Third Edition

Designing with the Mind in Mind

Simple Guide to Understanding User Interface Design Guidelines

Jeff Johnson

MK
MORGAN KAUFMANN

Designing with the Mind in Mind

Simple Guide to Understanding User Interface Design Guidelines

Third Edition

Jeff Johnson

ELSEVIER

MORGAN KAUFMANN PUBLISHERS

AN IMPRINT OF ELSEVIER

Morgan Kaufmann is an imprint of Elsevier
50 Hampshire Street, 5th Floor, Cambridge, MA 02139, United States

Copyright © 2021, 2014, 2010 Elsevier Inc. All rights reserved.

No part of this publication may be reproduced or transmitted in any form or by any means, electronic or mechanical, including photocopying, recording, or any information storage and retrieval system, without permission in writing from the Publisher. Details on how to seek permission, further information about the Publisher's permissions policies and our arrangements with organizations such as the Copyright

Clearance Center and the Copyright Licensing Agency, can be found at our website: www.elsevier.com/permissions.

This book and the individual contributions contained in it are protected under copyright by the Publisher (other than as may be noted herein).

Notices
Knowledge and best practice in this field are constantly changing. As new research and experience broaden our understanding, changes in research methods, professional practices, or medical treatment may become necessary.

Practitioners and researchers must always rely on their own experience and knowledge in evaluating and using any information, methods, compounds, or experiments described herein. In using such information or methods they should be mindful of their own safety and the safety of others, including parties for whom they have a professional responsibility.

To the fullest extent of the law, neither nor the Publisher, nor the authors, contributors, or editors, assume any liability for any injury and/or damage to persons or property as a matter of products liability, negligence or otherwise, or from any use or operation of any methods, products, instructions, or ideas contained in the material herein.

Library of Congress Cataloging-in-Publication Data
A catalog record for this book is available from the Library of Congress

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

ISBN: 978-0-12-818202-4

For information on all Morgan Kaufmann publications
visit our website at https://www.elsevier.com/books-and-journals

Publisher: Katey Birtcher
Acquisition Editor: Steve Merken
Editorial Project Manager: Chris Hockaday
Production Project Manager: Radhika Sivalingam
Cover Designer: Patrick C.Ferguson

Typeset by TNQ Technologies

ELSEVIER Book Aid International
Working together to grow libraries in developing countries
www.elsevier.com • www.bookaid.org

Foreword

Jeff Johnson's impactful books on user interface design are distinct in emphasizing the foundations of good design decisions in practical human psychology. This remains a key insight into design generally and user interface design specifically. Design is rarely just a matter of knowing and following rules; more typically, it involves recognizing how and when rules apply and do not apply, adapting rules in context, and formulating new rules.

How can designers manage such open-ended complexity? Johnson's answer is that designers should understand the rationales and foundations for design rules in the psychology of human perception and cognition. Humans perceive, think, and act in distinctive ways, with distinctive strengths and weaknesses. Understanding human characteristics allows designers to reason creatively about the applicability of design rules in context and to better explain and justify their decisions.

Designs ultimately succeed or fail because humans can understand them, enjoy them, learn from them, and use them effectively. This can seem obvious, until we take a critical look at our world, filled with designs that don't effectively convey what they do or how they do it, frustrate us and waste our time, and do not respect or leverage the amazing human mind. Designing with the Mind in Mind is an important tool to help designers succeed.

—John M. Carroll

John M. Carroll is Distinguished Professor of Information Sciences and Technology and Director of the Center for Human-Computer Interaction at Pennsylvania State University.

Acknowledgments

Although I am listed as the author of this book, I could not have written it without a lot of help and support from many people.

First are the students of the human–computer interaction course I taught as an Erskine Fellow at the University of Canterbury in New Zealand in 2006. It was for them that I developed a lecture providing a brief background in perceptual and cognitive psychology—just enough to enable them to understand and apply user-interface design guidelines. I expanded that lecture into a professional development course to present at conferences and client companies, then further expanded it to produce the first edition of *Designing with the Mind in Mind*.

In 2013, I had another Erskine Fellowship at the University of Canterbury and used the first edition to teach another human–computer interaction course there as well as give computer science department lectures at the UoC, the University of Waikato, and the CHI-NZ 2013 conference. I especially thank my colleagues at the University of Canterbury who provided ideas and illustrations for the second edition's new chapter on Fitts' law: Professor Andy Cockburn, Dr. Sylvain Malacria, and Mathieu Nancel. I also thank my colleague and friend Professor Tim Bell for sharing user-interface examples and for other help and support while I was writing the second edition. The feedback I received motivated me to add more comprehensive psychological background material, expand the topics covered, improve the explanations, and update the examples in producing the second edition, which was published in 2014.

I used the second edition as the basis for conference keynotes, guest lectures, and conference tutorials in the USA and elsewhere. In August 2016, I joined the Computer Science faculty of the University of San Francisco and now use the book in an upper-division UX design course I teach there.

By 2019, it became clear that *Designing with the Mind in Mind* needed updating again. Not only were many examples again looking dated, but digital technology had entered a new era in which mobile, artificial intelligence, and speech technology are much more prominent. It also was clear that the book, originally intended mainly for professional UI/UX designers, was being widely used as a college textbook. I suppose that should not have been surprising, since teaching college students was my inspiration for writing the first edition and is the main way I currently use the book.

The reviewers of all three editions made helpful comments and suggestions that allowed me to greatly improve the book:

- Edition 1: Susan Fowler, Robin Jeffries, Tim McCoy, and Jon Meads.
- Edition 2: Susan Fowler, Robin Jeffries, James Hartman, Victor Manuel González y González, Mihaela Vorvoreanu, and Karin M. Wiburg.
- Edition 3: Darren Hood, David W. Meyer, Matt Swaffer, Emily Wenk, and KS.

I am also grateful to several cognitive science researchers who directed me to important references, shared useful illustrations with me, or allowed me to bounce ideas off of them:

- Professor Edward Adelson, Department of Brain and Cognitive Sciences, Massachusetts Institute of Technology.
- Dr. Dan Bullock, Department of Cognitive and Neural Systems, Boston University.
- Dr. Eva Hudlicka, College of Information and Computer Sciences, University of Massachusetts, Amherst; Principal Scientist at Psychometrix Associates.
- Dr. Marisa Knight, Department of Psychology, University of San Francisco.
- Dr. Amy L. Milton, Department of Psychology and Downing College, University of Cambridge.
- Professor Dan Osherson, Department of Psychology, Princeton University.

I also am extremely grateful to Professor John M. "Jack" Carroll, Penn State University, for contributing a foreword to this new edition of the book.

The book also was helped greatly by the care, oversight, logistical support, and nurturing provided by the staff at Elsevier.

Last but not least, I thank my wife and friend, documentary photographer Karen Ande, for her love and support throughout the many years that I have been researching, writing, and revising this book.

Introduction

USER-INTERFACE DESIGN RULES: WHERE DO THEY COME FROM AND HOW CAN THEY BE USED EFFECTIVELY?

For as long as people have been designing interactive computer systems, some have attempted to promote *good* design by publishing user-interface design guidelines (also called design rules), including these early ones:

- **Cheriton** (1976) proposed user-interface design guidelines for early interactive (time-shared) computer systems.
- **Norman** (1983a,b) presented design rules for software user interfaces based on human cognition, including cognitive errors.
- **Smith and Mosier** (1986) wrote perhaps the most comprehensive set of user-interface design guidelines.
- **Shneiderman** (1987) included "Eight Golden Rules of Interface Design" in the first edition of his book *Designing the User Interface* as well as in all the editions that followed.
- **Brown** (1988) wrote a book of design guidelines appropriately titled *Human–Computer Interface Design Guidelines.*
- **Nielsen and Molich** (1990) offered a set of design rules for use in heuristic evaluation of user interfaces, and Nielsen and Mack (1994) updated them.
- **Marcus** (1992) presented guidelines for graphic design in online documents and user interfaces.

More recently, in the 21st century many more user-interface design guidelines have been published:

- Several authors have written books that include UI/UX design guidelines: Stone et al. (2005), Koyani et al. (2006), Johnson (2007), Shneiderman and Plaisant (2009), and Rosenberg (2020).
- A few organizations have published usability guidelines for the Web: W3C (2016), Yale University (2020).
- Computer companies have published guidelines for designing desktop apps on their platforms: Microsoft Corporation (2018), Apple Computer (2020a).

- Providers of mobile application platforms also have published guidelines to help developers create more usable apps for mobile devices: Oracle Corporation (2017), Google (2019), Apple Computer (2020b).

USER-INTERFACE DESIGN AND EVALUATION REQUIRES UNDERSTANDING AND EXPERIENCE

But how valuable are user-interface design guidelines? That depends on who applies them to design problems.

Following user-interface design guidelines is not as straightforward as following cooking recipes. Design rules often describe goals rather than actions. They are purposefully very general to make them broadly applicable, but that means their exact meaning and applicability to specific design situations is open to interpretation.

Complicating matters further, more than one rule will often seem applicable to a given design situation. In such cases, the applicable design rules often conflict: they suggest different designs. This requires designers to determine which competing design rule is more applicable to the given situation and should take precedence.

Design problems, even without competing design guidelines, often have multiple conflicting goals—for example:

- bright screen *and* long battery life
- lightweight *and* sturdy
- multifunctional *and* easy to learn
- powerful *and* simple
- high resolution *and* fast loading
- WYSIWYG (what you see is what you get) *and* usable by blind people

Satisfying all the design goals for a computer-based product or service usually requires trade-offs—lots and lots of trade-offs. Finding the right balance point between competing design rules requires further tradeoffs.

Given all these complications, user-interface design rules and guidelines must be applied thoughtfully, not mindlessly, by people skilled in the art of user-interface design and/or evaluation. User-interface design rules and guidelines are more like *laws* than like *rote recipes*. Just as a set of laws is best applied and interpreted by lawyers and judges well versed in the laws, a set of user-interface design guidelines is best applied and interpreted by people who understand the basis for the guidelines and have learned from experience in applying them.

Unfortunately, with a few exceptions (e.g., Norman, 1983a), user-interface design guidelines are provided as simple lists of design edicts with little or no rationale or background.

Furthermore, although many early members of the user-interface design and usability profession had educations that included studying cognitive psychology, most newcomers to the field do not. That makes it difficult for them to apply user-interface design guidelines sensibly. Providing that rationale and background education is the focus of this book.

COMPARING USER-INTERFACE DESIGN GUIDELINES

Table I.1 places the two best-known user-interface guideline lists side by side to show the types of rules they contain and how they compare (see the Appendix for additional guideline lists). For example, both lists start with a rule calling for consistency in design. Both lists include a rule about preventing errors. The Nielsen–Molich rule to "help users recognize, diagnose, and recover from errors" corresponds closely to the Shneiderman–Plaisant rule to "permit easy reversal of actions." "User control and freedom" corresponds to "make users feel they are in control." There is a reason for this similarity, and it is not simply that later authors were influenced by earlier ones.

Table I.1 Two Best-Known Lists of User-Interface Design Guidelines

Shneiderman (1987); Shneiderman and Plaisant (2009)	Nielsen and Molich (1990)
Strive for consistency	Consistency and standards
Cater to universal usability	Visibility of system status
Offer informative feedback	Match between system and real world
Design task flows to yield closure	User control and freedom
Prevent errors	Error prevention
Permit easy reversal of actions	Recognition rather than recall
Make users feel they are in control	Flexibility and efficiency of use
Minimize short-term memory load	Aesthetic and minimalist design
	Help users recognize, diagnose, and recover from errors
	Provide online documentation and help

WHERE DO DESIGN GUIDELINES COME FROM?

For present purposes, the detailed design rules in each set of guidelines, such as those in Table I.1, are less important than what they have in common: their basis and origin. Where did these design rules come from? Were their authors—like clothing fashion designers—simply trying to impose their own personal design tastes on the computer and software industries?

If that were so, the different sets of design rules would be very different from each other as the various authors sought to differentiate themselves from others. In fact, all of these sets of user-interface design guidelines are quite similar if we ignore

differences in wording, emphasis, and the state of computer technology when each set was written. Why?

The answer is that all the design rules are based on human psychology: how people perceive, learn, reason, remember, and convert intentions into action. Many authors of design guidelines had at least some background in psychology that they applied to computer system design.

For example, Don Norman was a professor, researcher, and prolific author in the field of cognitive psychology long before he began writing about human–computer interaction. Norman's early human–computer design guidelines were based on research—his own and others'—on human cognition. He was especially interested in cognitive errors that people often make and how computer systems can be designed to lessen or eliminate the impact of those errors.

Similarly, other authors of user-interface design guidelines—for example, Brown, Shneiderman, Nielsen, and Molich—used knowledge of perceptual and cognitive psychology to try to improve the design of usable and useful interactive systems.

Bottom line: User-interface design guidelines are based on human psychology.

By reading this book, you will learn the most important aspects of the psychology underlying user-interface and usability design guidelines.

INTENDED AUDIENCE OF THIS BOOK

This book is intended mainly for software design and development professionals who have to apply user-interface and interaction design guidelines. This includes interaction designers, user-interface designers, user-experience designers, graphic designers, and hardware product designers. It also includes usability testers and evaluators, who often refer to design heuristics when reviewing software or analyzing observed usage problems.

A second important audience is students of interaction design and human–computer interaction. In fact, the second edition of this book turned out to be a popular textbook for college-level courses on UI/UX design. Because of that, one of my goals in updating and refining the book to produce this third edition has been to make it a better textbook.

A third intended audience is software development managers who want enough of a background in the psychological basis of user-interface design rules to understand and evaluate the work of the people they manage.

CHAPTER

Our Perception is Biased

1

Our perception of the world around us is not a true depiction of what is actually there. Our perceptions are heavily biased by at least three factors:

- ***The past:*** our experience
- ***The present:*** the current context
- ***The future:*** our goals

PERCEPTION BIASED BY EXPERIENCE

Experience—your past perceptions—can bias your current perception in several different ways.

Perceptual priming

Imagine that you own a large insurance company. You are meeting with a real estate manager, discussing plans for a new campus of company buildings. The campus consists of a row of five buildings, the last two with T-shaped courtyards providing light for the cafeteria and fitness center. If the real estate manager showed you the map in Fig. 1.1, you would see five black shapes representing the buildings.

Now imagine that instead of a real estate manager, you are meeting with an advertising manager. You are discussing a new billboard ad to be placed in certain markets around the country. The advertising manager shows you the same image, but in this scenario the image is a sketch of the ad, consisting of a single word: LIFE. In this scenario, you see a word, clearly and unambiguously.

When your perceptual system has been *primed* to see building shapes, you see building shapes, and the white areas between the buildings barely register in your

Designing with the Mind in Mind. https://doi.org/10.1016/B978-0-12-818202-4.00001-5
Copyright © 2021 Elsevier Inc. All rights reserved.

FIGURE 1.1

Building map or word? What you see depends on what you were told to see.

FIGURE 1.2

Image showing the effect of mental priming of the visual system. What do you see?

perception. When your perceptual system has been primed to see text, you see text, and the black areas between the letters barely register.

A relatively famous example of how priming the mind can affect perception is an image, supposedly by R. C. James,[1] that initially looks to most people like a random splattering of paint (see Fig. 1.2) similar to the work of the painter Jackson Pollack. Before reading further, look at the image.

Only after you are told that it is a Dalmatian dog sniffing the ground near a tree can your visual system organize the image into a coherent picture. Moreover, once you have seen the dog, it is hard to go back to seeing just a random collection of spots.

[1] Published in Lindsay and Norman (1972), Figs. 3.17, p. 146.

These priming examples are visual, but priming can also bias other types of perception, such as sentence comprehension. For example, the headline "New Vaccine Contains Rabies" would probably be understood differently by people who had recently heard stories about contaminated vaccines than by people who had recently heard stories about successful uses of vaccines to fight diseases.

Familiar perceptual patterns or frames

Much of our lives is spent in familiar situations: the rooms in our homes, our yards, our routes to and from school or work, our offices, neighborhood parks, stores, restaurants, etc. Repeated exposure to each type of situation builds a pattern in our minds of what to expect to see there. These *perceptual patterns*, which some researchers call *frames*, include the objects or events usually encountered in a particular situation.

For example, you know most rooms in your home well enough that you need not constantly scrutinize every detail. You know their layout and where most objects are located. You can probably navigate much of your home in total darkness. But your experience with homes is broader than your specific home. In addition to having a pattern for your home, your brain has one for homes in general. It biases your perception of all homes, familiar and new. In a kitchen, you expect to see a stove and a sink. In a bathroom, you expect to see a toilet, sink, and shower or bathtub (or both).

