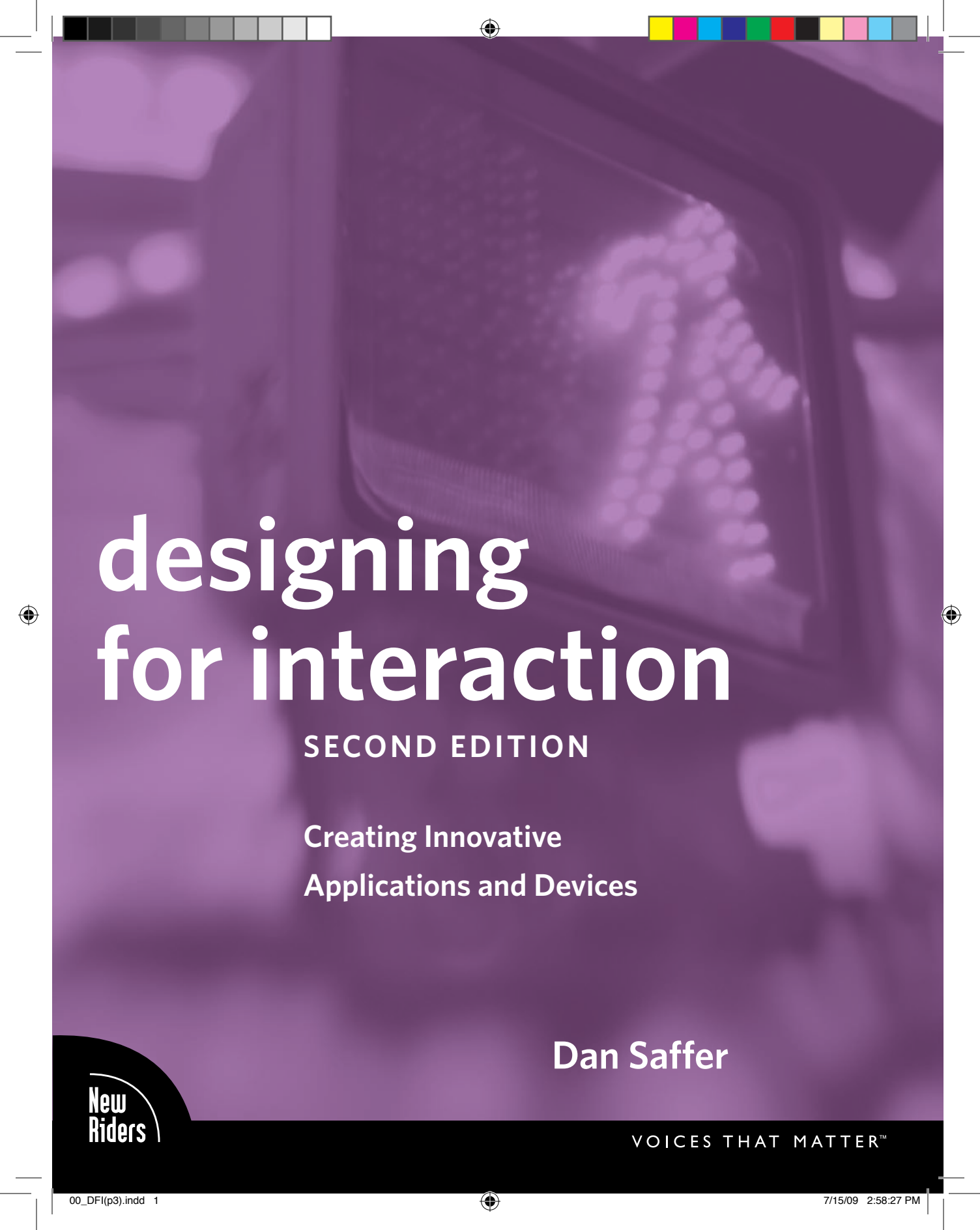


designing for interaction

SECOND EDITION

Creating Innovative Applications and Devices

Dan Saffer

New Riders

VOICES THAT MATTER™



Designing for Interaction, Second Edition:
Creating Innovative Applications and Devices
Dan Saffer

New Riders
1249 Eighth Street
Berkeley, CA 94710
510/524-2178
510/524-2221 (fax)

Find us on the Web at: www.newriders.com
To report errors, please send a note to errata@peachpit.com

New Riders is an imprint of Peachpit, a division of Pearson Education

Project Editor: Michael J. Nolan
Development Editor: Box Twelve Communications, Inc.
Production Editor: Becky Winter
Copyeditor: Rose Weisburd
Proofreader: Darren Meiss
Indexer: James Minkin
Cover designer: Aren Howell
Interior designer: Andrei Pasternak with Maureen Forys

ISBN 13: 978-0-321-64339-1
ISBN 10: 0-321-64339-9
9 8 7 6 5 4 3 2 1
Printed and bound in the United States of America


Acknowledgements

The shadow of the two years I spent steeping in design at Carnegie Mellon University looms large over this book. When I wrote the first edition, I found myself constantly referring to my notes from that time and hearing the echoes of my professors' words, including those of Dan Boyarski, Kristen Hughes, Karen Moyer, John Zimmerman, and Jodi Forlizzi. I want to particularly note the influence of Dick Buchanan, who immeasurably broadened my understanding of this discipline, and my friend and advisor Shelley Evenson, who taught me at least half of what I know about interaction design. Without her knowledge and experience, poorly filtered through me, this book would be shallow indeed.

In the second edition, the influence of my professional colleagues at Adaptive Path and now Kicker Studio can be felt. Particular kudos to Adaptive Pathers Brandon Schauer, Peter Merholz, and especially Henning Fischer, who helped lead me, sometimes kicking and screaming, into the world of design strategy. This book is much improved for its inclusion. My Kicker Studio partners Jennifer Bove and Tom Maiorana have been generous with their editing and design help, not to mention encouragement.

My interviewees were generous with their time and expertise and I'd like to especially thank them. Your presence in my book honors me.

I'm also grateful to companies who lent their case studies and beautiful product images to the book, illustrating my points better than I could have with words alone.

The staff at Peachpit/New Riders has been a tremendous help in making this book what it is, in this edition and the last. My editors Michael Nolan, Becky Winter, and Jeff Riley have polished my rough edges (and there were many) into the fine tome you have in your hands (or on your screen). Another special thanks goes to my friend and technical editor Bill DeRouchey, whose insights burnished this book.

Other friends who have lent their support and help with both this edition and the last: Phi-Hong Ha, Jesse James Garrett, Andrew Crow, Jannine Takahashi-Crow, Kristina Halvorson, Marc Rettig, Adam Greenfield, Ryan Freitas, Rae Brune, Jennifer Fraser, Lane Becker, Brian Oberkirch, Chad Thorton, Rob Adams, Kenneth Berger, Willow Stelzer, Kim Lenox, Todd Wilkens, Uday Gajendar, Chiara Fox, Dave Malouf, Kim Goodwin,

Nancy Broden, Alan Cooper, Dana Smith, Rachel Hinman, Erika Hall, Rachel Glaves, Samantha Soma, Sarah Nelson, Jared Spool, Jody Medich, Mike Scully, Laura Kirkwood-Datta, Liz Danzico, Kevin Daly, Shinohara Toshikazu, Zach Hettinger, my in-laws Mary and Barry King, and my sister, Meagan Duffy.

