by Tableau Software. Used by permission of Tableau Software; **557:** (16.2) "Figure 8 - Node-link Diagram with Force-Directed Layout" from "VisDock: A Toolkit for Cross-Cutting Interactions in Visualization" by Jungu Choi, Deok Gun Park, Yuet Ling Wong, Eli Fisher, and Niklas Elmqvist in IEEE Transactions on Visualization and Computer Graphics 21(9):1087–1100, March 2015. Copyright © 2015 by IEEE. Used by permission of IEEE; **558:** (16.3) Screenshot of Keshif. Copyright © by University of Maryland. Used by permission of University of Maryland; **559:** (16.4) "Figure 1 - A Graphical History Interface" from Graphical Histories for Visualization: Supporting Analysis, Communication, and Evaluation by Jeffrey Heer, Jock D. Mackinlay, Chris Stolte, and Maneesh Agrawala in IEEE Transactions on Visualization and Computer Graphics 14(6):1189–1196, November-December 2008. Copyright © 2008 by IEEE. Used by permission of IEEE; **560:** (16.5) Used with the permission of TIBCO Software Inc. © TIBCO Software Inc. All rights reserved; **561:** (16.6) Screenshot from TLondon's 1854 Cholera Outbreak: Data Mapping Halts an Epidemic. Copyright © by Tim Deak. Used by permission of Tim Deak; **563:** (16.7) Screenshot from jasondavies.com/wordcloud by Jason Davies. Copyright © by Jason Davies. Used by Permission from Jason Davies; **564:** (16.8) Screenshot from The Reviz Project: The Gun Problem We Don't Mention. Copyright © by Matt Chambers. Used by permission of Matt Chambers; **565:** (16.9) Screenshot from Typographic Map of Washington D.C. Copyright © by Axis Maps. Used by permission of Axis Maps; (16.10a) Visualization Toolkit (VTK); (16.10b) Visualization Toolkit (VTK); **566:** (16.11) Screenshot of Parallel Coordinates by Mike Bostock from "http://bl.ocks.org/mbostock/7586334". Used by permission of Mike Bostock; **567:** (16.12) Screenshot of Google Finance Line Graph. Copyright © by Google Inc. Used by permission of Google Inc.; (16.13) The Eventflow by Christopher Scott Dempwolf. Used by permission of the author; **568:** (16.14) Screenshot of Visual Action. Copyright © by Visual Action Software, Inc. Used by permission of Visual Action Software, Inc.; **569:** (16.15) NodeXL Social Media Network Map of Connections Among Twitter Users Who Mention "G20" - Dr. Marc A. Smith, Social Media Research Foundation, 2015.

Afterword

578: Ratchanida Thippayos/123RF; Monty Rakusen/Cultura/Getty Images; Monkey Business/Fotolia; RooM the Agency/Alamy Stock Photo; PhotosIndia.com LLC/Alamy Stock Photo

This page intentionally left blank