Our mental frames for situations bias our perception toward seeing the objects and events expected in each situation. They are a mental shortcut: by eliminating the need for us to constantly scrutinize every detail of our environment, they help us get around in our world. However, mental frames also make us see things that aren't really there.

For example, if you visit a house in which there is no stove in the kitchen, you might nonetheless later recall seeing one, because your mental frame for kitchens has a strong stove component. Similarly, part of the frame for eating at a restaurant is paying the bill, so you might recall paying for your dinner even if you absentmindedly walked out without paying. Your brain also has frames for backyards, schools, city streets, business offices, supermarkets, dentist visits, taxis, air travel, and other familiar situations.

Anyone who uses computers, websites, or smartphones has frames for the desktop and files, web browsers, websites, and various applications and online services. For example, when experienced Web users visit a new website, they expect to see a site name and logo, a navigation bar, some other links, and maybe a search box. When they book a flight online, they expect to specify trip details, examine search results, make a choice, and make a purchase. When they shop online, they expect a shopping cart and a checkout stage with a payment step.

Because users of computer software and websites have these perceptual frames, they often click buttons or links without looking carefully at them. Their perception of the display is based more on what their frame for the situation leads them to expect than on what is actually on the screen. This sometimes confounds software designers, who expect users to see what is on the screen—but that isn't how human vision and attention works.

For example, if the positions of the "Next" and "Back" buttons on the last page of a multistep screen sequence[2] are switched, many people would not immediately notice the change (see Fig. 1.3). Their visual system would have been lulled into inattention

[2] Multistep screen sequences are called *wizards* in user-interface design jargon.

FIGURE 1.3

Users may always perceive the Next button on the right, even when it isn't.

by the consistent placement of the buttons on the several pages that came before. Even after unintentionally going backward a few times by mistakenly clicking "Back" for "Next," they might continue to perceive the buttons in their standard locations. This is why consistent placement of controls is a recommended user-interface guideline, to ensure that reality matches the user's frame for the situation.

Similarly, if we are trying to find something but it is in a different place or looks different from usual, we might miss it even though it is in plain view because our mental frames tune us to look for expected features in expected locations. For example, if the "Submit" button on one form in a website is shaped differently or is a different color from those on other forms on the site, users might not find it. This expectation-induced blindness is discussed more later in this chapter in the "Perception Biased by Goals" section.

Habituation

A third way in which experience biases perception is called *habituation*. Repeated exposure to the same (or highly similar) perceptions dulls our perceptual system's sensitivity to them. Habituation is a very low-level phenomenon of our nervous system: it occurs at a neuronal level. Even primitive animals like flatworms and ameba, with very simple nervous systems, habituate to repeated stimuli (e.g., mild electric shocks or light flashes). People, with our complex nervous systems, habituate to a range of events, from low-level ones like a continually beeping tone, to medium-level ones like a blinking ad on a website, to high-level ones like a person who tells the same jokes at every party or a politician giving a long, repetitious speech.

We experience habituation in computer usage when the same error messages or "Are you sure?" confirmation messages appear again and again. People initially notice them and perhaps respond, but eventually they click them closed reflexively without bothering to read them.

Habituation is also a factor in a recent phenomenon variously labeled "social media burnout" (Nichols, 2013), "social media fatigue," or "Facebook vacations" (Rainie et al., 2013); newcomers to social media sites and tweeting are initially excited by the novelty of microblogging about their experiences, but sooner or later get tired of wasting time reading tweets about every little thing that their "friends" do or see—for example, "Man! Was that ever a great salmon salad I had for lunch today."

Attentional blink

Another low-level biasing of perception by past experience occurs just after we spot or hear something important. For a very brief period following the recognition—between 0.15 and 0.45 second—we are nearly deaf and blind to other visual stimuli, even though our ears and eyes stay functional. Researchers call this the *attentional blink* (Raymond et al., 1992; Stafford and Webb, 2005).[3] It is thought to be caused by the brain's perceptual and attention mechanisms being briefly fully occupied with processing the first recognition.

A classic example: You are in a subway car as it enters a station, planning to meet two friends at that station. As the train arrives, your car passes one of your friends, and you spot him briefly through your window. In the next split second, your window passes your other friend, but you fail to notice her because her image hit your retina during the attentional blink that resulted from your recognition of your first friend.

When people use computer-based systems and online services, attentional blink can cause them to miss information or events if things appear in rapid succession. A popular modern technique for making documentary videos is to present a series of still photographs in rapid succession.[4] This technique is highly prone to attentional blink effects—if an image really captures your attention (e.g., it has a strong meaning for you), you will probably miss one or more of the immediately following images. In contrast, a captivating image in an auto-running slideshow (e.g., on a website or an information kiosk) is unlikely to cause attentional blink (i.e., missing the next image) because each image typically remains displayed for several seconds.

PERCEPTION BIASED BY CURRENT CONTEXT

When we try to understand how our visual perception works, it is tempting to think of it as a bottom-up process, combining basic features such as edges, lines, angles, curves, and patterns into figures and ultimately meaningful objects. To take reading as an example, you might assume that our visual system first recognizes shapes as letters and then combines letters into words, words into sentences, and so on.

[3] Chapter 14 discusses the attentional blink interval along with other perceptual intervals.
[4] For an example, search YouTube for "history of the world in 2 minutes."

But visual perception—reading in particular—is not strictly a bottom-up process. It includes top-down influences too. For example, the word in which a character appears may affect how we identify the character (see Fig. 1.4).

Similarly, our overall comprehension of a sentence or a paragraph can even influence what words we see in it. For example, the same letter sequence can be read as different words depending on the meaning of the surrounding paragraph (see Fig. 1.5).

Contextual biasing of vision need not involve reading. The Müller-Lyer illusion is a famous example (see Fig. 1.6): the two horizontal lines are the same length, but the outward-pointing "fins" cause our visual system to see the top line as longer than the line with inward-pointing "fins." This and other optical illusions (see Fig. 1.7) trick us because our visual system does not use accurate, optimal methods to perceive the world. It developed through evolution, a semirandom process that layers jury-rigged—often incomplete and inaccurate—solutions on top of each other. It works fine most of the time but includes many approximations, kludges, hacks, and outright "bugs" that cause it to fail in certain cases.

FIGURE 1.4

The same character is perceived as H or A depending on the surrounding letters.

Fold napkins. *Polish silverware*. Wash dishes.

French napkins. *Polish silverware*. German dishes.

FIGURE 1.5

The same phrase is perceived differently depending on the list it appears in.

FIGURE 1.6

Müller-Lyer illusion: equal-length horizontal lines appear to have different lengths.

FIGURE 1.7

(A) The checkerboard does not bulge in the middle; (B) the triangle sides are not bent; and (C) the horizontal blue bars are horizontal, straight, and parallel. (Copyright © Victoria Skye, victoriaskye.com. Used by permission.)

The examples in Figs. 1.6 and 1.7 show vision being biased by *visual* context. However, biasing of perception by the current context works *between* different senses too. Perceptions in any of our five senses may affect simultaneous perceptions in any of our other senses. What we feel with our tactile sense can be biased by what we hear, see, or smell. What we see can be biased by what we hear, and what we hear can be biased by what we see. The following two examples of visual perception affect what we hear:

- ***McGurk effect.*** If you watch a video of someone saying "bah, bah, bah," then "dah, dah, dah," then "vah, vah, vah," but the audio is "bah, bah, bah" throughout, you will hear the syllable indicated by the speaker's lip movement rather than the syllable actually in the audio track.[5] Only by closing or averting your eyes do you hear the syllable as it really is. I will bet you did not know you could read lips, and in fact do so many times a day.

[5] See youtube.com/watch?v=PWGeUztTkRA, or search YouTube for "McGurk effect."

- ***Ventriloquism.*** Ventriloquists don't throw their voice; they just learn to talk without moving their mouths much. Viewers' brains perceive the talking as coming from the nearest moving mouth: that of the ventriloquist's puppet (Eagleman, 2012).

An example of the opposite—hearing biasing vision—is the illusory flash effect. When a spot is flashed *once* briefly on a display but is accompanied by *two* quick beeps, it appears to flash twice. Similarly, the perceived rate of a blinking light can be adjusted by the frequency of a repeating click (Eagleman, 2012).

Later chapters explain how visual perception, reading, and recognition function in the human brain. For now, I will simply say that the pattern of neural activity that corresponds to recognizing a letter, a word, a face, or any object includes input from neural activity stimulated by the context. This context includes other nearby perceived objects and events and even reactivated memories of previously perceived objects and events.

Context biases perception not only in people but also in lower animals. A friend of mine often brought her dog with her in her car when running errands. One day, as she drove into her driveway, a cat was in the front yard. The dog saw it and began barking. My friend opened the car door and the dog jumped out and ran after the cat, which turned and jumped through a bush to escape. The dog dove into the bush but missed the cat. The dog remained agitated for some time afterward.

Thereafter, for as long as my friend lived in that house, whenever she arrived at home with her dog in the car, he would get excited, bark, jump out of the car as soon as the door was opened, dash across the yard, and leap into the bush. There was no cat, but that didn't matter. Returning home in the car was enough to make the dog see one—perhaps even smell one. However, *walking* home on foot, as the dog did after being taken for his daily walk, did not evoke the "cat mirage."

PERCEPTION BIASED BY GOALS

In addition to being biased by our past experience and the present context, our perception is influenced by our goals and plans for the future. Specifically, our goals:

- ***Guide*** our perceptual apparatus, so we sample what we need from the world around us;
- ***Filter*** our perceptions—things unrelated to our goals tend to be filtered out preconsciously, never registering in our conscious minds.

For example, when people navigate through software or a website, seeking information or a specific function, they don't read carefully. They scan screens quickly and superficially for items that seem related to their goal. They don't simply ignore items unrelated to their goals; they often don't even notice them.

To see this, glance at Fig. 1.8 and look for scissors, and then immediately flip back to this page. Try it now.

FIGURE 1.8

Toolbox: Are there scissors here?

Did you spot the scissors? Now, without looking back at the toolbox, can you say whether there is a screwdriver in the toolbox too?

Our goals filter our perceptions in other perceptual senses as well as in vision. A familiar example is the "cocktail party" effect. If you are conversing with someone at a crowded party, you can focus your attention to hear mainly what he or she is saying even though many other people are talking near you. The more interested you are in the conversation, the more strongly your brain filters out surrounding chatter. If you are bored by what your conversational partner is saying, you will probably hear much more of the conversations around you.

The effect was first documented in studies of air traffic controllers, who were able to carry on conversations with the pilots of their assigned aircraft even though many different conversations were occurring simultaneously on the same radio frequency, coming out of the same speaker in the control room (Arons, 1992). Research suggests that our ability to focus on one conversation among several simultaneous ones depends not only on our interest level in the conversation, but also on objective factors, such as the similarity of voices in the cacophony, the amount of general "noise" (e.g., clattering dishes or loud music), and the predictability of what your conversational partner is saying (Arons, 1992).

This filtering of perception by our goals is particularly true for adults, who tend to be more focused on goals than children are. Children are more stimulus-driven; their perception is less filtered by their goals. This characteristic makes them more distractible than adults, but it also makes them less biased as observers.

A parlor game demonstrates this age difference in perceptual filtering. It is similar to the Fig. 1.8 exercise. Most households have a catch-all drawer for kitchen implements or tools. From your living room, send a visitor to the room where the catch-all drawer is with instructions to fetch you a specific tool, such as measuring spoons or a pipe wrench. When the person returns with the tool, ask whether another specific tool was in the drawer. Most adults will not know what else was in the drawer. Children—if they can complete the task without being distracted by all the cool stuff in the drawer—will often be able to tell you more about what else was there.

Perceptual filtering can also be seen in how people navigate websites. Suppose I put you on the home page of New Zealand's University of Canterbury (see Fig. 1.9)

FIGURE 1.9

University of Canterbury website: navigating sites requires perceptual filtering.

and asked you to find information about financial support for postgraduate students in the computer science department. You would quickly scan the page for words that were in the goal I gave you: "departments," "scholarships," "computer science," or "postgraduate." If you spotted a link containing one or more of those words, you would probably click on it. If you are a "search" person, you might instead go to the search symbol (magnifying glass, top right), click it and enter words related to the goal, and click "Go."

Whether you browse or search, it is likely that you would leave the home page without noticing that you were randomly chosen to win $100 (bottom right). Why? Because that was not related to your *goal*.

What is the mechanism by which our current goals bias our perception? There are two:

- ***Influencing where we look.*** Perception is active, not passive. Think of your perceptual senses not as simply *filtering* what comes to you but rather as *reaching out* into the world and *pulling in* what you need to perceive. Your hands, your primary touch sensors, literally do this, but the rest of your senses do it too. You constantly move your eyes, ears, hands, feet, body, and attention to sample exactly the things in your environment that are most relevant to what you are doing or about to do (Ware, 2008). If you are looking on a website for a campus map, your

eyes and pointer-controlling hand are attracted to anything that might lead you to that goal. You more or less ignore anything unrelated to your goal.

- ***Sensitizing our perceptual system to certain features.*** When you are looking for something, your brain can prime your perception to be especially sensitive to features of what you are looking for (Ware, 2008). For example, when you are looking for a red car in a large parking lot, red cars will seem to pop out as you scan the lot, and cars of other colors will barely register in your consciousness, even though you do in some sense see them. Similarly, when you are trying to find your spouse in a dark, crowded room, your brain "programs" your auditory system to be especially sensitive to the combination of frequencies that make up his or her voice.

TAKING BIASED PERCEPTION INTO ACCOUNT WHEN DESIGNING

All these sources of perceptual bias of course have implications for user-interface design. Here are three.

Avoid ambiguity

Avoid ambiguous information displays, and test your design to verify that all users interpret the display in the same way. Where ambiguity is unavoidable, either rely on standards or conventions to resolve it, or prime users to resolve the ambiguity in the intended way.

For example, displays on digital devices often add drop-shadows to user-interface components to make them look raised in relation to the background surface (see Fig. 1.10). This appearance relies on a convention, familiar to most people who use digital devices, that the light source is at the top of the screen. If a technology user does not know this convention, it may be ambiguous to them whether the object is raised or sunken.

Be consistent

Place information and controls in consistent locations. Controls and data displays that serve the same function on different pages should be placed in the same position on each page on which they appear. They should also have the same color, text fonts, shading, and so on. This consistency allows users to spot and recognize them quickly.

Understand the goals

Users come to a system with goals they want to achieve. Designers should understand those goals. Realize that users' goals may vary and that their goals strongly influence what they perceive. Ensure that at every point in an interaction, the information users

FIGURE 1.10

Components on digital device screens are given drop-shadows so they are perceived to float above the background, but that perception depends on a convention that the light source is at the top.

need is available, is prominent, and maps clearly to a possible user goal so users will notice and use the information.

IMPORTANT TAKEAWAYS

- Human perception is not an accurate reflection of what is "out there" in the world. It is biased by our experience, the current context, and our goals.
- Past experience can bias our perception by "priming" our perceptual systems to detect certain objects and events as well as "priming" them *not* to detect other objects and events. Repeated perception of an event over a short interval can cause *habituation*, increasing the chances that we will miss later occurrences of the event. With long-term experience, we develop *frames* for familiar situations that make us perceive things that aren't there or miss things that are.
- Because our attention has limited capacity, when it is overloaded we can miss other objects and events. This is called *attentional blink.*
- Perception operates in two ways simultaneously:
  - Our perception of whole objects and events is based on our perception of the parts.
  - Our perception of the parts is based on our perception of the whole.

- Our perception of objects and events can be biased by our emotional state.
- Our perceptual system—particularly our visual system—includes evolutionary hacks and bugs that cause us to misperceive certain stimuli.
- Our goals and plans strongly influence what we pay attention to and therefore what we perceive.
- By following design guidelines based on how human perception works, designers can create applications, websites, and appliances that are a good fit with the people who will use them.

CHAPTER

Our Vision is Optimized to See Structure

2

Early in the twentieth century, a group of German psychologists sought to explain how human visual perception works. They observed and cataloged many important visual phenomena. One of their basic findings was that human vision is holistic—our visual system automatically imposes structure on visual input and is wired to perceive whole shapes, figures, and objects rather than disconnected edges, lines, and areas. The German word for "shape" or "figure" is *Gestalt,* so these theories became known as the *Gestalt principles of visual perception*.

Today's perceptual and cognitive psychologists regard the Gestalt theory of perception more as a *descriptive* framework than as an *explanatory* and *predictive* theory. Current theories of visual perception are based heavily on the neurophysiology of the eyes, optic nerve, and brain (see Chapters 4–7).