Thanks to my parents, who bought me my first computer (a Timex Sinclair 1000) and a 300 baud modem and who paid the ensuing long-distance phone bills.

My daughter Fiona, a budding interaction designer herself, had to endure my writing when I could have been playing Wii with her. More time for Mario now.

Lastly, and most importantly, without the support of my wife, Rachael King, the creation of this book would have been impossible. All writers need time and space, and those are always her gifts to me. This book is as much a product of her generosity as it is of my words.

Contents

About the Author

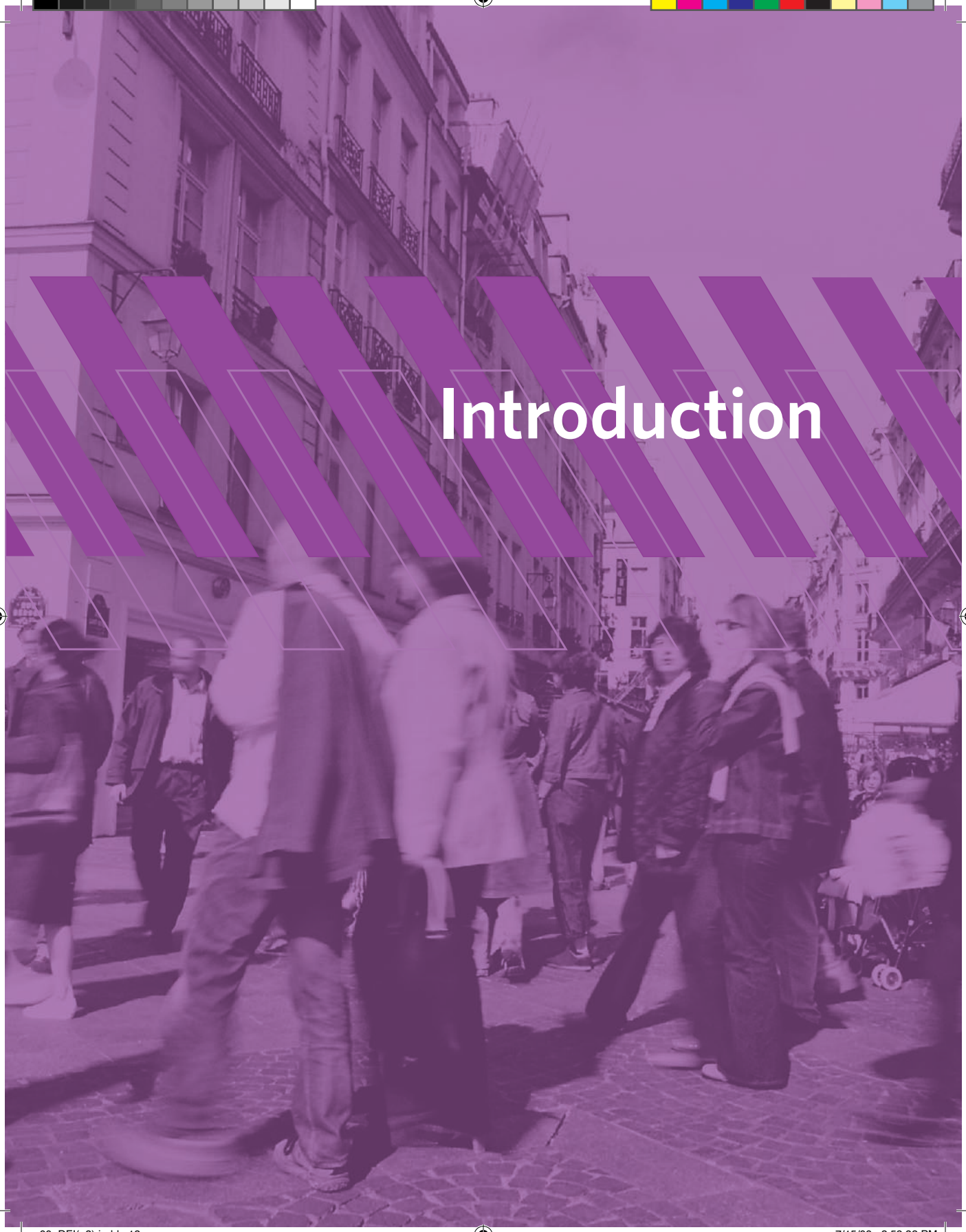
Although he wouldn't hear the term "interaction design" for another decade and a half, Dan Saffer did his first interaction design work as a teenager in the mid-1980s when he designed and ran a dial-up game on his Apple IIe, a 2600-baud modem, two floppy disk drives, and a phone line. And yes, it was in his parents' basement.

He's worked formally in interactive media and product design since 1995 as a webmaster, information architect, copywriter, developer, producer, creative lead, creative director, and, of course, interaction designer. Currently, he's one of the founders and principals of Kicker Studio, a product design consultancy in San Francisco.

Dan has designed a wide range of products, from Web sites to interactive TV services, from mobile and medical devices, to touchscreens, gestural interfaces, and robots. His clients have included Fortune 100 companies, government agencies, and startups.

He holds a Masters in Design, Interaction Design from Carnegie Mellon University, where he also taught interaction design fundamentals.

He lives and works in San Francisco and can be found online at http://www.odannyboy.com and on Twitter at @odannyboy.



Introduction

In the last decade, and especially in the three years since the first edition of *Designing for Interaction* was published, interaction design as a discipline has come into its own. Even people who have never heard of interaction design—which is to say, most people—understand that how their devices work is as important as how they look. A beautiful mobile phone that functions poorly will cause months of frustration. We know, and the popular press has celebrated, that the best products are those that are functionally—and aesthetically—beautiful.

The past several years have also brought us some absolutely wonderful examples of interaction design that have sparked the imagination: Apple's iPhone, Nintendo's Wii, iRobot's Roomba, Microsoft's Surface, Twitter, and social networks like Facebook. More and more, previously "dumb" products are being outfitted with microprocessors, sensors, and networking capabilities, while the Web has matured to a sophisticated platform for applications of all sorts. Desktop applications have become interwoven with the Internet for interesting combinations. Devices can locate themselves in physical space and provide geo-located information. Exploding processing power, cloud computing, and cheap digital storage make all sorts of new products possible.

All of these things mean the rules of interaction design (such as they are) are being rewritten. The paradigms of how we interact with computing devices, such as the desktop metaphor that we've used for around 40 years now, are changing and being added to. We relate to our products—and thus, to each other—in new ways. It's an exciting time to be in this field.