Not surprisingly, the findings of neurophysiological researchers support the observations of the Gestalt psychologists. We really are—along with other animals—"wired" to perceive our surroundings in terms of whole objects (Stafford and Webb, 2005; Ware, 2008). Consequently, the Gestalt principles are still valid—if not as a fundamental explanation of visual perception, at least as a framework for describing it. They also provide a useful basis for guidelines for graphic design and user-interface design (Soegaard, 2007).

For this book, the most important Gestalt principles are Proximity, Similarity, Continuity, Closure, Symmetry, Figure/Ground, and Common Fate. The following sections describe each principle and provide examples from both static graphic design and user-interface design.

GESTALT PRINCIPLE: PROXIMITY

The Gestalt principle of *Proximity* is that the relative distance between objects in a display affects our perception of whether and how the objects are organized into

Designing with the Mind in Mind. https://doi.org/10.1016/B978-0-12-818202-4.00002-7
Copyright © 2021 Elsevier Inc. All rights reserved.

FIGURE 2.1

Proximity: items that are closer appear grouped as rows (A) and columns (B).

FIGURE 2.2

Proximity: even dissimilar objects that are close together appear grouped.

groups and subgroups. Objects near each other (relative to others) appear grouped, while those farther apart do not.

In Fig. 2.1A, the stars are closer together horizontally than they are vertically, so we see three rows of stars, while the stars in Fig. 2.1B are closer together vertically than they are horizontally, so we see three columns.

Although the stars in Fig. 2.1 are similar looking, objects need not look similar for them to appear grouped when placed near each other. For example, in Fig. 2.2, the groups are defined by how close together the objects are, not by how they look.

The Proximity principle is useful when laying out control panels or data forms in software, websites, and electronic appliances. Designers who don't know about Proximity sometimes use group boxes and horizontal or vertical lines to separate groups of controls and data displays. For example, Outlook's Distribution List Membership dialogue box groups the **Add …**, **Remove**, and **Properties …** buttons together using Proximity but then associates that group with the listbox using an unnecessary

FIGURE 2.3

In Outlook's Distribution List Membership dialogue box, group boxes are used unnecessarily. Simple proximity—spacing—would be enough.

group-box widget (see Fig. 2.3). Even less necessary is the group-box labeled "Distribution list," since it contains only a single combo-box. "Group box around one item" is a common UI design blooper (Johnson, 2007).

Using Proximity, items on a display can be visually grouped simply by spacing them closer to each other than to other controls without using group-boxes or visible borders. Furthermore, Proximity can be applied hierarchically to define groups of subgroups. For example, in Firefox's keyboard text preferences dialogue box, the three checkboxes to control spell-checking, autocapitalization, and automatic addition of periods are grouped (see Fig. 2.4), and those are grouped with other controls as well as a table control. Graphic design experts recommend using Proximity to avoid visual clutter (Mullet and Sano, 1994), cut wasted ink or pixels that add no data to the presentation (Tufte, 2001), and reduce the amount of code needed to implement it.

Proximity also governs perception of the labels on controls. Too much space between a label and the item it labels, and people won't connect the label with the item. Conversely, if a label is too close to a different item, people may connect it with *that* item instead of the intended one. For example, poor spacing of radio button labels in a form at Delta.com (2015) could easily cause people to choose the wrong button (see Fig. 2.5A), whereas the spacing of radio button labels at United.com (2020) shows clearly which label goes with which button (see Fig. 2.5B).

FIGURE 2.4

In Firefox's keyboard text preferences dialogue box, controls are grouped using the Proximity principle without group boxes and borders.

FIGURE 2.5

Radio button labels: (A) Poor placement; (B) Good placement.

GESTALT PRINCIPLE: SIMILARITY

Another factor that affects our perception of grouping is expressed in the Gestalt principle of *Similarity*, which states that similar-looking objects appear grouped, all other things being equal. In Fig. 2.6, the slightly larger, "hollow" stars are perceived as a group.

Gmail uses similarity—**bold** versus nonbold text—to help users perceive unread emails as a distinct group from already-read ones (see Fig. 2.7A). Lyft's smartphone app uses similarity—car shapes—to let users see at a glance how many of its drivers are available in a potential rider's vicinity (see Fig. 2.7B).

FIGURE 2.6

Similarity: items appear grouped if they look more similar to each other than to other objects.

FIGURE 2.7

Similarity is used: (A) by Gmail to make unread emails stand out from already-read ones and (B) by Lyft to provide a quick overview of available rides.

The Page Setup dialogue box in Mac OS applications uses both Similarity and Proximity to convey groupings (see Fig. 2.8). The two very similar and tightly spaced Orientation settings are intended to appear grouped. The two menus are not so tightly spaced but look similar enough that they appear grouped even though that probably wasn't intended. The Cancel and OK buttons are positioned together, away from everything else. Even without the separator line, they would appear to be a group.

FIGURE 2.8

Mac OS Page Setup dialogue box. Similarity and Proximity are used to group settings and controls.

GESTALT PRINCIPLE: CONTINUITY

Several Gestalt principles describe our visual system's tendency to resolve ambiguity or fill in missing data so we perceive whole objects. The first such principle, the principle of *Continuity,* states that when visual elements are aligned with each other, our visual perception is biased to perceive them as continuous forms rather than disconnected segments.

For example, in Fig. 2.9A, we automatically see two crossing lines—one blue and one orange. We don't see two separate orange segments and two separate blue ones,

FIGURE 2.9

Continuity: Human vision is biased to see continuous forms, even adding missing data if necessary.

and we don't see a blue-and-orange V on top of an upside-down orange-and-blue V. In Fig. 2.9B, due to the vertical alignment of the pieces and the fact that they are spaced to match the curvature of the visible pieces, we see a sea monster in water, not three pieces of one. If we misaligned the pieces or spaced the pieces further than the curvature suggests, the illusion of continuity would disappear.

A well-known example of the use of the continuity principle in graphic design is the IBM logo. It consists of disconnected blue patches, and yet it is not at all ambiguous. The blue rectangles are stacked vertically with horizontal space between the stacks, so we see three bold letters, perhaps viewed through something like venetian blinds (see Fig. 2.10).

Slider controls are a user-interface example of the Continuity principle. We see a slider as depicting a single range controlled by a handle that appears somewhere on the slider, not as two separate ranges separated by the handle (see Fig. 2.11A). Even displaying different colors on each side of a slider's handle doesn't completely "break" our perception of a slider as one continuous object, although ComponentOne's choice of strongly contrasting colors (gray vs. red) certainly strains that perception a bit (see Fig. 2.11B).

FIGURE 2.10

The IBM company logo uses the Continuity principle to form letters from disconnected patches.

FIGURE 2.11

Continuity: we see a slider as a single slot with a handle somewhere on it, not as two slots separated by a handle: (A) Mac OS and (B) ComponentOne.

GESTALT PRINCIPLE: CLOSURE

Related to Continuity is the Gestalt principle of *Closure*: our visual system automatically tries to close open figures so they are perceived as whole objects rather than separate pieces. Thus, we perceive the disconnected arcs in Fig. 2.12A as a circle.

Our visual system is so strongly biased to see objects that it can even interpret a totally blank area as an object. We see the combination of shapes in Fig. 2.12B as a white triangle overlapping another triangle and three black circles, even though the figure really only contains three V shapes and three black Pac-Men.

The Closure principle is often applied in graphical user interfaces (GUIs). For example, GUIs often represent collections of objects (e.g., documents or messages) as *stacks* (see Fig. 2.13). Just showing one whole object and the edges of others "behind" it is enough to make users perceive a stack of objects, all whole.

FIGURE 2.12

Closure: Human vision is biased to see whole objects, even when they are incomplete.

FIGURE 2.13

Desktop icons depicting stacks of objects exhibit the Closure principle: partially visible objects are perceived as whole.

GESTALT PRINCIPLE: SYMMETRY

A third fact about our tendency to see objects is captured in the Gestalt principle that in German is named "prägnanz," which literally means "good Figure" but is often translated to Simplicity or *Symmetry*. It states that we tend to parse complex scenes in a way that reduces the complexity. The data in our visual field usually has more than one possible interpretation, but our vision automatically organizes and interprets the data to simplify it and give it symmetry, making it easier to comprehend.

For example, we see the complex shape on the far left of Fig. 2.14 as two overlapping diamonds, not as two touching corner bricks or a pinch-waist octahedron with a square in its center. A pair of overlapping diamonds is simpler than the other two interpretations shown on the right—it has fewer sides and more symmetry than the other two interpretations.

The Symmetry principle also predicts that we will see Fig. 2.15 as five overlapping rings and not as a jumble of interconnected arcs.

FIGURE 2.14

Symmetry: the human visual system tries to resolve complex scenes into combinations of simple, symmetrical shapes.

FIGURE 2.15

Symmetry predicts that people will perceive this figure as five overlapping rings.

Our visual system's reliance on symmetry can be exploited to make complex information easier to scan and understand. For example, presenting information in a table—a symmetric way to present data—makes it easier to extract the desired information (see Table 2.1).

Table 2.1 Tables Use Symmetry to Make Data Easier to Scan Than a Prose Presentation Would.

Student	Quiz 1	Quiz 2	Quiz 3	Quiz 4
Fred (B)	95	92	98	90
Susan H.	99	98	97	95
Sergei L.	83	91	92	88
Hannah N.	75	87	92	83

GESTALT PRINCIPLE: FIGURE/GROUND

The next Gestalt principle that describes how our visual system structures the data it receives is *Figure/Ground*. This principle states that our mind separates the visual field into the figure (the foreground) and ground (the background). The foreground consists of the elements of a scene that are the object of our primary attention, and the background is everything else.

The Figure/Ground principle also specifies that the visual system's parsing of scenes into figure and ground is influenced by characteristics of the scene. For example, when a small object or color patch overlaps a larger one, we tend to perceive the smaller object as the figure and the larger object as the ground (see Fig. 2.16).

FIGURE 2.16

Figure/Ground: when objects overlap, we see the smaller as the figure and the larger as the ground.

However, our perception of figure versus ground is not completely determined by scene characteristics. It also depends on the viewer's focus of attention, as illustrated by Fig. 2.17. Is it a vase or two faces?

In user-interface and Web design, the Figure/Ground principle is often used to place an impression-inducing background "behind" the primary displayed content. The background can convey information, such as where the user is in the system as in the Android desktop in Fig. 2.18, or it can suggest a theme, brand, or mood for interpretation of the content.

FIGURE 2.17

Vase or faces? Perception of figure versus ground depends on a viewer's focus of attention.

FIGURE 2.18

Figure/Ground is used in mobile phones, tablets, and computers to display a device's "home" or "desktop" screen.

FIGURE 2.19

Figure/Ground can be used to display temporary information "over" the page content: (A) call-to-action at PBS.org's mobile website and (B) Android settings pulldown.

Figure/Ground is also often used to pop up information over other content. Content that was formerly the figure—the focus of the user's attention—temporarily becomes the *background* for new information, which appears briefly as the new *figure* (see Fig. 2.19). This approach is usually better than temporarily *replacing* the old information with the new information, because it provides context that helps keep people oriented regarding their place in the interaction.

GESTALT PRINCIPLE: COMMON FATE

The previous six Gestalt principles concerned perception of static (unmoving) figures and objects. One final Gestalt principle—Common Fate—concerns moving objects. The Common Fate principle is related to the Proximity and Similarity principles—like them, it affects whether we perceive objects as grouped. The Common Fate principle states that objects that move together are perceived as grouped or related.

For example, in a display showing dozens of pentagons, if seven of them wiggled in synchrony, people would see them as a related group, even if the wiggling pentagons were separated from each other and looked no different from all the other pentagons (see Fig. 2.20).

Common motion—implying common fate—is used in some animations to show relationships between entities. For example, Google's Gapminder graphs animate dots representing nations to show changes in various factors of economic development over time. Countries that move together share development histories (see Fig. 2.21).

FIGURE 2.20

Common Fate: items appear grouped or related if they move together.

FIGURE 2.21

Common fate: Gapminder animates dots to show which nations have similar development histories (for details, animations, and videos, visit Gapminder.org).

GESTALT PRINCIPLES: COMBINED

Of course, in real-world visual scenes, the Gestalt principles work in concert, not isolation. For example, a typical Mac OS desktop usually exemplifies six of the seven principles described here, excluding Common Fate): Proximity, Similarity, Continuity, Closure, Symmetry, and Figure/Ground (see Fig. 2.22). On a typical desktop, Common Fate is used (along with Similarity) when a user selects several files or folders and drags them as a group to a new location (see Fig. 2.23).

With all these Gestalt principles operating at once, a design may imply *unintended* visual relationships. A recommended practice after designing a display is to view it with each of the Gestalt principles in mind—Proximity, Similarity, Continuity, Closure, Symmetry, Figure/Ground, and Common Fate—to see if the design suggests any relationships between elements that you do *not* intend.

FIGURE 2.22

All of the Gestalt principles except Common Fate play a role in the Mac OS desktop.

FIGURE 2.23

Similarity and Common Fate: when users drag folders that they have selected, common highlighting and motion make the selected folders appear grouped.

IMPORTANT TAKEAWAYS

For easy reference, here are the Gestalt principles of visual perception covered in this chapter, which are those most relevant to user-interface design:

- **Proximity**: Objects near each other (relative to other objects) appear grouped, while those farther apart do not.
- **Similarity**: Objects that look similar appear grouped.
- **Continuity**: Our visual perception is biased to perceive continuous forms rather than disconnected segments.
- **Closure**: Our visual system automatically tries to close figures that are open so they are perceived as whole objects rather than separate pieces.
- **Symmetry**: Our visual system parses complex scenes in a way that reduces their complexity by recognizing symmetries in the scene.
- **Figure/Ground**: Our mind separates the visual field into the figure (the foreground) and ground (the background).
- **Common Fate**: Objects that move together are perceived as grouped or related.

CHAPTER

We Seek and Use Visual Structure

3

Chapter 2 showed how our visual system is optimized to perceive structure according to the Gestalt principles of visual perception. Perceiving structure in our environment helps us make sense of objects and events quickly. Chapter 2 also explained that when people are navigating through software or websites, they don't scrutinize screens carefully and read every word. They scan quickly for information relevant to their goals. This chapter (1) shows that when information is presented in a terse, structured way, it is easier for people to scan and understand and (2) explains *how* best to structure information.

STRUCTURED INFORMATION IS EASIER TO SCAN

Consider two presentations of the same information about an airline flight reservation. The first presentation is unstructured prose text; the second is structured text in outline form (see Fig. 3.1). The structured presentation of the reservation can be scanned and understood much more quickly than the prose presentation.

Unstructured:

You are booked on United flight 237, which departs from Auckland at 14:30 on Tuesday 15 Oct and arrives at San Francisco at 11:40 on Tuesday 15 Oct.

Structured:

Flight: **United 237, Auckland → San Francisco**
Depart: **14:30 Tue 15 Oct**
Arrive: **11:40 Tue 15 Oct**

FIGURE 3.1

Structured presentation of airline reservation information is easier to scan.

Designing with the Mind in Mind. https://doi.org/10.1016/B978-0-12-818202-4.00003-9
Copyright © 2021 Elsevier Inc. All rights reserved.

Unstructured:

Here is what I plan to do this week: clear out the garage, pull up the weeds in the back yard, repaint the back fence, plant the garden, call parents, buy birthday gift for Susan.

Structured:

To do this week:

- **clear garage**
- **pull back yard weeds**
- **repaint back fence**
- **plant garden**
- **call parents**
- **buy birthday gift for Susan**

FIGURE 3.2

Structured (bulleted) list is easier to scan.

Driver License Renewals, Duplicates, and Changes

Immediate Attention:

You will need a federal compliant card such as a passport, military ID, or REAL ID to board a domestic flight or access some federal facilities beginning October 1, 2020. The DMV started offering REAL ID driver licenses and ID cards on January 22, 2018. A field office visit is required to apply for a REAL ID.

Renewal

- How to renew your driver license by mail
- How to renew your driver license by internet
- How to renew your driver license in person
- How to renew your instruction permit
- How to renew your identification (ID) card

FIGURE 3.3

Driver License Renewals, Duplicates, and Changes page of the California Department of Motor Vehicles (DMV) website (January 2020) buried important information in a wordy prose paragraph and repetitive list.

Next, consider two ways to present a list. As above, the first presentation is unstructured prose text; the second is structured using bullets (see Fig. 3.2). The bulleted list is faster to scan and understand than the prose paragraph. That would also be true if the list were numbered.

The more structured and brief the presentation of information, the more quickly and easily people can scan and comprehend it. Look at the Driver License Renewals, Duplicates, and Changes page from the California Department of Motor Vehicles (see Fig. 3.3). Even with a bulleted list, the wordy, repetitive links slow users down and "bury" the important words they need to see.