This book is about the discipline that defines how digital products behave. It doesn't contain any code; indeed, I've tried to be as technology and platform agnostic as possible. I've written this book for both new designers who are just getting started, as well as more advanced designers who might want to refine their processes or add to their set of design tools.

What's New in This Edition

This book addresses a fairly serious flaw in the first edition, namely that while there was a lot of good information, there was no process to help new designers put all that information into an order, into practice. In this edition, Chapters 3 through 8 step through a general design process that can be used for a wide variety of projects. Not every step needs to be followed,

and the process is in an ideal order that seldom happens in designing. But at least there is a process.

Additionally, several significant new topics have been added. Design strategy (Chapter 3) is brand new in this edition and I daresay does the best job I've seen in distilling this step (and growing field unto itself) down to its essentials. In the first edition, the translation of research into models and then into concepts was poorly done; this edition addresses that crucial stage. Likewise, there was no mention of design principles, and this was an unfortunate oversight.

Service design, which was its own chapter in the first edition, has been more integrated into the book for two reasons. The first is that service design has become its own area of study. The second reason is that the line between services and products has gotten blurrier. It is difficult to find products, and especially the networked products interaction designers work on, that aren't part of a service of some kind.

Readers of the first edition also asked for references and recommendations to dive deeper into the various topics, so each chapter now has a "For Further Reading" section at the end as well as footnotes to specific articles.

I hope this book is a starting point for your work in interaction design. It is, however, only a book, and books alone can't make you a great designer. Only designing will do that. I urge you to try out everything in this book for yourself, change it as necessary to fit your working style, your company, your users, and the project you're on.

So get to it—there's much to be designed.

San Francisco
June 2009

We become what we behold. We shape our tools, and thereafter our tools shape us.

—Marshall McLuhan

1

What Is Interaction Design?

Every moment of every day, millions of people send e-mail, talk on mobile phones, instant message each other, record TV shows on digital video recorders (DVRs), and listen to music on MP3 players. All of these things are made possible by good engineering. But it's interaction design that makes them usable, useful, and fun.

You benefit from good interaction design every time you:

- Go to an automatic teller machine (ATM) and withdraw cash with a few simple touches on a screen.
- Become engrossed in a computer game.
- Cut and paste cells on a spreadsheet.
- Buy something online.
- Twitter from your mobile phone.
- Update your status on Facebook.

But the reverse is often also true. We suffer from poor interaction design *all around us*. Thousands of interaction design problems wait to be solved—such as when you:

- Try to use the self-checkout at a grocery store and it takes you half an hour.
- Can't get your car to tell you what's wrong with it when it breaks down.
- Wait at a bus stop with no idea when the next bus will arrive.
- Struggle to synchronize your mobile phone to your computer.
- Can't figure out how to set the clock in your microwave oven.

Any time behavior—how a product works—is involved, interaction designers could be involved. Indeed, for the best experience, they *should* be involved.

Back in 1990, Bill Moggridge (**Figure 1.1**), a principal of the design firm IDEO, realized that for some time he and some of his colleagues had been creating a very different kind of design. It wasn't product design exactly, but they were definitely designing products. Nor was it communication design, although they used some of that discipline's tools as well. It wasn't computer science either, although a lot of it had to do with computers and software. No, this was something different. It drew on all those disciplines, but was something else, and it had to do with connecting people through the products they used. Moggridge called this new practice **interaction design**.

In the decades since then, interaction design has grown from a tiny, specialized discipline to one practiced by tens of thousands of people all over the world, many of whom don't call themselves interaction designers and may not even be aware of the discipline. Universities now offer degrees in it, and you'll find practitioners of interaction design at every major software and design firm, as well as in banks such as Wells Fargo, hospitals such as the Mayo Clinic, and appliance manufacturers such as Whirlpool.

COURTESY BILL MOGGRIDGE

Figure 1.1

Bill Moggridge, author of *Designing Interactions* and industrial designer for one of the first laptop computers, the GRiD Compass, coined the term "interaction design" after being talked out of the term "soft-face."

The rise of the commercial Internet in the mid 1990s and the widespread incorporation of microprocessors into machines such as cars, dishwashers, and phones where previously they hadn't been used led to this explosive growth in the number of interaction designers because suddenly a multitude of serious interaction problems needed to be solved. Our gadgets became digital, as did our workplaces, homes, transportation, and communication devices. Our everyday stuff temporarily became unfamiliar to us; the confusion we once collectively had about how to set the clock on the VCR spread to our entire lives. We had to relearn how to dial a phone number and work the stereo and use our computers. It was the initial practitioners of interaction design—mostly coming from other disciplines—who helped us begin to make sense of our newly digitized world and the Internet, and these same people, now aided by new interaction designers, continue to refine and practice the craft as our devices, and our world, grow ever more complex.

What Are Interactions and Interaction Design?

Although we experience examples of good and bad interaction design every day, interaction design as a discipline is tricky to define. In part, this is the result of its interdisciplinary roots: in industrial and communication design, human factors, and human-computer interaction. It's also because a lot of interaction design is invisible, functioning behind the scenes. Why do the Windows and Mac operating systems, which basically do the same thing and can, with some tinkering, even look identical, *feel* so different? Interaction design is about behavior, and behavior is much harder to observe and

understand than appearance. It's much easier to notice and discuss a garish color than a subtle transaction that may, over time, drive you crazy.

An interaction, grossly speaking, is a transaction between two entities, typically an exchange of information, but it can also be an exchange of goods or services. This book is called *Designing **for** Interaction* because it is this sort of exchange that interaction designers try to engender in their work. Interaction designers design *for* the possibility of interaction. The interaction itself takes place between people, machines, and systems, in a variety of combinations.

Three Ways of Looking at Interaction Design

There are three major schools of thought when it comes to defining interaction design:

- A technology-centered view.
- A behaviorist view.
- The Social Interaction Design view.

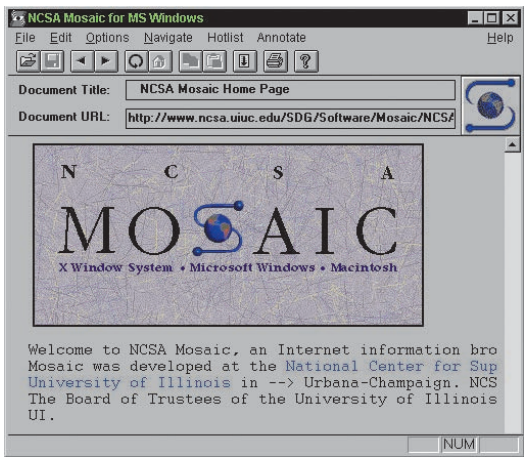


Figure 1.2
Designed by Marc Andreessen, the Mosaic browser (which eventually evolved into Netscape Navigator) was a fantastic piece of interaction design, making the Web accessible to everyday people. It introduced interaction design paradigms still in use today, such as the back button.

What is common about all three views is that interaction design is seen as an art—an applied art, like furniture making; it's not a science, although some tried and true rules have emerged (see Chapter 7). Interaction design is by its nature contextual: it solves specific problems under a particular set of circumstances using the available materials. For example, even though a 1994 Mosaic browser (**Figure 1.2**) was an excellent piece of interaction design, you wouldn't install it on your computer now. It served its purpose *for its time and context.*

Like other applied arts, such as architecture, interaction design involves many methods and methodologies in its tasks, and ways of working go in and out of vogue and often compete for dominance. Currently, a very user-centered design methodology in which products are generated with users is in style, but this hasn't always been the case, and recently these methods

have been challenged (see Chapter 2). Microsoft performs extensive user testing and research; Apple, known for its innovative interaction design, does very little.

The Technology-Centered View

Interaction designers make technology, particularly digital technology, useful, usable, and pleasurable to use. This is why the rise of software and the Internet was also the rise of the field of interaction design. Interaction designers take the raw stuff produced by engineers and programmers and mold it into products that people enjoy using.

The Behaviorist View

As Jodi Forlizzi and Robert Reimann succinctly put it in 1999 in their presentation "Interaction Designers: What we are, what we do, & what we need to know,"[1] interaction design is about "defining the behavior of artifacts, environments, and systems (for example, products)." This view focuses on functionality and feedback: how products behave and provide feedback based on what the people engaged with them are doing.

The Social Interaction Design View

The third, and broadest, view of interaction design is that it is inherently social, revolving around facilitating communication between humans through products. This perspective is sometimes called Social Interaction Design. Technology is nearly irrelevant in this view; any kind of object or device can make a connection between people. These communications can take many forms; they can be one-to-one as with a telephone call, one-to-many as with a blog, or many-to-many as with the stock market.

Why Interaction *Design*?

The term "design" can be difficult to get a handle on. Consider this infamous sentence by design history scholar John Heskett: "Design is to design a design to produce a design."

1 Download it online at http://goodgestreet.com/docs/AIGAForlizzi_Reimann2001.pdf

People have many preconceived notions about design, not the least of which is that design concerns only how things look: design as decoration or styling. And while there is nothing wrong with appealing aesthetics, design can be more than that. Communication (graphic) and industrial design bring ways of working that interaction designers embrace as well. Here are some of the approaches that interaction design employs:

Focusing on Users

Designers know that users don't understand or care how the company that makes a product is run and structured. They care about doing their tasks and achieving their goals within their limits. Designers are advocates for end users.

Finding Alternatives

Designing isn't about choosing among multiple options—it's about creating options, finding a "third option" instead of choosing between two undesirable ones. This creation of multiple possible solutions to problems sets designers apart. Consider, for example, Google's AdWords. The company needed advertising for revenue, but users hated traditional banner ads. Thus, designers came up with a third approach: text ads.

Using Ideation and Prototyping

Designers find their solutions through brainstorming and then, most important, building models (**Figure 1.3**) to test the solutions. Certainly, scientists and architects and even accountants model things, but design involves a significant difference: design prototypes aren't fixed. Any particular prototype doesn't necessarily represent *the* solution, only *a* solution. It's not uncommon to use several prototypes to create a single product. Jeff Hawkins, designer of the original PalmPilot, famously carried around small blocks of wood, pretending to write on them and storing them in his shirt pocket until he came upon the right size, shape, and weight for the device.

COURTESY CHERYL RIEDEL

Figure 1.3

Interaction designers should plan to create (and throw away) a variety of prototypes of various fidelities to test their concepts.

Collaborating and Addressing Constraints

Few designers work alone. Designers usually need resources (money, materials, developers, printers, and so on) to produce what they dream up, and these resources come with their own constraints. Designers seldom have carte blanche to do whatever they want. They must address business goals, compromise with teammates, and meet deadlines. Designing is almost always a team effort.

Creating Appropriate Solutions

Most designers create solutions that are appropriate only to a particular project at a particular point in time. Designers certainly carry experience and wisdom from one project to the next, but the ultimate solution should uniquely address the issues of that particular problem. This is not to say that the solution (the product) cannot be used in other contexts—experience tells us it can and will be—but that the same exact solution cannot (or shouldn't anyway) be exactly copied for other projects. Amazon has a great e-commerce model, but it can't be exactly replicated elsewhere (although pieces of it certainly can be); it works well within the context of the Amazon site. Design solutions have to be appropriate to the situation.

Drawing on a Wide Range of Influences

Because design touches on so many subject areas (psychology, ergonomics, economics, engineering, architecture, art, and more), designers bring to the table a broad, multidisciplinary spectrum of ideas from which to draw inspiration and solutions.

Incorporating Emotion

In analytical thinking, emotion is seen as an impediment to logic and making the right choices. In design, products without an emotional component are lifeless and do not connect with people. Emotion needs to be thoughtfully included in design decisions. What would the Volkswagen Beetle be without whimsy?

A (Very) Brief History of Interaction Design

There's a tendency to think that interaction design began around the time that Bill Moggridge named it, in 1990, but that's not really true. Interaction design probably began, although obviously not as a formalized discipline, in prerecorded history, when Native Americans and other tribal peoples used smoke signals to communicate over long distances, and the Celts and Inuit used stone markers called cairns or inuksuit as landmarks, to communicate over time (**Figure 1.4**).

COURTESY ISTOCKPHOTO

Figure 1.4

A modern cairn. In ancient times, cairns were used for many purposes: to mark mountain summits, as directional markers, and as indicators of burial sites.

1830s to 1940s

Many centuries later, in the mid 1830s, Samuel Morse created a system to turn simple electromagnetic pulses into a language of sorts and to communicate those words over long distances. Over the next 50 years, Morse code and the telegraph spread across the globe (**Figure 1.5**). Morse not only invented the telegraph, but also the entire system for using it: everything from the electrical systems, to the mechanism for tapping out the code, to the training of telegraph operators. This didn't happen overnight, naturally, but the telegraph was the first instance of communication technology that, unlike the printing press, was too sophisticated for a small number of people to install and use. It required the creators to design an entire system of use.

COURTESY ISTOCKPHOTO

Figure 1.5

Morse code transmitter. The telegraph was the first technology system that wired the world—the so-called "Victorian Internet."