FIGURE 3.4

Possible (fictional) improvement of the "Driver License Renewals, Duplicates, and Changes" page of the California Department of Motor Vehicles (DMV) website, eliminating unneeded text and making it easier to scan.

Compare that with a less wordy, more structured hypothetical design that factors out needless repetition and marks as links only those words that represent options (see Fig. 3.4).The revision provides the same information and options as the website's actual page but is easier to scan.

Displaying search results is another situation in which structuring data and avoiding repetitive "noise" can improve people's ability to scan quickly and find what they seek. In 2006, search results at HP.com included so much repeated navigation data and metadata for each retrieved item that they were useless. Recent HP websites eliminate the repetition and structure the results, making them easier to scan and more useful (see Fig. 3.5).

Of course, for information displays to be easy to scan, it is not enough merely to make them terse, structured, and nonrepetitious. They must also conform to the rules of graphic design, some of which were presented in Chapter 2.

For example, a prerelease version of a mortgage calculator on a real estate website presented its results in a table that violated at least two important rules of graphic design (see Fig. 3.6A). First, people usually read (online or offline) from top to bottom, but the labels for calculated amounts were *below* their corresponding values. Second, the labels were just as close to the value below as to their own value, so proximity (see Chapter 2) could not be used to perceive that labels were grouped with their values. To understand this mortgage results table, users had to scrutinize it carefully and slowly figure out which labels went with which numbers.

The revised design, in contrast, allows users to perceive the correspondence between labels and values without conscious thought (see Fig. 3.6B).

FIGURE 3.5

In 2006, HP.com's site search produced repetitious, "noisy" results (A), but later (2019) was improved (B).

FIGURE 3.6

(A) Mortgage summary presented by a software mortgage calculator; (B) an improved design.

VISUAL HIERARCHY HELPS PEOPLE FIND RELEVANT INFORMATION

One of the most common ways to structure information presentations is to provide a visual hierarchy—an arrangement that:

- breaks the information into distinct sections and breaks large sections into subsections;
- labels each section and subsection prominently and in such a way that content is clearly identified;

(A)

Create a Clear Visual Hierarchy

Organize and prioritize the contents of a page by using size, prominence, and content relationships. Let's look at these relationships more closely. The more important a headline is, the larger its font size should be. Big bold headlines help to grab the user's attention as they scan the Web page. The more important the headline or content, the higher up the page it should be placed. The most important or popular content should always be positioned prominently near the top of the page, so users can view it without having to scroll too far. Group similar content types by displaying the content in a similar visual style, or in a clearly defined area.

(B)

Create a Clear Visual Hierarchy

Organize and prioritize the contents of a page by using size, prominence, and content relationships.

Let's look at these relationships more closely:

- **Size.** The more important a headline is, the larger its font size should be. Big bold headlines help to grab the user's attention as they scan the Web page.
- **Prominence.** The more important the headline or content, the higher up the page it should be placed. The most important or popular content should always be positioned prominently near the top of the page, so users can view it without having to scroll too far.
- **Content Relationships.** Group similar content types by displaying the content in a similar visual style, or in a clearly defined area.

FIGURE 3.7

Find the advice about prominence in each of these displays. Prose text format (A) forces users to read or skim everything. Visual hierarchy (B) lets users skip irrelevant content and go straight to the information most relevant to their goals.

- presents the sections and subsections as a hierarchy, with higher-level sections presented more strongly than lower-level ones.

A visual hierarchy allows people, when scanning information, to quickly separate what is relevant to their goals from what is irrelevant and to focus their attention on the relevant information. They find what they are looking for more quickly because they can easily skip everything else.

Try it for yourself. Look at the two information displays in Fig. 3.7 and find the information about prominence. How much longer does it take you to find it in the nonhierarchical presentation?

The examples in Fig. 3.7 show the value of visual hierarchy in a textual, read-only information display. Visual hierarchy is equally important in interactive control panels and forms—perhaps even more so. Compare dialogue boxes from two different software products (see Fig. 3.8). The "Generate SoundTrack" dialogue box of the music app Band-in-a-Box has poor visual hierarchy, making it hard for users to scan it and find settings quickly. Adobe Acrobat Pro's "Print" dialogue box, by contrast, has better visual hierarchy, so users can quickly find the desired settings.

BEYOND VISUAL HIERARCHY: *INFORMATION* HIERARCHY

Visual hierarchy should actually be called "information hierarchy" because the basic design principle applies more broadly than just visual design. Organizing information hierarchically, using headings, subheadings, and other groupings to define sections

(A)

(B)

FIGURE 3.8

Visual hierarchy in interactive control panels and forms lets users find settings quickly: (A) Band-in-a-Box (2017—bad) and (B) Adobe Acrobat Pro (2019–good).

and subsections, can help even sight-impaired and blind users find what they are looking for.

Consider a blind person using a screen reader to search the screens shown in Fig. 3.7 for the information about "prominence" or trying to find a particular setting on the dialogue boxes shown in Fig. 3.8.

With Fig. 3.7A, blind users would have to command their screen reader to read methodically through most of the information before reaching the desired content, but with the information separated with headings into sections as in Fig. 3.7B, blind users can have their screen reader read just the section headings, skipping content in irrelevant sections. That would get them much more quickly to the information about "prominence."

Similarly, with Band-in-a-Box's "Generate SoundTrack" dialogue box (see Fig. 3.8A), blind users would have to laboriously tab through most of the settings to find the one they want, but with Acrobat Pro's "Print" dialogue box (see Fig. 3.8B), assuming the area headings are marked as headings, they could quickly find the relevant section, then look in it for the control they want.

To help Web developers and designers create information displays that are useable by people with visual impairments, the World Wide Web Consortium (W3C) created the Web Content Accessibility Guidelines (WCAG 2.0) (W3C, 2008a). The WCAG 2.0 guidelines advise designers to using cascading style sheets to mark text sections and subsections, and screen divisions and subdivisions, with appropriate-level headings (W3C, 2008b).

"CHUNKING" HELPS PEOPLE SCAN AND ENTER DATA

Even small amounts of data can be made easier to scan if they are structured. Two examples are telephone and credit card numbers (see Fig. 3.9). Such numbers should be broken into parts—often called "chunks"—to make them easier to scan and remember. The right size for chunks was determined by psychologists and human-factors researchers decades ago when telephone numbers and credit card numbers were introduced (e.g., Miller, 1956). It is based on the capacity of human short-term memory (Moran, 2016) (see also Chapter 7).

Hard: 4155551212
Easy: 415-555-1212

Hard: 1234567890123456
Easy: 1234 5678 9012 3456

FIGURE 3.9

Telephone and credit card numbers are easier to scan, understand, and remember when separated into "chunks."

FIGURE 3.10

(A) At KP.org (2020), credit card numbers cannot include spaces or other punctuation, making them hard to scan and verify. (B) At Amazon.com (2020), spaces are added automatically as users enter the number.

FIGURE 3.11

BankOfAmerica.com (2020) chunks phone numbers into three parts, making it impossible to enter an incorrectly formatted number.

Unfortunately, many of today's digital presentations of phone and credit card numbers neither chunk the numbers nor allow users to include spaces, hyphens, parentheses, or other punctuation (see Fig. 3.10A). This makes it harder for people to scan a number or verify that they typed it correctly and so is considered a user-interface design blooper (Johnson, 2007).

To be more user-friendly, a user interface should support chunking of numbers by letting users separate them into parts using spaces or other punctuation (see Fig. 3.10B).

Alternatively, designers can support chunking of long numbers by providing a separate field for each part of the number. Separate data fields can provide useful visual structure even when the data to be entered is not, strictly speaking, a number. Dates and phone numbers are an example of situations in which segmented fields can improve readability and help prevent data entry errors (see Fig. 3.11).

"Chunking" data helps people scan, comprehend, and recall not only long numbers but also other types of data. It makes data conform better to the limitations of human memory and attention, which are discussed more fully in Chapters 7 and 8.

EVEN MORE STRUCTURE FOR INPUT: DATA-SPECIFIC CONTROLS

A step-up in structure from segmented text fields is data-specific controls. Designers can combine text fields and other controls to create hybrid controls for getting values of specific types from users. With data-specific controls, users cannot enter invalid data, so one possible type of user error is eliminated.

For example, most airlines these days use a text field and a calendar control to let customers choose flight dates (see Fig. 3.12).

Structured data-specific controls can also be constructed from menus. For example, Southwest.com used Month, Day, and Year menus to build a date-of-birth control (see Fig. 3.13A). Menus can also be combined with text fields, as in Google's date-of-birth control (see Fig. 3.13B).

FIGURE 3.12

For getting flight dates, most airlines use a hybrid control: text field + calendar.
(A) AirNewZealand.com (B) mobile.Southwest.com.

FIGURE 3.13

Birthdate control: (A) built with menus on Southwest.com (2020); (B) built with menus + text fields on Google.com (2020).

IMPORTANT TAKEAWAYS

- Presenting information in a terse, structured manner makes it easier for people to scan, understand, and remember. Eliminate needless words and repetition. Use bullets and numbered lists instead of paragraphs of text.
- Visual hierarchy—breaking content up into clearly labeled sections and subsections—is a "best practice" way to structure information.
- Hierarchical structuring of information, using section headings and labels that screen readers can recognize as headings, helps people with visual impairments find their desired content.
- Chunking data—such as phone numbers, credit card numbers, serial numbers—helps people scan, enter, and remember it.

Used by permission, www.OKCancel.com.

CHAPTER

Our Attention is Limited; Our Memory is Imperfect

7

Using an app, website, or digital appliance requires controlling and focusing our *attention* and using our *memory*.

We focus *attention* on what we are trying to achieve. As we work, the app, website, or appliance produces results—visual, tactile, audible—that (hopefully) capture our attention and allow us to determine whether we have achieved or are approaching our goal.

Our *memory* comes into play in several ways. First, it allows us to *learn* to use the technology: every time we use a digital product or service, we apply what we *remember* from previous experience, reading instructions, or what another person told us, and we store new information. Our memory allows us to retain the *knowledge* of how to use the technology from one session to the next, and it allows us to increase our knowledge over time. In addition, our memory—augmented by feedback from the app, website, or appliance—allows us to keep track of our goal and what we have already done.

Because memory and attention are crucial components in people's use of digital technology, it is important for designers of digital products and services to know something about them. That is what this chapter is about.

Just as the human visual system has strengths and weaknesses, so do human attention and memory. This chapter describes some of those strengths and weaknesses as the background for understanding how to design interactive systems to support and augment attention and memory rather than burdening or confusing them. We will start with an overview of how memory works and how it is related to attention.

SHORT- VERSUS LONG-TERM MEMORY

Psychologists historically have distinguished *short-term* memory from *long-term* memory. Short-term memory covers situations in which information is retained for intervals ranging from a fraction of a second to a few minutes. Long-term memory

Designing with the Mind in Mind. https://doi.org/10.1016/B978-0-12-818202-4.00007-6
Copyright © 2021 Elsevier Inc. All rights reserved.

FIGURE 7.1

Traditional (antiquated) view of short-term versus long-term memory.

covers situations in which information is retained over longer periods (e.g., hours, days, years, or even lifetimes).

It is tempting to think of short- and long-term memory as separate memory stores. Indeed, some theories of memory have considered them separate. After all, in a digital computer, the short-term memory stores (central processing unit data registers) are separate from the long-term memory stores (random-access memory, hard disk, flash memory, CD-ROM, etc.). More direct evidence comes from findings that damage to certain parts of the human brain results in short-term but not long-term memory deficits, or vice versa. Finally, the speed with which information or plans can disappear from our immediate awareness contrasts sharply with the seeming permanence of our memory of important events in our lives, faces of significant people, activities we have practiced, and information we have studied. These phenomena led many researchers to theorize that short-term memory is a separate store in the brain where information is held temporarily after entering through our perceptual senses (e.g., visual or auditory) or after being retrieved from long-term memory (see Fig. 7.1).

A MODERN VIEW OF MEMORY

Recent research on memory and brain function indicates that short- and long-term memory are functions of a single memory system—one more closely linked with perception than previously thought (Jonides et al., 2008).

Long-term memory

Perceptions enter through the visual, auditory, olfactory, gustatory, or tactile sensory systems and trigger responses starting in areas of the brain dedicated to each sense (e.g., visual cortex, auditory cortex), then spread into other areas of the brain *not* specific to any particular sensory modality. The sensory modality-specific areas of the brain detect only simple features of the data, such as a dark/light edge, diagonal line, high-pitched tone, sour taste, red color, or rightward motion. Downstream areas of the brain combine low-level features to detect higher-level features of the input, such as animal, the word "duck," Uncle Kevin, minor key, threat, or fairness.

As described in Chapter 1, the set of neurons activated by a perceived stimulus depends on both the *features* and the *context* of the stimulus. Context is as important as features in determining what neural patterns are activated. For example, a dog

barking near you when you are walking in your neighborhood activates a different pattern of neural activity in your brain than the same sound heard when you are listening to a recording. The more similar that two perceptual stimuli are—that is, the more features and contextual elements they share—the more overlap there is between the sets of neurons that fire in response to them.

The initial strength of a perception depends on how much it is amplified or dampened by other brain activity. All perceptions create some kind of trace, but some are so weak that they can be considered not registered: the pattern was activated once but never again.

Memory formation consists of changes in the neurons involved in a neural activity pattern that make the pattern easier to reactivate in the future.[1] Some such changes result from chemicals released near neural endings that boost or inhibit their sensitivity to stimulation. These changes last only until the chemicals dissipate or are neutralized by other chemicals. More permanent changes occur when neurons grow and branch, forming new connections with others.

Activating a memory consists of reactivating the same pattern of neural activity that occurred when the memory was formed. Somehow the brain distinguishes initial activations of neural patterns from *re*activations—perhaps by measuring the relative ease with which the pattern was reactivated. New perceptions very similar to the original ones reactivate the same patterns of neurons, resulting in *recognition* if the reactivated perception reaches awareness. In the absence of a similar perception, stimulation from activity in other parts of the brain can also reactivate a pattern of neural activity, which if it reaches awareness results in *recall*.

The more often a neural memory pattern is reactivated, the stronger it becomes—that is, the easier it is to reactivate—which in turn means that the perception it corresponds to is easier to recognize and recall. Neural memory patterns can also be strengthened or weakened by excitatory or inhibitory signals from other parts of the brain.

A particular memory is not located in any specific spot in the brain. The neural activity pattern comprised in a memory involves a network of millions of neurons extending over a wide area. Activity patterns for different memories overlap depending on which features they share. Removing, damaging, or inhibiting neurons in a particular part of the brain typically does not completely wipe out memories that involve those neurons but rather just reduces the detail or accuracy of the memory by deleting features.[2] However, some areas in a neural activity pattern may be critical pathways, so removing, damaging, or inhibiting them may prevent most of the pattern from activating, thereby effectively eliminating the corresponding memory.

---

[1] There is evidence that the long-term neural changes associated with learning occur mainly during sleep, suggesting that separating learning experiences by periods of sleep facilitates learning (Stafford and Webb, 2005).

[2] This is similar to the effect of cutting pieces out of a holographic image: it reduces the overall resolution of the image, rather than removing areas of it as with an ordinary photograph.

For example, researchers have long known that the hippocampus, twin seahorse-shaped neural clusters near the base of the brain, plays an important role in storing long-term memories. The modern view is that the hippocampus is a controlling mechanism that directs neural rewiring to "burn" memories into the brain's wiring. The amygdala, two jellybean-shaped clusters on the frontal tips of the hippocampus, has a similar role, but it specializes in storing memories of emotionally intense, threatening situations (Eagleman, 2012).

Cognitive psychologists view human long-term memory as consisting of three distinct functions:

- ***Semantic*** long-term memory stores *facts and relationships*, such as "Germany is in Europe," "Berlin is the capital of Germany," "dogs are mammals," and "2 + 2 = 4."
- ***Episodic*** long-term memory records *past events*, such as the time you hit a baseball through a neighbor's window, the time you met your spouse, or yesterday's dinner.
- ***Procedural*** long-term memory remembers *action sequences*, such as how to tie your shoes, execute a karate move, or play a song on your guitar.

Everything we do, including using digital technology, uses all three types of long-term memory. For example, when you text a friend on your smartphone, you use semantic memory to generate what you want to say, episodic memory to recall what you and your friends did yesterday, and procedural memory to guide your hands in tapping out the text of your message and sending it. Furthermore, your phone's texting app *supports* all three types of memory. It supports semantic memory by helping you keep track of your friends and their likes and dislikes and by sending you facts to add to your knowledge; it supports episodic memory by showing you texts you have sent and received as well as by providing news about events, and it supports procedural memory—or perhaps hampers it—by autocompleting many of the words you type.

Short-term memory

So far, we've discussed long-term memory. What about short-term memory? What psychologists call short-term memory is actually a *combination* of phenomena involving perception, attention, and retrieval from long-term memory.

One component of short-term memory is perceptual. Each of our perceptual senses has its own very brief short-term "memory" resulting from residual neural activity after a perceptual stimulus ceases, like a bell that rings briefly after it is struck. Until they fade away, these residual perceptions are available as possible input to our brain's attention and memory-storage mechanisms, which integrate input from our various perceptual systems, focus our awareness on some of that input, and store some of it in long-term memory. These sensory-specific residual perceptions together make up a minor component of short-term memory. Here, we are only interested in them as potential inputs to working memory.

Also available as potential input to working memory are long-term memories reactivated through recognition or recall. As explained earlier, each long-term memory corresponds to a specific pattern of neural activity distributed across our brain. While activated, a memory pattern is a candidate for our attention and therefore potential input for working memory.

The human brain has multiple attention mechanisms, some voluntary and some involuntary. They focus our awareness on a very small subset of perceptions and activated long-term memories while ignoring everything else. The tiny subset of all available information from our perceptual systems and long-term memories that we are aware of *right now* is the main component of our short-term memory, the part that cognitive scientists often call *working memory*. It integrates information from all our sensory modalities and long-term memory. Henceforth, we will restrict our discussion of short-term memory to working memory.

So what is working memory? First, here is what it is *not*: it is not a store—it is not a *place* in the brain where memories and perceptions *go* to be worked on. And it is nothing like accumulators or fast random-access memory in digital computers.

Instead, working memory is the combined focus of our attention: everything that we are conscious of at a given time. More precisely, it is a few perceptions and long-term memories that are activated enough that we remain aware of them over a short period. Psychologists also view working memory as including an executive function—based mainly in the frontal cerebral cortex—that manipulates items we are attending to, and if needed refreshes their activation so they remain in our awareness (Baddeley, 2012).

A useful—if oversimplified—analogy for memory is a huge, dark, musty warehouse. The warehouse is full of long-term memories piled haphazardly (not stacked neatly), intermingled and tangled, and mostly covered with dust and cobwebs. Doors along the walls represent our perceptual senses: sight, hearing, smell, taste, touch. They open briefly to let perceptions in. As perceptions enter, they are briefly illuminated by light coming in from outside, but they quickly are pushed (by more entering perceptions) into the dark tangled piles of old memories.

In the ceiling of the warehouse are a small fixed number of searchlights controlled by the attention mechanism's executive function (Baddeley, 2012). They swing around and focus on items in the memory piles, illuminating them for a while until they swing away to focus elsewhere. Sometimes one or two searchlights focus on new items after they enter through the doors. When a searchlight moves to focus on something new, whatever it had been focusing on is plunged into darkness.

The small fixed number of searchlights represents the limited capacity of working memory. What is illuminated by them (and briefly through the open doors) represents the contents of working memory—out of the vast warehouse's entire contents, the few items we are attending to at any moment (see Fig. 7.2).

The warehouse analogy is too simple and should not be taken too seriously. As Chapter 1 explained, our senses are *not* just passive doorways into our brains through which our environment "pushes" perceptions. Rather, our brain actively and continually *seeks out* important events and features in our environment and

FIGURE 7.2

Modern view of memory: a dark warehouse full of stuff (long-term memory) with searchlights focused on a few items (short-term memory).

"pulls" perceptions in as needed (Ware, 2008). Furthermore, the brain is buzzing with activity most of the time, and its internal activity is only modulated—not determined—by sensory input (Eagleman, 2012). Also, as described earlier, memories are embodied as networks of neurons distributed around the brain, not as objects in a specific location. Finally, activating a memory in the brain can activate related ones; our warehouse-with-searchlights analogy doesn't represent that.

Nonetheless, the analogy—especially the part about the searchlights—illustrates that working memory is a *combination* of several *foci of* attention (the currently activated neural patterns of which we are aware), its capacity is extremely limited, and its content at any given moment is very volatile.

What about the earlier finding that damage to some parts of the brain causes short-term memory deficits, while other types of brain damage cause long-term memory deficits? The current interpretation is that some types of damage decrease or eliminate the brain's ability to focus attention on specific objects and events, while other types of damage harm the brain's ability to store or retrieve long-term memories.

CHARACTERISTICS OF ATTENTION AND WORKING MEMORY

As noted, working memory is equal to the focus of our attention. Whatever is in that focus is what we are conscious of at any moment. But what determines what we attend to and how much we can attend to at any given time?

Attention is highly focused and selective

Most of what is going on around you at this moment you are unaware of. Your perceptual system and brain sample very selectively from your surroundings because they don't have the capacity to process everything.

Right now you are conscious of the last few words and ideas you have read but probably not the color of the wall in front of you. But now that I have shifted your attention, you *are* conscious of the wall's color and may have forgotten some of the ideas you read on the previous page.

Chapter 1 described how our perception is filtered and biased by our goals. For example, if you are looking for your friend in a crowded shopping mall, your visual system "primes" itself to notice people who look like your friend (including how he or she is dressed), and barely notice everything else. Simultaneously, your auditory system primes itself to notice voices that sound like your friend's voice and even footsteps that sound like those of your friend. Human-shaped blobs in your peripheral vision and sounds localized by your auditory system that match those of your friend snap your eyes and head toward them. While you look, anyone looking or sounding similar to your friend attracts your attention, and you won't notice other people or events that would normally have interested you.

Besides focusing on objects and events related to our current goals, our attention is drawn to:

- ***Movement, especially movement near or toward us***. For example, something jumps at you while you walk on a street, or something swings toward your head in a haunted house ride at an amusement park, or a car in an adjacent lane suddenly swerves toward your lane (see the discussion of the *flinch reflex* in Chapter 14).
- ***Threats***. Anything that signals or portends danger to us or people in our care.
- ***Faces of other people***. We are primed from birth to notice faces more than other objects in our environment.
- ***Sex and food***. Even if we are happily married and well fed, these things attract our attention. Even the mere words probably got your attention.

These things, along with our current goals, draw our attention involuntarily. We don't become aware of something in our environment and then orient ourselves toward it. It is the other way around: our perceptual system detects something attention-worthy and orients us toward it preconsciously, and only afterward do we become aware of it.[3]

[3] Exactly *how long* afterward is discussed in Chapter 14.

Working memory (aka attention) is limited

The primary characteristics of working memory are its low capacity and volatility. But what is that capacity? In terms of the warehouse analogy presented earlier, what is *the small fixed number* of searchlights?

Many college-educated people have read about "the magical number seven, plus or minus two" proposed by cognitive psychologist George Miller in 1956 as the limit on the number of simultaneous unrelated items in human working memory (Miller, 1956).

Miller's characterization of the working-memory limit naturally raises several questions:

- ***What are the items in working memory?*** They are current perceptions and retrieved memories. They are goals, numbers, words, names, sounds, images, odors—anything one can be aware of. In the brain, they are patterns of neural activity.
- ***Why must items be unrelated?*** Because if two items are related, they correspond to one big neural activity pattern—one set of features—and hence one item, not two.
- ***Why the fudge factor of plus or minus two?*** Because researchers cannot measure with perfect accuracy how much people can keep track of as well as individual differences in working-memory capacity.

Later research in the 1960s and 1970s found Miller's estimate to be too high. In the experiments Miller considered, some of the items presented to people to remember could be "chunked" (i.e., considered related), making it appear that people's working memories were holding more items than they actually were. Furthermore, all the subjects in Miller's experiments were college students. Working-memory capacity varies within the general population. When the experiments were revised to disallow unintended chunking and include noncollege students as subjects, the average capacity of working memory was shown to be more like four plus or minus one—that is, three to five items (Broadbent, 1975; Mastin, 2010). Thus, in our warehouse analogy, there would be only four searchlights.

More recent research has cast doubt on the idea that the capacity of working memory should be measured in whole items or "chunks." It turns out that in early working-memory experiments, people were asked to briefly remember items (e.g., words or images) that were quite different from each other—that is, they had very few features in common. In such a situation, people don't have to remember every feature of an item to recall it a few seconds later; remembering some of its features is enough. So people appeared to recall items as a whole, and therefore working-memory capacity seemed measurable in whole items.

Recent experiments have given people items to remember that are similar—that is, they share many features. In that situation, to recall an item and not confuse it with other items, people must remember more of its features. In these experiments, researchers found that people remember more details (i.e., features) of some items than of others, and the items they remember in greater detail are the ones they paid more attention to (Bays and Husain, 2008). This suggests that the unit of attention—and therefore the capacity of working memory—is best measured in item features rather than whole items or "chunks" (Cowan et al., 2004). This jibes with the modern view of the brain as a feature-recognition device, but it is controversial among memory researchers, some of whom argue that the basic capacity of human working memory is three to five whole items but is reduced when people attend to a large number of the details (i.e., features) of items (Alvarez and Cavanagh, 2004).

Bottom line: The true capacity of human working memory is still a research topic.

The second important characteristic of working memory is how volatile it is. Cognitive psychologists used to say that new items arriving in working memory often bump old ones out, but that way of describing the volatility is based on the view of working memory as a temporary storage place for information. The modern view of working memory as the current focus of attention makes it even clearer: focusing attention on new information turns it away from some of what it was focusing on. That is why the searchlight analogy is useful.

However we describe it, information can easily be lost from working memory. If items in working memory don't get combined or rehearsed, they are at risk of having the focus shifted away from them. This volatility applies to goals as well as to the details of objects. Losing items from working memory corresponds to forgetting or losing track of something you were doing. We have all had such experiences, for example:

- Going to another room for something, but once there we cannot remember why we came.
- Taking a phone call and afterward not remembering what we were doing before the call.
- Something yanks our attention away from a conversation and then we can't remember what we were talking about.
- In the middle of adding a long list of numbers, something distracts us, so we have to start over.

WORKING MEMORY TEST

To test your working memory, copy these instructions onto a separate sheet of paper, get a pen or pencil and two blank sheets of paper, and follow these instructions:

1. Flip to the next page of the book but don't look at it. Cover it with a blank sheet of paper.
2. Pull the paper cover down a little, read the **black numbers** at the top for a few seconds, then slide the blank sheet back up to cover the book page again. *Don't peek ahead at other numbers on that page unless you want to ruin the test.* Look back at these instructions.
3. Say your phone number backward, out loud.
4. Now write down the black numbers from memory … now check the black numbers again. Did you get all of them?
5. Pull the cover down a bit more, read the **red numbers** (under the black ones), and look back at these instructions.
6. Write down the numbers from memory. These would be easier to recall than the first ones if you noticed that they are the first seven digits of π (3.141,592), because then they would be only one number, not seven.
7. Return to the book page, read the **green numbers** for 3 seconds, then look back here.
8. Write down the numbers you remember. If you noticed that they are odd numbers from 1 to 13, they will be easier to recall because you only need to remember three chunks ("odd #s, 1 to 13" or "seven odd #s, from 1"), not seven.
9. Return to the book page, read the **orange words** for 3 seconds, then look back here.
10. Write down the words from memory … could you recall them all?
11. Return to the book page, read the **blue words** for 3 seconds, then look back here.
12. Write down the words from memory … it was a lot easier to recall them because they form a sentence, so they could be stored as one sentence rather than seven words.

3 8 4 7 5 3 9

3 1 4 1 5 9 2

1 3 5 7 9 11 13

town river corn string car shovel

what is the meaning of life

IMPLICATIONS OF WORKING-MEMORY CHARACTERISTICS FOR USER-INTERFACE DESIGN

The capacity and volatility of working memory have many implications for the design of interactive computer systems. The basic implication is that user interfaces should help people remember essential information from one moment to the next. Don't require people to remember system status or what they have done, because their attention is focused on their primary goal and progress toward it. Specific examples follow.

Voice user interfaces

Voice user interfaces (VUIs) place a higher load on short-term memory than graphical user interfaces do (Pearl, 2018a). The most obvious design implication of the limited capacity of human working memory is that computer systems that present information via speech output should limit how much information they provide at once so they don't overload users' short-term memory. This includes telephone voice-response systems like those that companies provide to help you before they let you speak with a real person. It also includes personal digital assistants like Siri, Alexa, Google Assistant, and Cortana.

For example, VUIs should *not* issue long instructions in a single long utterance. Bad: *"Welcome to ACME travel service, your gateway to anywhere. I can help you with flight reservations, hotel bookings, or car rentals. Just tell me which one*

you want. If you need help, say 'help'. *Unfortunately, our car rental service is not available now. What would you like?"* Few people will be able to remember all of that. Instead, get rid of nonessential text and break the interaction up into short segments. Better: *"ACME travel. Say* 'flight reservation', 'book hotel', *or* 'rent car'. *"* (User says "rent car.") "OK, you want to rent a car. In what city?" (etc.)

A related VUI design guideline is that verbal lists of options or search results should not be longer than four or five items. People can't remember more than that (Budiu and Laubheimer, 2018) unless the items are familiar, such as close friends, family members, or favorite restaurants (Pearl, 2018b).

Some VUIs run on devices that also have screens, such as smartphones, tablets, laptop computers, and some smart speakers. In such cases, the VUI can supplement its verbal output by displaying information on the screen, thus reducing what it needs to say and thereby decreasing the burden on users' short-term memory (Pearl, 2018a).

Modes

The limited capacity and volatility of working memory is one reason user-interface design guidelines often say to either avoid designs that have *modes* or provide adequate mode feedback. In a moded user interface, some user actions have different effects depending on what mode the system is in. For example:

- In a car, pressing the accelerator pedal can move the car either forward, backward, or not at all depending on whether the transmission is in drive, reverse, or neutral. The transmission sets a mode in the car's user interface.
- In many digital cameras, pressing the shutter button can either snap a photo or start a video recording depending on which mode is selected.
- In a drawing program, clicking and dragging normally selects one or more graphic objects on the drawing, but when the software is in "draw rectangle" mode, clicking and dragging adds a rectangle to the drawing and stretches it to the desired size.

Moded user interfaces have advantages; that is why many interactive systems have them. Modes allow a device to have more functions than controls: the same control provides different functions in different modes. Modes allow an interactive system to assign different meanings to the same gestures to reduce the number of gestures users must learn.

However, one well-known *dis*advantage of modes is that people often make *mode errors*: they forget what mode the system is in and do the wrong thing by mistake (Johnson, 1990). This is especially true in systems that give poor feedback about what mode the system is in. Because of the problem of mode errors, many user-interface design guidelines say to either avoid modes or provide strong feedback about which mode is the current one. Working memory is too unreliable for designers to assume that users can, without clear, continuous feedback, keep track of what mode the system is in, even when the users are the ones changing modes.

Search results

When people use a search function on a computer to find information, they enter the search terms, start the search, and then review the results. Evaluating the results often

requires knowing what the search terms were. If working memory were less limited, people would always remember, when browsing the results, what they had entered as search terms just a few seconds earlier. But as we have seen, working memory is very limited. When the results appear, a person's attention naturally turns away from what he or she entered and toward the results. Therefore, it should be no surprise that people viewing search results often do not remember the search terms they just typed.

Unfortunately, some designers of online search functions don't understand that. Search results sometimes do not show the search terms that generated the results. For example, in 2006, the search results page at Slate.com provided search fields so users could search again, but didn't show what a user had searched for (see Fig. 7.3A). The 2020 version of the site shows the user's search terms (see Fig. 7.3B), reducing the burden on user working memory.

Calls to action

A well-known "netiquette" guideline for writing email messages, especially messages that require responses or ask the recipients to do something, is to restrict each message to one topic. If a message contains multiple topics or requests, its recipients may focus on one of them (usually the first one), get engrossed in responding to that, and forget to respond to the rest of the email. The guideline to put different topics or requests into separate emails is a direct result of the limited capacity of human attention.

Web designers are familiar with a similar guideline: avoid putting competing calls to action on a page. Each page should have only *one* dominant call to action—or one for each possible user goal—to not overwhelm users' attention capacity and cause them go down paths that don't achieve their (or the site owner's) goals. A related guideline: once users have specified their goal, don't distract them from accomplishing it by displaying extraneous links and calls to action. Instead, guide them to the goal by using a design pattern called the *process funnel* (van Duyne et al., 2002; see also Johnson, 2007).

Navigation depth

Using a software product, digital device, phone menu system, or website often involves navigating to the user's desired information or goal. It is well established that broad, shallow navigation hierarchies are easier for most people—especially those who are nontechnical—to find their way around in than narrow, deep hierarchies (Cooper, 1999). This applies to hierarchies of application windows and dialogue boxes as well as to menu hierarchies (Johnson, 2007).