Similarly, other mass communication technologies, from the telephone to radio to television, required engineers to design systems of use and interfaces for the new technologies. And these systems and interfaces were needed not only for the receiving devices—the telephones, radios, and television sets—but also for the devices used to create and send messages: the telephone switches, microphones, television cameras, control booths, and so on. All of these components required interaction design, although it certainly wasn't called that at the time. Indeed, it is very common for the first

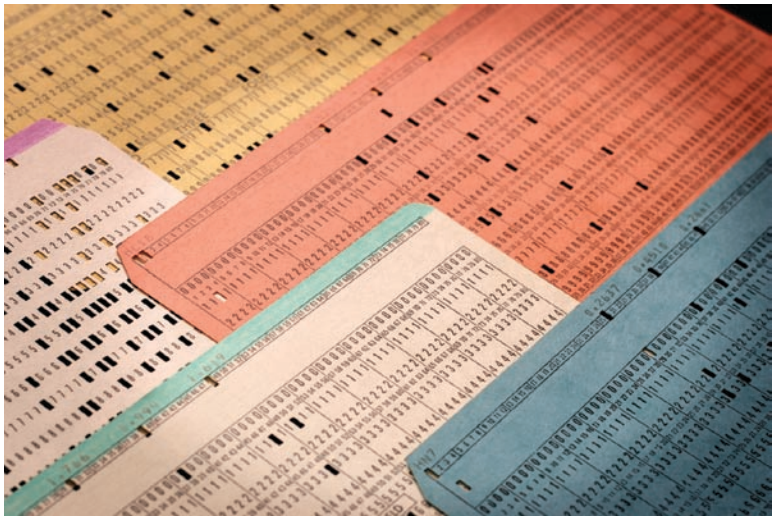
practitioners of interaction design in any new platform or medium to be the engineers who created the technology itself.

But the machines that fueled these technologies were, for the most part, just that: machines. They responded to human input, certainly, but not in a sophisticated way. They didn't have any awareness that they were being used. For that, we needed computers.

1940s to 1960s

The first wave of computers—ENIAC and its ilk—were engineered, not designed. Humans had to adapt to using them, not vice versa, and this meant speaking the machines' language, not ours. Entering anything into the computer required days plugging in cables or, in later machines, hours preparing statements on punch cards or paper tape for the machine to read. These paper slips were the interface (**Figure 1.6**). Engineers expended very little design effort to make the early computers more usable. Instead, they worked to make them faster and more powerful, so the computers could solve complicated computational problems.

At the same time as these developments were occurring in the computing field, other disciplines that eventually informed interaction design were



COURTESY ISTOCKPHOTO

Figure 1.6

Punch cards—one of the first interfaces with computers, as well as a means of data storage. By the 1980s, almost all of them had been phased out by command-line or GUI interfaces.

growing, too. Engineers and industrial designers such as Henry Dreyfuss created the new field of human factors, which focused on the design of products for different sizes and shapes of people. The field of ergonomics focused on workers' productivity and safety, determining the best ways to perform tasks. Cognitive psychology, focusing on human learning and problem solving, experienced a resurgence, led by such academics as Allen Newell and George Miller.

In 1945, *Atlantic Monthly* published a seminal article titled "As We May Think"[2] (reportedly written in 1936) by Vannevar Bush, in which he introduced the Memex, a microfilm-based device for storing books, records, and communications, which is mechanized so that it may be consulted with exceeding speed and flexibility.

It consists of a desk, and while it can presumably be operated from a distance, it is primarily a piece of furniture. On the top are slanting translucent screens, on which material can be projected for convenient reading. There is a keyboard, and sets of buttons and levers. Otherwise it looks like an ordinary desk.

The Memex (**Figure 1.7**) was Bush's concept for augmenting human memory. While just a concept, it was the first imagining of hypertext, and one of the first for a desktop computing system. It has influenced generations of interaction designers since, starting with Douglas Engelbart and Ted Nelson in the 1960s.

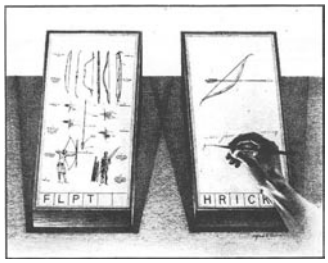


Figure 1.7

One of the drawings of Vannevar Bush's Memex device as it appeared in *Life* magazine in 1945. Note the stylus—an input device decades ahead of its time.

1960s to 1970s

As computers became more powerful, engineers began to focus on the people using computers in the 1960s, and began to devise new methods of input and new uses for the machines. Engineers added control panels to the front of computers, allowing input through a complicated series of switches, usually in combination with a set of punch cards that were processed as a group (batch processing).

2 Read it online at http://www.theatlantic.com/doc/194507/bush

In 1960, Ted Nelson started his Project Xanadu, with the goal of creating computer networks with simple user interfaces. While it never really came to fruition, it was the first attempt at a hypertext system. Nelson, in fact, coined the term "hypertext" in 1963.

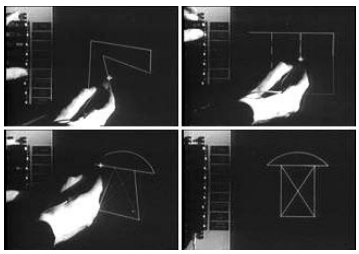


Figure 1.8
Ivan Sutherland's Sketchpad. One of Sketchpad's innovations was master drawings of which users could create duplicates. If the user changed the master drawing, all the instances of the drawing would change as well.

1963 also brought Ivan Sutherland's Sketchpad (**Figure 1.8**), the first computer program to utilize a fully graphical user interface and a light pen for input. Using Sketchpad, users could draw both horizontal and vertical lines and combine them into figures and shapes. Sutherland in 1968 created The Sword of Damocles, which is widely considered to be the first virtual reality system. (The head-mounted display worn by the user was so heavy it had to be suspended from the ceiling, thus inspiring the name.)

Sometime around 1965, the first "killer application," e-mail, was invented as a way for multiple users of a time-sharing mainframe computer to communicate. By 1966, e-mail had expanded to allow users to send messages between different computers. By 1971, e-mail was being sent across ARPANET, the precursor to the Internet. Ray Tomlinson, who created the e-mail standards still in use (such as the @ symbol in e-mail addresses), sent the first e-mail between different host systems, reportedly something insignificant like "QWERTYUIOP."

The ARPANET (Advanced Research Projects Agency Network) was developed by ARPA of the United States Department of Defense and was the predecessor of the global Internet. Conceived as the "Intergalactic Computer Network" in 1962 by J.C.R. Licklider, the first two links of the network (UCLA and Stanford) connected on November 21, 1969. While ARPANET certainly wasn't a design milestone, its creation lead to the platform and medium that caused interaction design to flourish: the Internet.

In 1968, Doug Engelbart did a 90-minute presentation that is now known as "The Mother of All Demos"[3] (**Figure 1.9**). In it, Engelbart showed the work he'd been doing for the previous several years, essentially creating the next two decades of interaction design. As well as being the first public

3 Watch it online at http://sloan.stanford.edu/MouseSite/1968Demo.html

demonstration of the mouse, Engelbart demonstrated an incredible variety of interaction design paradigms we now take for granted, such as point and click, hyperlinks, cutting and pasting, and networked collaboration.

Figure 1.9

December 9, 1968, was Doug Engelbart's "Mother of All Demos" at the Fall Joint Computer Conference in San Francisco. Engelbart demonstrated a mouse, video conferencing, e-mail, and hypertext on the NLS (oNLine System) to 1,000 attendees.