A related guideline: in hierarchies deeper than two levels, provide navigation "bread crumb" paths to constantly remind users where they are (Nielsen, 1999; van Duyne et al., 2002).

These guidelines, like those mentioned earlier, are based on the limited capacity of human working memory. Requiring a user to drill down through eight levels of dialogue boxes, web pages, menus, or tables—especially with no visible reminders of their location—will probably exceed the user's working-memory capacity, thereby causing him or her to forget where he or she came from or what his or her overall goals were.

(A)

(B)

FIGURE 7.3

Slate.com search results: in 2007 (A), users' search terms were not shown, but in 2020 (B), search terms are shown.

CHARACTERISTICS OF LONG-TERM MEMORY

Long-term memory differs from working memory in many respects. Unlike working memory, it actually *is* a memory store.

However, specific memories are not stored in any one neuron or location in the brain. As described earlier, memories, like perceptions, consist of activation patterns of large sets of neurons, with subsets of neurons and individual neurons in each pattern encoding specific features of the memory. Related memories correspond to overlapping patterns of activated neurons, because similar memories share details—i.e., features. For example, my tablet computer is different from my smartphone, but my memories of how to read and send email on the two devices share features because the devices and the email apps on them are similar, so the neural firing patterns that encode both memories overlap quite a lot. Every memory is stored in a distributed fashion, spread among many parts of the brain. In this way, long-term memory in the brain is similar to holographic light images.

Long-term memory evolved to serve our ancestors and us very well in getting around in our world. However, it has many weaknesses: it is error-prone, impressionist, free-associative, idiosyncratic, retroactively alterable, and easily biased by a variety of factors at the time of recording or retrieval. Let's examine some of these weaknesses.

We will begin our examination of long-term memory weaknesses by considering the capacity of human long-term memory. Unlike working memory, the capacity of human long-term memory seems almost unlimited. No human has ever run out of "memory space." Why?

Before we answer that question, let's estimate how much information a human brain can hold. Adult human brains each contain about 86 billion neurons (Herculano-Houzel, 2009). If each neuron encodes 1 bit of memory, that suggests that the information capacity of the human brain is about 86 gigabits, which is 10.75 gigabytes. By that reckoning, the human brain has less memory than most mobile phones.

But wait: neurons can vary their firing rate, so maybe a neuron can represent several bits, which suggests a *larger* total information capacity. On the other hand, not all neurons in the brain are involved in storing memories, suggesting a *lower* total capacity. However, as described earlier, individual neurons do not store memories; memories are encoded by *networks* of neurons acting together. Even if only *some* of the brain's neurons are involved in memory, the large number of those neurons and the fact that each one can participate in many neural networks allows for an astronomical number of combinations of neurons into networks, with each network potentially encoding a different memory. By that reckoning, the maximum memory-capacity of the human brain is much, much greater than what we estimated above—perhaps terabytes. The overall result is that researchers have not yet measured or even provided a plausible estimate of the maximum information capacity of the human brain.[4]

[4] The closest researchers have come is Landauer's (1986) use of the average human learning rate to calculate the amount of information a person can learn in the average lifetime: 10^9 bits, or a few hundred megabytes; not much compared with today's computers, phones, and flash drives.

However, for practical purposes it doesn't matter what the maximum information capacity of the human brain may be at any instant in time, because the brain does not store information like a digital computer. New memories "steal" neurons from old memories (Eagleman, 2015). A new experience is never totally new; it shares features with prior experiences, so the network it activates shares neurons with the networks that encode those older experiences. Over time, a new memory is strengthened by repeated experience or practice, while related older memories lose details—researchers say that old memories lose *resolution*.

For example, if you don't see someone you know for months or years, your memory of their face slowly loses details—features—as your brain reuses some of the neurons that encode that memory to encode new face-memories. Similarly, as you learn to use a new computer or smartphone app, you will forget details of how to use related apps if you don't practice using them too. To paraphrase an old saying: "Old memories don't disappear; they just lose details."

Bottom line: nearly everything we've ever experienced is stored in our long-term memory but at different degrees of detail, and the detail of each memory can increase with practice and decrease with disuse.

Error-prone

Long-term memory is not an accurate, high-resolution recording of our experiences. In terms familiar to computer engineers, one could characterize long-term memory as using heavy compression methods that drop a great deal of information. Images, concepts, events, sensations, actions—all are reduced to combinations of features. Different memories are stored at different levels of detail—that is, with more or fewer features. Also, memories in the human brain can be changed by the retrieval process; that is, *reading* a memory also *writes* into it. That is very unlike memory in digital computers, in which reading and writing are distinct operations.

For example, the face of a man you met briefly who is not important to you might be stored simply as an average Caucasian male face with a beard with no other details—a whole face reduced to three features. If you were asked later to describe the man in his absence, the most you could honestly say was that he was a "white guy with a beard." You would not be able to pick him out of a police lineup of other Caucasian men with beards. In contrast, your memory of your best friend's face includes many more features, allowing you to give a more detailed description and pick your friend out of any police lineup. Nonetheless, it is still a set of features, not anything like a bitmap image.

As another example, I have a vivid childhood memory of being run over by a plow and badly cut, but my father claimed that it happened to my brother. One of us is wrong.

In the realm of human–computer interaction, a Microsoft Word user may remember that there is a command to insert a page number but may not remember which *menu* the command is in. That specific feature may not have been recorded when the user learned how to insert page numbers. Alternatively, perhaps the menu-location feature *was* recorded but just does not reactivate with the rest of the memory pattern when the user tries to recall how to insert a page number.

Weighted by emotions

Chapter 1 described a dog that remembered seeing a cat in his front yard every time he returned home in the family car. The dog was excited when he first saw the cat, so his memory of it was strong and vivid.

A comparable human example would be that an adult could easily have strong memories of her first day at nursery school but probably not of her tenth. On the first day, she was probably upset about being left at the school by her parents, whereas by the 10th day, being left there was nothing unusual.

Retroactively alterable

Suppose that while you are on an ocean cruise with your family, you see a whale shark. Years later, when you and your family are discussing the trip, you might remember seeing a whale, and one of your relatives might recall seeing a shark. For both of you, some details in long-term memory were dropped because they did not fit a common concept.

A true example comes from 1983, when the late President Ronald Reagan was speaking with Jewish leaders during his first term as president. He spoke about being in Europe during World War II and helping to liberate Jews from the Nazi concentration camps. The trouble was, he was never in Europe during World War II. When he was an actor, he was in a *movie* about World War II, made entirely in Hollywood. That important detail had been dropped from his memory.

A LONG-TERM MEMORY TEST

Test your long-term memory by answering the following questions:

1. Was there a roll of tape in the toolbox in Chapter 1?
2. What was your *previous* phone number?
3. Which of these words were *not* in the list presented in the working-memory test earlier in this chapter: city, stream, corn, auto, twine, spade?
4. What was your first-grade teacher's name? Second grade? Third grade?
5. What website was presented earlier that does not show search terms when it displays search results?

Regarding question 3: When words are memorized, often what is retained is the *concept* rather than the exact word presented. For example, one could hear the word "town" and later recall it as "city."

IMPLICATIONS OF LONG-TERM MEMORY CHARACTERISTICS FOR USER-INTERFACE DESIGN

The main thing that the characteristics of long-term memory imply is that people need tools to augment it. Since prehistoric times, people have invented technologies to help them remember things over long periods: notched sticks, knotted ropes, mnemonics, verbal stories and histories retold around campfires, writing, scrolls, books, number systems, shopping lists, checklists, phone directories, datebooks, accounting ledgers, oven timers, computers, portable digital assistants (PDAs), online shared calendars, etc.

Given that humankind has a need for technologies that *augment* memory, it seems clear that software designers should try to provide software that fulfills that need. At the very least, designers should avoid developing systems that *burden* long-term memory. Yet that is exactly what many interactive systems do.

Authentication is one functional area in which many software systems place burdensome demands on users' long-term memory. For example, a web application developed a few years ago told users to change their personal identification number (PIN) "to a number that is easy to remember" but then imposed restrictions that made it impossible to do so (see Fig. 7.4). Whoever wrote those instructions seems to have realized that the PIN requirements were unreasonable, because the instructions end by advising users to write down their PIN! Never mind that writing a PIN down creates a security risk and adds yet *another* memory task: users must remember where they hid their written-down PIN.

A contrasting example of burdening people's long-term memory for the sake of security comes from United.com. To make flight reservations, visitors must register. The site requires users to select a security question from a menu (see Fig. 7.5). What if you can't answer *any* of the questions in a unique way that you will recall when the site challenges you? What if you don't recall your childhood favorite fruit or vegetable,

FIGURE 7.4

Instructions tell users to create an easy-to-remember PIN, but the restrictions make that impossible.

FIGURE 7.5

United.com's registration (2019) burdens long-term memory: a user may not have a unique, memorable answer for any security question.

your favorite subject in school, the color of your childhood home, or the answers to the other questions?

But that isn't where the memory burden ends. Some questions could have several possible answers. Many people had several favorite subjects, childhood homes, or cold-weather activities. To register, they must choose a question and then *remember* which answer they gave to United.com. How? Probably by writing it down somewhere. Then, when United.com asks them the security question, they have to *remember* where they put the answer. Why burden people's memory, when it would be easy to let users make up a security question for which they can easily recall the one possible answer?

FIGURE 7.6

NetworkSolutions.com lets users create a security question if none on the menu work for them.

Such unreasonable demands on people's long-term memory counteract the security and productivity that computer-based applications supposedly provide (Schrage, 2005), as users:

- place sticky notes on or near computers or "hide" them in desk drawers;
- contact customer support to recover passwords they cannot recall;
- use passwords that are easy for others to guess;
- set up systems with no login requirements at all or with one shared login and password.

The registration form at NetworkSolutions.com represents a small step toward more useable security. Like United.com, it offers a choice of security questions, but it also allows users to create their own security question—one for which they can more easily remember the answer (see Fig. 7.6).

Another implication of long-term memory characteristics for interactive systems is that learning and long-term retention are enhanced by user-interface consistency.

The more consistent the operation of different functions, or the more consistent the actions on different types of objects, the less users have to learn. User interfaces that have many exceptions and little consistency from one function or object to another require users to store in their long-term memory many features about each function or object and its correct usage context. The need to encode more features makes such user interfaces harder to learn. It also makes it more likely that a user's memory will drop essential features during storage or retrieval, increasing the chances that the user will fail to remember, misremember, or make other memory errors.

Even though some have criticized the concept of consistency as ill defined and easy to apply badly (Grudin, 1989), the fact is that consistency in a user interface

greatly reduces the burden on users' long-term memory. Mark Twain once wrote, "If you tell the truth, you never have to remember anything." One could also say, "If everything worked the same way, you would not have to remember much." We will return to the issue of user-interface consistency in Chapter 11.

IMPORTANT TAKEAWAYS

- Human attention and memory have strengths and weaknesses. By understanding those strengths and weaknesses, we can design interactive systems to support and augment attention and memory rather than burdening them.
- For the purpose of user-interface design, the most important type of short-term memory is what psychologists call "working memory." It is not a separate memory store but rather everything we are *paying attention* to at a given time. The capacity of working memory—the number of concepts or ideas that we can attend to simultaneously—is five plus or minus one. Think of it as a small set of searchlights illuminating—a few at a time—ideas and concepts in a vast warehouse representing long-term memory.
- Working memory has severely limited capacity, so our attention is highly focused and selective. We miss most of what occurs around us because of this.
- Long-term memories record our experiences and thoughts. Thoughts and experiences trigger large patterns of neurons throughout the brain based on features. A memory is recorded as a collection of neurons that tend to fire in concert when stimulated by perceptions or thoughts. Psychologists divide long-term memory into semantic, episodic, and procedural.
- Unlike computer memory, reading information from human long-term memory can change it. It is also influenced by emotions. These factors make it error prone; it can easily lose or add details.
- VUIs spread information out over time, so they place a higher load on short-term memory than graphical user interfaces do.
- A user interface has modes if a control or user-action has different effects in different contexts. User interfaces that have modes burden working memory by requiring users to mentally keep track of what mode the system is in.
- Implications of memory and attention limitations for user-interface design:
  - Keep voice output brief so as not to overburden users' working memory. Augmenting voice output with screen output or indicator lights can reduce demands on working memory.
  - Avoid moded user-interface designs where possible.

- Minimize the number of calls to action on a page or screen. Avoid competing calls to action.
- Don't expect people to remember arbitrary facts or procedures over long periods. Provide memory aids and tools.
- Design user interfaces to attract users' attention where needed.

CHAPTER

Limits on Attention Shape Our Thought and Action

8

When people interact purposefully with the world around them, including computer systems, some aspects of their behavior follow predictable patterns, some of which result from the limited capacity of attention and short-term memory. When interactive systems are designed to recognize and support those patterns, they fit better with the way people operate. Some user-interface design rules, then, are based directly on the patterns and thus indirectly on the limits of short-term memory and attention. This chapter describes seven important patterns.

WE FOCUS ON OUR GOALS AND PAY LITTLE ATTENTION TO OUR TOOLS

As Chapter 7 explained, our attention has very limited capacity. When people are doing a task—trying to accomplish a goal—most of their attention is focused on the goals and data related to that task. Normally, people devote very little attention to the tools they are using to perform a task, whether they are using computer applications, online services, or interactive appliances. Instead, people think about their tools only superficially, and then only when necessary.

We are, of course, *capable* of attending to our tools. However, attention (i.e., short-term memory) is limited in capacity. When people refocus their attention on their tools, it is pulled away from the details of the task. This shift increases the chances of users losing track of what they were doing or exactly where they were in doing it.

For example, if your lawn mower stops running while you are mowing your lawn, you will immediately stop and focus on the mower. Restarting the mower becomes your primary task, with the mower itself as the focus. You pay little attention to any tools you use to restart the mower, just as you paid little attention to the mower when your primary focus was the lawn. After you restart the mower and resume mowing

Designing with the Mind in Mind. https://doi.org/10.1016/B978-0-12-818202-4.00008-8
Copyright © 2021 Elsevier Inc. All rights reserved.

the lawn, you probably won't remember where you were in mowing the lawn except that the lawn itself shows you.

Other tasks—for example, reading a document, measuring a table, counting goldfish in a fish tank—might not provide such a clear reminder of your interrupted task and position in it. You might have to start over from the beginning. You might even forget what you were doing altogether and go off to do something else.

That is why most software design guidelines state that software applications and most websites should not call attention to themselves; they should fade into the background and allow users to focus on their own goals. That design guideline is even the title of a popular Web design book: *Don't Make Me Think* (Krug, 2014). That is, if your software or website makes me think about *it* rather than what I am trying to do, you've lost me.

WE NOTICE THINGS MORE WHEN THEY ARE RELATED TO OUR GOALS

Chapter 1 described how our immediate goals filter and bias our perception. Chapter 7 discussed the connections between attention and memory. This chapter provides examples showing that perceptual filtering and biasing, attention, and memory are all closely related.

Our environment is full of perceptual details and events. Obviously, we cannot notice and keep track of everything that happens around us. Nonetheless, it is surprising to most people how little we notice of what goes on around us. Because of our extremely limited short-term memory and attention, we don't waste those resources. When an event happens, the few details about it that we notice and remember later are usually those that were important for our goals at the time of the event. This is demonstrated by two related psychological phenomena: *inattentional blindness* and *change blindness*.

Inattentional blindness

When our mind is intensely occupied with a task, goal, or emotion, we sometimes fail to notice objects and events in our environment that we otherwise would have noticed and remembered. This phenomenon has been heavily studied by psychologists and labeled *inattentional blindness* (Simons and Chabris, 1999; Simons, 2007).

A clear demonstration of inattentional blindness is an experiment in which human subjects watched a video of two basketball teams passing a ball from one player to another. Subjects were told to count the number of times the white-suited team passed the ball. While they watched the video and counted the ball passes, a person in a gorilla suit sauntered onto the basketball court, thumped his chest, and then walked out of view (see Fig. 8.1). Afterward, subjects were asked what they remembered from the video. Surprisingly, about half of them did not notice the gorilla. Their attention was fully occupied with the task (Simons and Chabris, 1999).[1]

[1] For more information and videos, search the Web or YouTube for "inattentional blindness."

FIGURE 8.1

Scene from video used in the "invisible gorilla" study. For more information about the study and to see the video, go to theinvisiblegorilla.com. (Figure provided by Daniel Simons.)