Many of these paradigms were to find a home at Xerox PARC (Palo Alto Research Center), founded in 1970. The head of Xerox PARC, Bob Taylor, urged employees to think of computers not as just processing devices, but instead as communication devices.

Xerox PARC remains legendary. Its contributions to the field, many of which are contained in its signature products the Xerox Alto (**Figure 1.10**) and the Xerox Star, are everything from windowing and icons and the desktop metaphor to WYSIWYG text editing. Employees included Alan Kay, who conceived of the first laptop computer, the Dynabook, in 1968; Larry Tesler and Tim Mott, who conceived of the desktop metaphor and such now-standard interactions as cut-and-paste; and Robert Metcalfe, who invented Ethernet networking in 1973.

COURTESY MARCIN WICHARY

Figure 1.10

Xerox Alto. One of the first personal computers, and the first to use the desktop metaphor.

Famously, Steve Jobs got a demo of the Xerox Star and proceeded to include its innovations into Apple's subsequent computers, the Lisa and, eventually, the Macintosh.

In the mid-to-late 1970s, experiments like Myron Krueger's VIDEOPLACE explored virtual reality experiences and gestural interfaces, and the first touchscreen devices became commercially available.

The 1970s also began the computer gaming industry with games such as Pong (1972) and the Atari 2600 gaming console (1977). This reflected another major trend in the 1970s: the shifting focus from the computer

itself—the hardware—to the software that runs it, particularly software that was not designed by computer scientists and engineers for themselves or trained operators. Designers and engineers in the 1970s refined and expanded the command-line interface (which had begun in the 1950s) into such industry-defining software, as VisiCalc, the first spreadsheet software, introduced in 1979, and WordStar, a popular word-processing program introduced in 1978 (**Figure 1.11**).

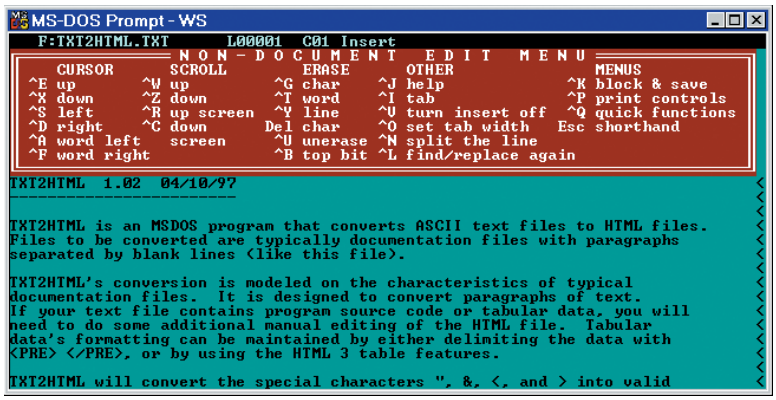


Figure 1.11

WordStar and its ilk were some of the first pieces of commercial software that weren't designed by programmers for programmers. WordStar dominated the word processing market from its release in 1978 until the early 1990s, when it was surpassed by Microsoft Word.

1980s

This new emphasis on users came to fruition in the early 1980s with the explosion of the graphical user interface—spearheaded by Apple Computer, first in the Lisa (**Figure 1.12**) and then in the Macintosh—to a mass audience. Like at Xerox PARC, the interaction design of the Lisa and Macintosh was a group effort, featuring designers such as Joy Mountford, Jef Raskin, and Bill Atkinson.

The 1980s was the era of the personal computer. For the first time, most people working with computing devices were working with their own, and thus had a more one-to-one relationship with one than in previous decades. 1981 also saw some of the first portable computers, such as the Osborne 1. The increasing memory and power of the devices allowed for more sophisticated software such as Mitch Kapor's Lotus 1-2-3 (1983).

COURTESY MARCIN WICHARY

Figure 1.12

Apple Lisa was a precursor (of sorts) to the Macintosh, although more powerful and, in many ways, more advanced. It was, however, a commercial failure.

This increasing sophistication and power was demonstrated most capably in the surge of so-called "video" or "arcade" games. Gaming consoles such as the Sega Genesis (1989) and the Super Nintendo Entertainment System (1990) brought unprecedented graphics and computing power to a mass audience. This era also featured game designers such as the legendary Shigeru Miyamoto, the "Father of Modern Video Games" and creator of Mario, Legend of Zelda, and Donkey Kong. Gaming provided a new set of parallel interaction design paradigms that exist alongside the more "traditional" or "professional" ones for the desktop. (Mobile and touchscreen devices are other similar parallel tracks.)

In the mid-1980s, bulletin board systems (BBSs) like The WELL (1985) and Prodigy (1988) sprung up so that people could leave e-mail and messages for one another on remote computers using dial-up modems.

In the late 1980s, Mark Weiser and John Seely Brown at Xerox PARC began putting together the frameworks and definitions for what would become known as ubiquitous computing, or **ubicomp**. It's taken about two decades, but the era of ubicomp has likely already begun (see Chapter 9).

1990s

The era of networked computing, and the beginning of interaction design as a formal discipline, began in earnest during the 1990s. The World Wide Web, which allowed anyone to easily publish hypertext documents accessible to anyone with a modem worldwide, and the mass adoption of e-mail, brought the need for better interaction design to the forefront. Marc Andreessen's Mosaic browser (1993) was an important piece of interaction design, introducing such paradigms as the back button.

It is no exaggeration to state that the advent of the commercial, public Internet changed the world and the relationship of humans to computing devices and even to information. The early Web was as much a sandbox for new interactions as was the desktop a decade before, if not more so. The Web, along with technologies such as Adobe's Flash, allowed for experimentation on a grand scale, and for a time, everything—including general controls like scrollbars and buttons—were up for grabs. Eventually, in the late 1990s, standards began to emerge and the Web stabilized as a platform.

At the same time, engineers and designers began building sensors and microprocessors, which were getting smaller, cheaper, and more powerful, into things that weren't considered computers: cars, appliances, and electronic equipment. Suddenly, these physical objects could demonstrate kinds of behavior that they previously couldn't; they could display an "awareness" of their environment and of how they were being used that was previously inconceivable. Cars could monitor their own engines and alert drivers to problems before they occurred. Stereos could adjust their settings based on the type of music being played. Dishwashers could lengthen their wash cycles depending on how dirty the dishes were. All these behaviors needed to be designed and, most important, communicated to the human beings using the objects.

Other pieces of technology facilitated interactions among people, mostly in the entertainment space. Karaoke spread from bars in China and Japan to the United States (**Figure 1.13**). Arcade video games like Dance Dance Revolution allowed expression in front of crowds. Multiplayer games on computers and game consoles like the Sony PlayStation facilitated competition and collaboration in new ways. Online communities like EverQuest and The Sims Online incorporated sophisticated economies that rivaled those of offline countries.