Change blindness

Another way researchers have shown that our goals strongly focus our attention and memory is by showing people a picture, then showing them an altered version of the same picture and asking them how the two pictures differ. What is surprising is that the second picture can differ from the first in fairly clear ways without people noticing. The reason is that people's attentions were captured by features in the image other than those that changed, and human attention can only keep track of a few features at a time. This is a type of inattentional blindness, but it is so common that it has its own name: *change blindness* (Angier, 2008).

Look at Fig. 8.2, then look at Fig. 8.3. Flip back and forth a few times. What is different about the two pictures?

Even if you noticed that the tree branch at the top of Fig. 8.2 is missing in Fig. 8.3, you may have missed the fact that the person on the far left in Fig. 8.2 is missing in Fig. 8.3. If so, it was probably because your attention was captured by the globe, water, clouds, building, etc.

To explore further, researchers gave people questions to answer about the first picture, which affected their goals in looking at it and therefore what features they paid attention to. The result: people tend to miss changes in features other than those their goals made them pay attention to. So for example, if I showed you Fig. 8.2 and asked you to notice any people in it, then showed you Fig. 8.3 and asked what was different, you would certainly notice the absence of the man.

A striking example of change blindness comes from experiments in which researchers holding city maps posed as lost tourists and asked local passersby for directions.

FIGURE 8.2

Look at this image, then at Fig. 8.3. What is different? (Credit: Wikipedia article on "Change Blindness.")

FIGURE 8.3

Look at this image, then at Fig. 8.2. What is different? (Credit: Wikipedia article on "Change Blindness.")

When the local person focused on the tourist's map to find the best route, two workmen—actually, more researchers—walked between the tourist and the local person carrying a large door, and a new researcher-tourist replaced the first one. After the door passed, over half the people continued helping the "tourist" without noticing any change, even when the two "tourists" differed in hair color or whether they had

FIGURE 8.4

Users of RoadScholar.org may not notice that when they change the date of a trip (bottom center), the price changes (middle right).

a beard (Simons and Levin, 1998).[2] Some people even failed to notice changes in *gender*. The researchers' explanation: people focus attention on the tourist only enough to decide if he or she is a threat or worth helping, mentally record only that the person is a tourist needing help, and then focus on the map and the task of giving directions.

When people interact with software, electronic appliances, or online services, it is not uncommon for them to fail to notice important changes in what is displayed. For example, in a study of older adults using travel websites, some participants exhibited change blindness; when they changed trip options (e.g., departure date or city), they did not notice that the price changed (Finn and Johnson, 2013) (see Fig. 8.4).

The user-interface design guideline that follows from such findings is to make important changes highly salient—so obvious that they are hard to miss—and take steps to draw users' attention to the change. For example, a way to draw users' attention to a new error message is to vibrate it briefly when it first appears (see Chapter 6) or highlight it briefly before it reverts to a "normal" appearance.

What happens in our brains

Using functional magnetic resonance imagery and electrical encephalography, researchers have studied the effect of attention on how our brains respond to objects displayed on a computer screen.

When people passively watch a computer display with objects appearing, moving around, and disappearing, the visual cortex of their brains registers a certain activity level. When people are told to look for (i.e., pay attention to) certain objects, the activity level of their visual cortex increases significantly. When they are told to ignore certain objects, the neural activity level in their visual cortex actually drops when those objects appear. Later, their memory for which objects they saw and didn't see correlates with the degree of attention they paid to them and the level of brain activity (Gazzaley, 2009).

WE USE EXTERNAL MEMORY AIDS TO KEEP TRACK OF WHAT WE ARE DOING

Because our short-term memory and attention are so limited, we learn not to rely on them. Instead, we mark up our environment to show us where we are in a task. Examples include:

- ***Counting objects***. If possible, we move already counted objects into a different pile to indicate which objects have already been counted. If we cannot move an object, we point to the last object counted. To keep track of the number we are on, we count on our fingers, draw marks, or write numbers.

[2] For demonstrations of change blindness, search YouTube for those words, or for "door study" or "person swap."

FIGURE 8.5

The macOS software update shows which updates are done (green check) versus which are in progress (rotating circle).

- ***Reading books***. When we stop reading, we insert bookmarks to show what page we were on.
- ***Arithmetic***. We learn methods of doing arithmetic on paper, or we use a calculator.
- ***Checklists***. We use checklists to aid both our long- and short-term memory. In critical or rarely performed tasks, checklists help us remember everything that needs to be done. In that way, they augment our faulty long-term memory. While doing the task, we check off items as we complete them. That is a short-term memory aid. A checklist that we cannot mark up is hard to use, so we copy it and mark the copy.
- ***Editing documents***. People often keep to-be-edited documents, documents that are currently being edited, and already edited documents in separate folders.

One implication of this pattern is that interactive systems should indicate what users have done versus what they have not yet done. Most email and texting apps do this by marking already-read versus unread messages, most websites do it by marking visited versus unvisited links, and many applications do it by marking completed steps of a multipart task (see Fig. 8.5).

A second design implication is that interactive systems should allow users to mark or move objects to indicate which ones they have worked on versus which ones they have not. MacOS lets users assign colors to files. Like moving files between folders, this technique can be used to keep track of where one is in a task (see Fig. 8.6).

FIGURE 8.6

The macOS lets users assign colors to files or folders; users can use the colors to track their work.

Used by permission, OK/Cancel.com.

WE FOLLOW INFORMATION "SCENT" TOWARD OUR GOAL

Focusing our attention on the things relevant to our goals makes us interpret what we see on a display or hear in a telephone menu in a very *literal* way. People don't think deeply about instructions, command names, option labels, icons, navigation bar items, or any other aspect of the user interface of computer-based tools. If the goal in their head is to make a flight reservation, their attention will be attracted by words like "buy," "flight," "ticket," or "reservation" or symbols depicting money, tickets, or flights. Other items a designer or marketer might hope will attract customers, such as "bargain hotels," will not attract the attention of people who are trying to book a flight, although they might be noticed by people who are seeking bargains.

For each task below, what on the screen would attract your attention?

- Pay a bill
- Transfer money to your savings account
- Pay your dentist by funds transfer
- Change your PIN
- Open a new account
- Purchase travelers' cheques

FIGURE 8.7

ATM screen—our attention is drawn toward items that match our task and goal.

This tendency of people to notice only things on a computer screen that match their goal, and the literal thinking that people exhibit when performing a task on a computer, has been called "following the *scent* of information toward the goal" (Chi et al., 2001; Nielsen, 2003). Consider the ATM display shown in Fig. 8.7. What is the first thing on the screen that gets your attention when you are trying to do each of the tasks listed?

You probably noticed that some of the tasks direct your attention initially to the wrong option. Is "Pay your dentist by funds transfer" under "Payment" or "Transfer"? "Open a new account" probably sent your eyes briefly to "Open-End Fund" even though it is actually under "Other Service." Did the task "Purchase travelers' cheques" make you glance at "Request Cheque Book" because of the word they share?

The goal-seeking strategy of following the information scent, observed across a wide variety of situations and systems, suggests that interactive systems should be designed so that the scent is strong and really leads users to their goals. To do that, designers need to understand likely user goals at each decision point in a task and ensure that each choice point in the software provides options for every important user goal and clearly indicates which option leads to which goal. Analyzing software in this way is useful, and is called a *cognitive walkthrough* (Wharton et al., 1994).

For example, imagine that you want to cancel a reservation you made or payment you scheduled. You tell the system to cancel it, and a confirmation dialogue box appears asking if you really want to do that. How should the options be labeled? Given that people interpret words literally in following the information scent toward their goal, the standard confirmation button labels "OK" (for yes) and "Cancel" (for no) would give a misleading scent. If we compare a cancellation confirmation dialogue box from Marriott.com with one from Bank of New Zealand, we see that Marriott.com's labeling provides a clear scent, while Bank of New Zealand's button-labels would mislead users into clicking on the wrong buttons (see Fig. 8.8).

FIGURE 8.8

The button labels on Marriott's reservation cancellation confirmation dialogue (A) provide a clear information "scent" leading users toward their goal, while Bank of New Zealand's wire-transfer confirmation dialogue (B), in which "OK" cancels the transaction and "Cancel" cancels the cancellation, does not.

WE PREFER FAMILIAR PATHS

People know that their attention is limited, and they act accordingly. While pursuing a goal, they take *familiar* paths whenever possible rather than exploring *new* ones, especially when working under deadlines. As explained more fully in Chapter 10, exploring new paths is problem-solving, which severely taxes our attention and short-term memory. In contrast, taking familiar, well-learned routes can be done fairly automatically and does not consume much attention and short-term memory.

Years ago, in a usability test session, a test participant in the middle of a task said to me:

> **I'm in a hurry, so I'll do it the long way.**

He knew there probably was a more efficient way to do what he was doing, but he also knew that learning the shorter way would require time and thought that he was unwilling to spend.

Once we learn one way to perform a certain task using a software application, we may continue to do it that way and *never* discover a more efficient way. Even if we discover or are told that there is a "better" way, we may stick with the old way because it is familiar, comfortable, and most important, requires little thought. Avoiding thought when using computers is important. People are willing to type *more* to think *less*.

Why is that? Are we mentally lazy? Usually, yes. Conscious thought is slow, strains working memory, and consumes much energy. We essentially run on batteries—the food we eat—so energy conservation is an important feature of our makeup. Operating via automatic processes is fast, doesn't strain working memory, and conserves energy. So the brain tries to run on automatic as much as possible (Kahneman, 2011; Eagleman, 2012, 2015).

The human preference for familiar, mindless paths and "no-brainer" decisions has several design implications for interactive systems:

- ***Sometimes mindlessness trumps keystrokes***. With software intended for casual or infrequent use, such as bank ATMs or household accounting applications, allowing users to become productive quickly and reducing their need to problem-solve while working is more important than saving keystrokes. Such software simply isn't used enough for the number of keystrokes per task to matter much. On the other hand, in software used all day in intensive work environments by highly trained users, such as airline telephone reservations operators, unnecessary keystrokes required to perform a task are very costly.
- ***Guide users to the best paths***. From its first screen or home page, software should show users the way to their goals. This is basically the guideline that software should provide a clear information scent.
- ***Help experienced users speed up***. Make it easy for users to switch to faster paths after they have gained experience. The slower paths for newcomers should show users faster paths if there are any. This is why most applications show keyboard accelerators for frequently used functions in their menu-bar menus.

OUR THOUGHT CYCLE: GOAL, EXECUTE, EVALUATE

Over many decades, researchers studying human behavior have found a cyclical pattern that seems to hold across a wide variety of activities. Cognitive scientist Don Norman (1988) named the pattern the "human action cycle":

- Form a ***goal*** (e.g., open a bank account, make dinner, or revise a document).
- Choose and ***execute*** actions to try to make progress toward the goal.
  - Translate goal into an unordered collection of tasks (e.g., make main course, make salad, make dessert).

  - Order the tasks into an action sequence (i.e., a plan) that moves toward the goal.
  - Execute the action sequence.
- ***Evaluate*** whether the actions worked—that is, whether the goal has been reached or is nearer than before.
  - Check the results of executing the action sequence.
  - Compare the results with the expected results.
  - Determine whether at the goal or closer to the goal than before.
- ***Repeat*** until the goal is reached (or appears unreachable).

People cycle through this pattern constantly (Card et al., 1983; Norman, 1988). In fact, we run through it at many different levels simultaneously. For example, we might be trying to insert a picture into a document, which is part of the higher-level task of writing a term paper, which is part of the higher-level task of passing a history course, which is part of the higher-level task of completing college, which is part of the higher-level goal of getting a good job, in order to achieve our top-level goal of having a comfortable life.

As an example, let's run through the cycle for a typical computer task—buying an airline ticket online. The person first forms the primary goal of the task and then begins to break that down into actions that appear to lead toward the goal. Promising actions are selected for execution, executed, and then evaluated to determine whether they have moved the person closer to the goal.

- ***Goal***: Buy airline ticket to Berlin using your favorite travel website.
- ***Step 1***: Go to travel website. You are still far from the goal.
- ***Step 2***: Search for suitable flights. This is a very normal, predictable step at travel websites.
- ***Step 3***: Look at search results. Choose a flight from those listed. If no flights on the results list are suitable, return to Step 2 with new search criteria. You are not at the goal yet, but you feel confident of getting there.
- ***Step 4***: Go to checkout. Now you are getting so close to your goal that you can almost smell it.
- ***Step 5***: Confirm flight details. Check it—all correct? If no, back up; otherwise, proceed. Almost done.
- ***Step 6***: Purchase ticket with credit card. Check credit card information. Everything look okay?
- ***Step 7***: Save or print e-ticket. Goal achieved. Add departure time and date to your calendar.

To keep the airline ticket example brief, we didn't get into the details of each step. If we had, we would have seen substeps that followed the same *goal–execute–evaluate* cycle.

Let's try another example, this time examining the details of some of the higher-level steps. This time the task is sending flowers to a friend. If we simply look at the top level, we see the task like this:

Send flowers to friend.

If we want to examine the goal–execute–evaluate cycle for this task, we must break it down a bit. We must ask, *How* do we send flowers to a friend? To do that, we break the top-level task down into subtasks:

Send flowers to friend.
 Find flower delivery website.
 Order flowers to be delivered to friend.

For many purposes, the two steps we have identified are enough detail. After we execute each step, we evaluate whether we are closer to our goal. But *how* is each step executed? To see that, we have to treat each major step as a subgoal and break it down into substeps:

Send flowers to friend.
 Find flower delivery website.
  Open Web browser.
  Go to Web search page (e.g., Google).
  Type "flower delivery" into search-text box.
  Scan the first page of search results.
  Visit some of the listed links.
  Choose a flower delivery service.
 Order flowers to be delivered to friend.
  Review service's flower selection.
  Choose flowers.
  Specify delivery address and date.
  Pay for flowers and delivery.

After each substep is executed, we evaluate to see if it is getting us closer to the subgoal of which it is part. If we want to examine how a substep is executed and evaluated, we have to treat it as a sub-subgoal and break it into its component steps:

Send flowers to friend.
 Find flower delivery website.
  Open Web browser.
   • Click browser icon on taskbar, startup menu, or desktop.

Go to Web search page (e.g., Google).
- If search site isn't browser's starting page, choose it from favorites list.
- If search site is not on favorites list, type its address (e.g., "Google.com') into browser's address box.

Type "flower delivery" into search-text box.
- Set text-insertion point in search-text box.
- Type the text.
- Correct typo: "floowers" to "flowers."

Visit some of the resulting links.
- Move screen pointer to link.
- Click on link.
- Look at resulting Web page.

Choose a flower delivery service.
- Enter chosen service's URL into browser.

...

You get the idea. We could keep expanding down to the level of individual keystrokes and pointer movements, but we rarely need that level of detail to be able to understand the task well enough to design software to fit its steps and the goal–execute–evaluate cycle applied to each step.

How can software support users in carrying out the goal–execute–evaluate cycle? Any of these ways:

- ***Goal***. Provide clear paths—including initial steps—for the user goals that the software is intended to support.
- ***Execute***. Software concepts (objects and actions) should be based on task rather than implementation (see Chapter 11). Don't force users to figure out how the software's objects and actions map to those of the task. Provide a clear information scent at choice points to guide users to their goals. Don't make them choose actions that seem to take them away from their goal to achieve it.
- ***Evaluate***. Provide feedback and status information to show users their progress toward the goal. Allow users to back out of tasks that didn't take them toward their goal.

Fig. 8.9A shows a progress indicator that provides clear feedback about the user's progress through a series of steps. The indicator in Fig. 8.9B shows even more clearly where the user is in the sequence. Either design provides input for the Evaluate part of the action cycle. By the way, does Fig. 8.9A seem familiar? If so, it is because you saw it in Chapter 4 (see Fig. 4.17), and your brain recognized it.

FIGURE 8.9

Indicators showing progress through a series of steps. (A) Good; (B) best.

FIGURE 8.10

Progress indicators on mobile websites: (A) doodle.com; (B) kp.org.

FIGURE 8.11

Password strength indicators show users how close they are to the goal.

Real examples of progress indicators can be seen on the websites of the Doodle scheduling service (doodle.com) and Kaiser Permanente (kp.org). Doodle's poll-creation function simply shows users what step number they are on out of the total number of steps (see Fig. 8.10A). That is better than no indication, but it is better to show the steps visually, as KP.org's appointment function does (see Fig. 8.10B).

Even when progress toward a goal is not measured by its location in a predefined series of steps, designers can provide feedback to help users see how close they are to the goal. Password strength indicators are a good example (see Fig. 8.11).