COURTESY ISTOCKPHOTO

Figure 1.13

Although the butt of jokes in the US, the karaoke machine is a surprisingly rich example of interaction design. It provides a way to communicate emotionally with friends.

Mobile phones and devices—which had existed since the 1980s—enjoyed explosive market growth in the 1990s. Today, billions of customers carry these devices with them. Starting as simply a means of making calls on the go, mobile phones can now contain myriad digital features that rival those of desktop computers. Personal digital assistants (PDAs) got off to a shaky start with the failure of Apple's Newton in 1995, but by the end of the decade, they had gained traction with devices like the PalmPilot and BlackBerry PDAs.

2000s to Present

The turn of the millennium also coincided with the era of social software and the beginning of the era of ubiquitous computing. No longer did many people have a one-to-one relationship with devices, but instead had access to many devices able to interact with each other and the Internet over a network. By 2003, laptops had started outselling desktop systems. As of this writing (2009), nearly as many people access the Web via a mobile device as with a traditional desktop or laptop, and that number is likely to be surpassed shortly.

As the Internet matured, so did the technologies creating and driving it. Since the end of the 1990s, the Internet has become less about reading content than about doing things: executing stock trades, making new (and

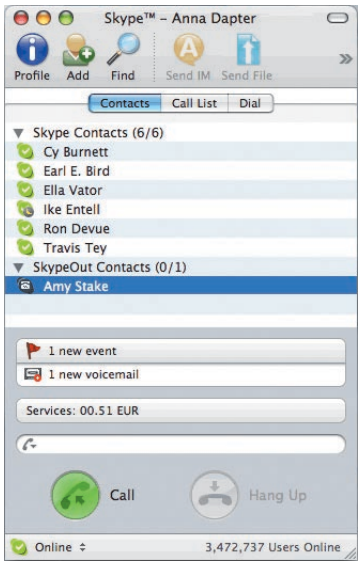


Figure 1.14

Skype takes a familiar paradigm, the buddy list from instant messaging, and couples it with a new technology, Voice over IP (VoIP), in order to make phone calls via the Internet.

finding old) acquaintances, selling items, manipulating live data, sharing photos, making personal connections between one piece of content and another. The Internet also provides several new ways of communicating, among them instant messaging, Voice over Internet Protocol (VoIP) (**Figure 1.14**), and Twitter.

The Internet has become a platform for applications, in much the same way that Microsoft DOS once was, but these applications can take advantage of the many features of the Internet: collective actions like the SETI@Home project in which people compete to see who can find extraterrestrial activity first, data that is collected passively from large numbers of people as with Amazon's "People who bought this also bought..." feature, far-flung social communities such as that of online photography site Flickr, aggregation of many sources of data in XML and RSS feeds, near real-time access to timely data like stock quotes and news, and easy sharing of content such as blogs and YouTube.

Access to the Internet, through broadband connections and wireless networks on portable devices, is changing the types of interactions we can have and where we can have them. Our cities and towns are becoming platforms and data sources for geo-located services. Services themselves are being affected by interaction design (see "Products and Services" later in this chapter).

Gestural interfaces and touchscreen devices such as Nintendo's Wii and Apple's iPhone have ushered in a new era of interaction design, where taps on a screen or gestures in space are becoming a new set of commands for our devices.

There's never been a better time to be an interaction designer. The discipline's future (see Chapter 9) contains both many challenges and many possibilities.

Marc Rettig on Interaction Design's History and Future

Marc Rettig is a designer, educator, and researcher, as well as founder and principal of Fit Associates. He has taught at Carnegie Mellon's Graduate School of Design (where he held the 2003 Nierenberg Distinguished Chair of Design) and the Institute of Design, IIT, in Chicago. Marc served as chief experience officer of the user experience firm HannaHodge, and was a director of user experience at Cambridge Technology Partners.

When does the history of interaction design begin?

I'll pick the work at Xerox PARC on the Star interface as a very early example of self-conscious interaction design, the publication of which influenced others to begin working in a similar way. As just one example, the idea of associating a program with a picture was born there. We call them icons, and forget what a breakthrough connection between interface element and underlying meaning that once was. That was the early-to-mid 1970s, and the Star papers are still great reading.

What fields have had the greatest influence on interaction design?

As it is currently practiced? Well, software development and graphic design. To some extent, industrial design. A dab of psychology and human factors. A dab of business.

What I imagine we need more of: filmmaking and theater, biology, counseling and therapy (the professionals at acquiring and checking an empathetic point of view), maybe anthropology. And especially linguistics—some new branch of linguistics that nobody is yet carving out: the linguistics of designed interactions.

What can interaction designers learn from noninteractive tools?

I'd like to spin the question slightly by observing that to an interaction designer, watching a tool in use is the same as observing a conversation. Everything, in a sense, has its inputs and outputs. From that point of view, the boundary between "interactive" and "noninteractive" tools starts to dissolve.

Interaction design is largely about the meaning that people assign to things and events, and how people try to express meanings. So to learn from any tool, interactive or not, go watch

Marc Rettig on Interaction Design's History and Future *(continued)*

people using it. You'll hear them talk to the tool. You'll see them assign all sorts of surprising interpretations to shapes, colors, positioning, dings, dents, and behaviors. You'll see them fall in love with a thing as it becomes elegantly worn. You'll see them come to hate a thing and choose to ignore it, sell it, or even smash it. And I guarantee you won't have to do much of this before you encounter someone who makes a mental mapping you would never dream possible. And you'll learn from that.

I've been using tea kettles as an example in some of my teaching, because on the one hand kettles are so familiar to us, and they're only interactive in a borderline, predictable, mechanical sort of way. But once you start to examine the meanings involved with kettles in use, you realize they have things to say that people would love to know, but most designs don't allow them to be said. "I'm getting hot, but I have no water in me." "My water is a good temperature for a child's cocoa." "I'm too hot to touch." "I need to be cleaned." And so on. I'd love the chance to take a serious interaction design approach to something like a tea kettle.

A Stew of Disciplines

Interaction design as a formal discipline has been around for less than two decades. It's a young field, still defining itself and figuring out its place among sister disciplines such as information architecture (IA), industrial design (ID), visual (or graphic) design, user experience (UX) design, and human factors. In addition, some of these other disciplines are also new and still discovering their boundaries as well, or are radically changing to accommodate changing design landscape. **Figure 1.15** attempts to clarify the relationships between them.

As you can see, most of the disciplines fall at least partially under the umbrella of user-experience design, the discipline of looking at all aspects—visual design, interaction design, sound design, and so on—of the user's encounter with a product, and making sure they are in harmony.