AFTER WE ACHIEVE A TASK'S PRIMARY GOAL, WE OFTEN FORGET CLEANUP STEPS

The goal–execute–evaluate cycle interacts strongly with short-term memory. This interaction makes perfect sense: short-term memory is really just the focus of attention at any given moment. Part of the focus of attention is our current goal. The rest of our attentional resources are directed toward obtaining the information needed to achieve our current goal. The focus shifts as tasks are executed and the current goal shifts from high-level goals to lower-level ones before returning to the next high-level goal.

Attention is a very scarce resource. Our brain does not waste it by keeping it focused on anything that is no longer important. Therefore, when we complete a task, the attentional resources focused on that task's main goal are freed to be refocused on other things that are now more important. The impression we get is that once we achieve a goal, everything related to it often immediately "falls out" of our short-term memory—that is, we forget about it.

One result is that people often forget loose ends of tasks. For example, people often forget to do these things:

- Turn car headlights off after arrival.
- Remove the last pages of documents from copiers and scanners.
- Turn stove burners and ovens off after use.
- Add closing parentheses and quotation marks after typing text passages.
- Turn off turn signals after completing turns.
- Take books they were reading on a flight with them when they exit the plane.
- Log out of public computers when finished using them.
- Set devices and software back to normal mode after putting them into a special mode.

These end-of-task short-term memory lapses are completely predictable and avoidable. When they happen to us, we call ourselves "absent-minded," but they are the result of how the brain works (or doesn't) combined with a lack of support from our devices.

To avoid such lapses, which are called "closure slips" (see Chapter 15), interactive systems can and should be designed to remind people that loose-end steps remain. In some cases, it may even be possible for the system to complete the task itself. For example:

- Cars already turn off turn signals after a turn.
- Cars should (and now do) turn off headlights automatically when the car is no longer in use or at least remind drivers that the lights are still on.
- Copiers and scanners should automatically eject all documents when tasks are finished or at least signal that a page has been left behind.

- Stoves should signal when a burner is left on with no pot present for longer than some suitable interval, and ovens should do likewise when left on with nothing in them.
- Computers should issue warnings if users try to power them down or put them to sleep before the computer has finished a background task (e.g., saving files or sending a document to a printer).
- Special software modes should revert to "normal" automatically, either by timing out—as some appliances do—or through the use of spring-loaded mode controls that must be physically held while in the nonnormal state and that revert to normal when released (Johnson, 1990).

Software designers should consider whether the tasks supported by a system they are designing have cleanup steps that users are likely to forget, and if so, they should design the system either to help users remember or eliminate the need for users to remember. See Chapter 15 for more about designing to decrease the likelihood of closure slips and other user errors.

IMPORTANT TAKEAWAYS

People do the following when using digital tools and services:

- Focus on our goals and pay little attention to our tools due to the limited capacity of working memory and attention. Design to let users do that. Design so your software fades into the background.
- Notice things more when they are related to our goals. We often miss events, changes, and objects that are right in front of us if they aren't relevant to our goals.
- Use external memory aids to keep track of what we are doing. Design software to show users what they have already done. For example, mark visited links, messages read, steps completed.
- Follow information "scent" toward our goal. When designing a digital product or service, analyze users' likely goals, and for each user goal, make sure every page or screen includes clues indicating which action will lead toward that goal.
- Prefer familiar paths. Users tend to stick to what they know because they usually prefer to run on autopilot. Design so users can, after some experience, fall into automatic habits.
- Achieve goals via a constant thought cycle: goal, execute, evaluate. Design to support this cycle.
- Often forget cleanup steps. User interfaces should remind users of anything left undone after the goal is achieved.

CHAPTER

Recognition is Easy; Recall is Hard

9

Chapter 7 described the strengths and limitations of long-term memory and their implications for the design of interactive systems. This chapter extends that discussion by describing important differences between two functions of long-term memory: recognition and recall.

RECOGNITION IS EASY

The human brain was "designed," through millions of years of natural selection and evolution, to recognize things quickly. By contrast, recalling memories—that is, retrieving them without perceptual support—must not have been as crucial for survival, because our brains are much worse at that.

Remember how our long-term memory works (see Chapter 7): Perceptions enter through our sensory systems, and their signals, when they reach the brain, cause complex patterns of neural activity. The neural pattern resulting from a perception is determined not only by the features of the perception but also by the context in which it occurs. Similar perceptions in similar contexts cause similar patterns of neural activity. Repeated activation of a particular neural pattern makes that pattern easier to reactivate in the future. Over time, connections between neural patterns develop in such a way that activating one pattern activates the other. Roughly speaking, each pattern of neural activity constitutes a different memory.

Patterns of neural activity, which is what memories are, can be activated in two different ways:

1. By more perceptions coming in from the senses
2. By other brain activity

If a perception comes in that is similar to an earlier one and the context is close enough, it easily stimulates a similar pattern of neural activity, resulting in a sense of

Designing with the Mind in Mind. https://doi.org/10.1016/B978-0-12-818202-4.00009-X
Copyright © 2021 Elsevier Inc. All rights reserved.

recognition. Recognition is essentially perception and long-term memory working in concert.

As a result, we assess situations very quickly. Our distant ancestors on the East African savannah had only a second or two to decide whether an animal emerging from the tall grasses was something they would regard as food or something that would regard *them* as food (see Fig. 9.1). Their survival depended on it.

Similarly, people recognize human faces very quickly—usually in a fraction of a second (see Fig. 9.2). Until recently, the workings of this process were a mystery. However,

FIGURE 9.1

Early humans had to recognize quickly whether animals they saw were prey or predators.

FIGURE 9.2

How long did it take you to recognize these faces?[1]

[1] U.S. Presidents Barack Obama and Bill Clinton.

that was when scientists assumed recognition was a process in which perceived faces were stored in separate short-term memory and compared with those in long-term memory. Because of the speed with which the brain recognizes faces, cognitive scientists assumed that the brain must search many parts of long-term memory simultaneously via what computer scientists call *parallel processing*. However, even a massively parallel search process could not account for the astounding rapidity of facial recognition.

Nowadays, perception and long-term memory are considered closely linked, which demystifies the speed of facial recognition somewhat. A perceived face stimulates activity in millions of neurons in distinct patterns. Individual neurons and groups of neurons that make up the pattern respond to specific features of the face and the context in which the face is perceived. Different faces stimulate different patterns of neural response. If a face was perceived previously, its corresponding neural pattern will already have been activated. The same face perceived again reactivates the same pattern of neural activity, only more easily than before. That *is* the recognition. There is no need to search long-term memory—the new perception reactivates the same pattern of neural activity, more or less, as the previous one. Reactivation of a pattern *is* the reactivation of the corresponding long-term memory.

In computer jargon, we could say that the information in human long-term memory is *addressed by its content*, but the word "addressed" wrongly suggests that each memory is located at a specific spot in the brain. In fact, each memory corresponds to a pattern of neural activity extending over a wide area of the brain.

That explains why, when presented with faces we have not seen before and asked if they are familiar, we don't spend a long time searching through our memories to try to see if that face is stored in there somewhere (see Fig. 9.3). There is no search.

FIGURE 9.3

How long did it take you to realize that you do not recognize this face?[2]

[2]Average male face (FaceResearch.org).

FIGURE 9.4

We can recognize complex patterns quickly.

A new face stimulates a pattern of neural activity that has not been activated before, so no sense of recognition results. Of course, a new face may be so similar to a face we have seen that it triggers a *misrecognition*, or it may be just similar enough that the neural pattern it activates triggers a familiar pattern, causing a feeling that the new face reminds us of someone we know.

An interesting aside is that face recognition is a special type of recognition. It has its own dedicated mechanisms in our brains, hardwired in by evolution; we do not have to learn to recognize human faces (Eagleman, 2012, 2015).

Similar mechanisms make our visual system fast at recognizing complex patterns, although unlike face recognition, they develop largely through experience rather than being wired in from birth. Anyone with at least a high school education quickly and easily recognizes a map of Europe and a chessboard (see Fig. 9.4). Chess masters who have studied chess history may even recognize the chess position as Kasparov versus Karpov, 1986.

RECALL IS HARD

In contrast, *recall* is long-term memory reactivating old neural patterns without immediate similar perceptual input. That is much harder than reactivating a neural pattern with the same or similar perceptions. People *can* recall memories, so it obviously *is* possible for activity in other neural patterns or input from other areas of the brain to reactivate a pattern of neural activity corresponding to a memory. However, the coordination and timing required to recall a memory increase the likelihood that the wrong pattern or only a subset of the right pattern will be activated, resulting in a failure to recall.

Whatever the evolutionary reasons, our brain did not evolve to recall facts. Many schoolchildren dislike history class because it demands that they remember facts, such as the year the English Magna Carta was signed, the capital city of Argentina, and the names of all 50 US states. Their dislike is not surprising; the human brain is not well suited for that sort of task.

Because people are bad at recall, they develop methods and technologies to help them remember facts and procedures (see Chapter 7). Orators in ancient Greece used the *method of loci* to memorize the main points of long speeches. They imagined a large building or plaza and mentally placed their talking points in spots around it. When presenting the speech, they mentally "walked" through the site, picking up their talking points as they passed.

Today we rely more on external recall aids than on internal methods. Modern-day speakers remember their talking points by writing them down on paper or displaying them in overhead slides or presentation software. Businesses keep track of how much money they have, owe, or are owed by keeping account books. To remember contact information of friends and relatives, we use address books. To remember appointments, birthdays, anniversaries, and other events, we use calendars and alarm clocks. Electronic calendars are best for remembering appointments, because they actively remind us; we don't have to remember to look at them.

RECOGNITION VERSUS RECALL: IMPLICATIONS FOR USER-INTERFACE DESIGN

The relative ease with which we recognize things rather than recall them is the basis of the graphical user interface (GUI) (Johnson et al., 1989). The GUI is based on two well-known user-interface design rules:

- ***See and choose—or hear and choose—is easier than recall and type***. Show or tell users their options and let them choose rather than force them to recall the options and tell the computer what they want. This rule is the reason GUIs have almost replaced command-line user interfaces (CLIs) in personal computers (see Fig. 9.5), and CLIs are absent in tablets and smartphones. "Recognition rather than recall" is one of Nielsen and Molich's (1990) widely used heuristics for evaluating user interfaces. By contrast, using language to control a software application sometimes allows more expressiveness and efficiency than a GUI would. Thus, *recall and type* remains a useful approach, especially in cases where users can easily recall what to type, such as when entering target keywords into a search box.

- ***Use pictures where possible to convey function***. People recognize pictures very quickly, which also stimulates the recall of associated information. For this reason, today's user interfaces often convey function through pictures (see Figs. 9.6 and 9.7), such as desktop or toolbar icons, error symbols, and graphically

FIGURE 9.5

The main design rule behind GUIs: "See and choose is easier than remember and type."

FIGURE 9.6

Wordpress.com uses symbols plus text to label functional pages on the dashboard.

depicted choices. Pictures that people recognize from the physical world are useful because they can be recognized without needing to be taught. This recognition is good as long as the familiar meaning matches the intended meaning in the computer system (Johnson, 1987). However, using familiar pictures from the physical world is not absolutely crucial. Computer users can learn to associate new icons and symbols with their intended meaning if these graphics are well designed. Memorable icons and symbols hint at their meaning, are distinguishable from others, and are used consistently to mean the same thing even across applications.

The GUI originated in the mid-1970s and became widespread in the 1980s and 1990s. Since then, additional design rules have arisen based on human perception in general and on recognition and recall in particular. The following sections outline a few of these newer rules.

FIGURE 9.7

Desktop icons convey function via recognition—by analogy with physical objects or by experience.

Use thumbnail images to depict full-sized images compactly

Recognition is fairly insensitive to the size at which objects and events are displayed. After all, we are able to recognize objects regardless of their distance from us. What is important are features: as long as most of the same features are present in the new picture that were in the original one, the new perception stimulates the same neural pattern, resulting in recognition.

Therefore, a great way to display pictures people have already seen is to present them as thumbnail images. The more familiar a picture, the smaller the thumbnails of it can be and still be recognizable. Displaying thumbnails instead of full-sized images allows people to see more of their options, data, history, etc. at once.

FIGURE 9.8

Microsoft PowerPoint can show slides as thumbnails, providing an overview based on recognition.

FIGURE 9.9

The Mozilla Firefox Web browser lists logos of frequently visited sites for quick recognition and choice.

Photo management and presentation applications use thumbnail images to give users an overview of their images or slides (see Fig. 9.8). Web browsers use website logos to show sites a user has frequently visited (see Fig. 9.9).

The larger the number of people who will use a function, the more visible the function should be

For the reasons described before, recall often fails. If a software application hides its functionality and requires its users to recall what to do, some percentage of users will fail at doing so. If the software has many users, that percentage who fail to recall—even

if it is small—adds up to a significant number. Software designers obviously don't want a significant number of users to fail in using their product.

The solution is to make the functions that many people need highly visible so users see and *recognize* their options rather than having to *recall* them. By contrast, functionality that few people will use—especially when those few people are highly trained—can be hidden, for example, behind "Details" panels, in right-click menus, or via special key combinations.

Provide cues to let users recognize where they are

Visual recognition is fast and reliable, so designers can use visual cues to show users instantly where they are. For example, it is a well-known Web design rule that all pages in a website should have a common distinctive visual style so people can easily tell whether they are still on the site or have gone to a different one. Slight but systematic variations of a site's visual style can show users which section of the site they are in.

Some desktop operating systems allow users to set up multiple desktops ("rooms" or "workspaces") as locations for different categories of work. Each has its own background graphic to help users recognize where they are.

Avoid changing prominent visual cues on a given page needlessly—it can disorient users

Since visual cues help indicate where users are in an information space, *extraneous* changes of visual cues can mislead users into thinking they have moved to a new screen or page. If a prominent image or color on an app screen or Web page changes without warning, some users, especially older adults, may mistakenly believe they are on a different page (Finn and Johnson, 2013; Johnson and Finn, 2017) (see Fig. 9.10).

FIGURE 9.10

GrandCircleTravel.com's homepage has large images that change automatically, causing some visitors—especially older adults—to think they are on a different page.

Make authentication information easy to recall

People know it is hard to recall arbitrary facts, words, and sequences of letters or digits. That is why they often write passwords and challenge-question answers down and keep the information in places that are easy to reach and thus insecure. Or they base passwords on their children's initials, their birthdates, their street address, and other information they know they can recall. Unfortunately, such passwords are often too easy for other people to guess (Schrage, 2005). How can designers help users avoid such unsafe behavior?

We can at least not make it hard for people to recall their login information, like the systems cited in Chapter 7 that impose burdensome password restrictions or offer a limited choice of challenge questions.

Instead, we can give users the freedom to create passwords they can remember and challenge questions for which they can remember the correct response. We can also let users supply password *hints* that the system can present to them, under the assumption that users can devise hints that will serve as a recall probe for them but not identify the password to third parties.

Authentication methods that do not rely on users to recall the authentication data would seem to be a solution. Biometric authentication methods, such as iris scans, digital fingerprint scans, and voice identification, fall into this category. However, many people regard these methods as privacy threats because they require the collection and storage of individuals' biometric data, creating the potential for information leaks and abuse. Therefore, while biometric authentication does not burden a user's memory, it would have to be implemented in a way that meets stringent privacy requirements to be widely accepted.

IMPORTANT TAKEAWAYS

- The human brain is basically a recognition engine. Similar perceptions in similar contexts activate similar patterns of neural activity. That is recognition. It occurs very quickly, without what computer scientists would regard as searching. Recognition—or nonrecognition—of human faces is especially fast.
- Recall requires reactivating a neural pattern when the original stimulus is absent. The stimulus comes from within the brain—from other neural activity. The brain is not as good at recall as at recognition. Therefore, people throughout history have invented methods and tools to help them recall information, such as writing, books, printing, calendars, grocery lists, alarm clocks, computers, smartphones, and slide presentations.

- Recognition versus recall—implications for UI design:
  - See and choose—or hear and choose—is easier than recall and type. Design to let people perceive their options rather than requiring users to recall their options.
  - Use pictures where possible to convey function: desktop icons, toolbar symbols, error symbols, etc.
  - Use thumbnail images to depict full-size images compactly. Once someone has seen a full-sized picture, thumbnail images are usually enough to spark recognition. Using small thumbnails let people see more of their data at once.
  - The larger the number of people who will use a function, the more visible it should be. If many people will use a function, it should be highly visible. If only a few highly trained people will use a function, it can be hidden, requiring extra steps to get to it.
  - Provide cues to let users recognize where they are. Avoid changing prominent visual cues on a page needlessly—it can disorient users.
  - Make authentication information easy to recall. Don't force users to devise passwords they cannot recall. Provide support for managing, recovering, and changing passwords. Transition to biometric authentication, but keep users' data secure.