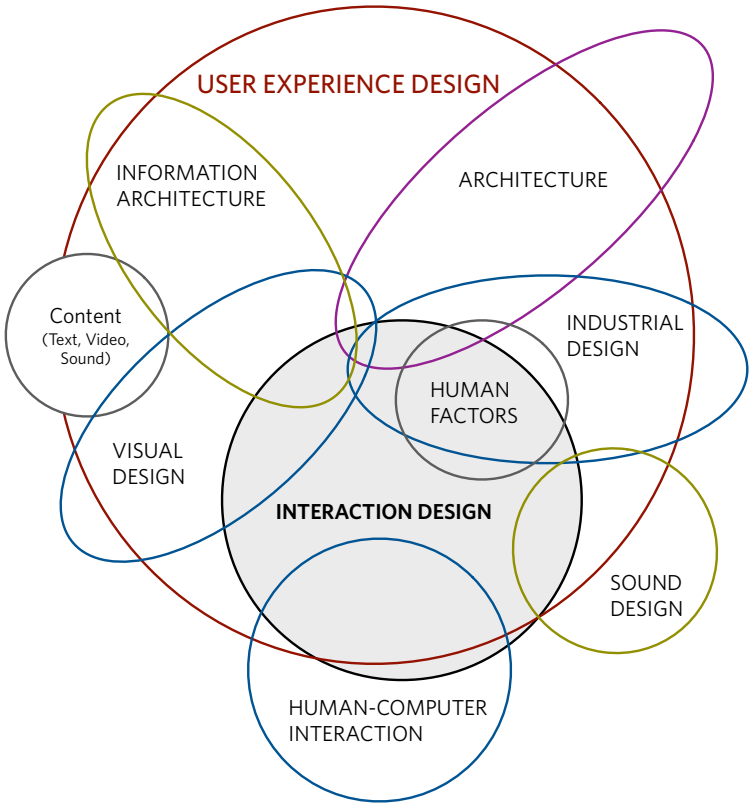


Figure 1.15

The disciplines surrounding interaction design.

Information architecture is concerned with the structure of content: how to best organize and label content so that users find the information they need. Yahoo, with its dozens of labeled and categorized content areas, offers an excellent illustration of information architecture. Visual design is about creating a visual language to communicate content. The fonts, colors, and layout of user interfaces and printed materials like this book provide examples of visual design. Industrial design is about form—shaping objects in a way that communicates their use while also making them functional. Physical objects like furniture, kitchenware, and mechanical objects illustrate industrial design. Human factors ensure our products conform to the limitations of the human body, both physically and psychologically. Human-computer interaction is closely related to interaction design, but its methods are more quantitative, and its methods are more those of engineering and

computer science than of design. Architecture concerns itself with physical spaces: their form and use ("program"). Sound design defines a set of noises, spoken word, or music to create an aural landscape.

It's easy to see why people are confused!

Although these disciplines are separate, as the figure illustrates, they still overlap a great deal. In fact, where the disciplines overlap can be major areas of practice, such as interface design, where visual and interaction design meet; or navigation, where visual and interaction design meet information architecture.

The best products involve multiple disciplines working in harmony. What is a laptop computer except a blend of the fruits of many of these disciplines? Separating them can be nearly impossible.

You'll also notice that many of these disciplines have parts that lie outside the user experience realm. This is because many of these disciplines have tasks that have to do with getting their designs produced, developed, and built, and those tasks may have little to do with what the user experiences.

It is also important to note that not every organization needs a specialist working in each discipline; within an organization, one person, who might be called anything from an information architect to a user-interface engineer, can—and probably will—shift back and forth as needs require. It's the role that is important, not the title. The "imagineer" at Disney might do a job similar to that of the "user-interface architect" at a startup company.

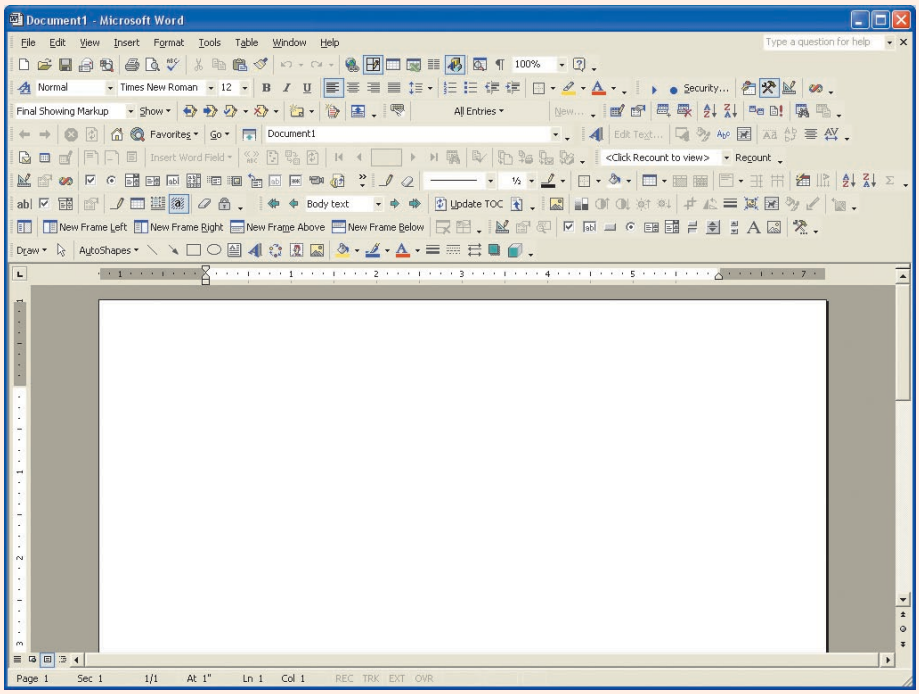
Case Study: Microsoft Office 2007

The Company

Microsoft, the world's largest software company.

The Problem

In the early 2000s, it was clear to many inside Microsoft that something had to be done about their best-selling, nearly ubiquitous software suite Microsoft Office. The original interaction and interface design, created a decade before, was not scaling well. New features were being hidden by the interface, and even features users had requested and had been put into new versions of the product couldn't be found by those very same users. The software appeared bloated, inefficient, and unwieldy. For example, 50 menu items and 2 toolbars from Microsoft Word 1.0 had ballooned to 260 menu items and over 30 toolbars by Word 2003.



Case Study: Microsoft Office 2007 *(continued)*

The Process

The Microsoft design team started by analyzing anonymous data collected about how people were using Office 2003. They looked for two important things: desirable features with low usage numbers (which meant people couldn't find them) and frequently-used features that were hard to get to (which meant people really wanted them). They focused on the design principle (see Chapter 6) "Use of a broader set of tools" and did several years of iterative prototyping to come up with a new set of interaction design paradigms for users.

The Solution

Microsoft Office 2007 has literally 1000 enhancements to it, all of which take up less screen space than previous versions. One main (and controversial) UI change was the Ribbon (pictured), which clusters pieces of functionality at the top of the screen in large, easy-to-click targets. Another innovation was known as "the Minibar," which appeared near objects that were highlighted and allowed users to quickly modify the selection without having to fiddle with menus or the Ribbon. The new design has been a best-seller, and the headline for the review in the *New York Times* read "From Bloated to Sleek."