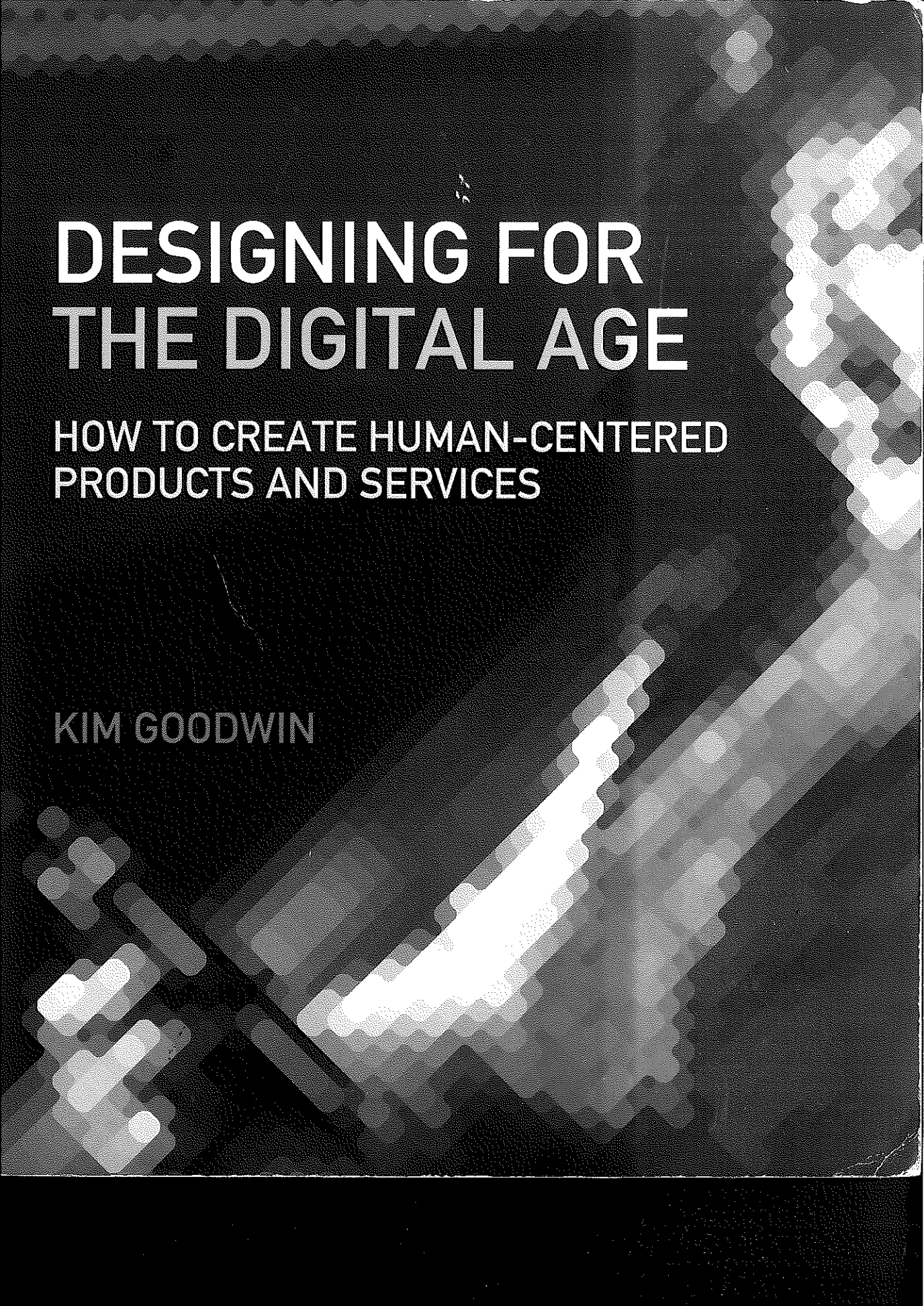
DESIGNING FOR
THE DIGITAL AGE
HOW TO CREATE HUMAN-CENTERED
PRODUCTS AND SERVICES
KIM GOODWIN

CHAPTER 1

Goal-Directed Product and Service Design

To a greater extent than any other creature, we humans shape the world around us to suit ourselves. Some of that shaping is unintentional, but much of it is deliberate. We create our environments by constructing buildings, roads, furnishings, and landscapes. We make our daily lives easier and more enjoyable by inventing tools, from kitchen utensils and earth-to-orbit spacecraft to social networking and enterprise-spanning IT systems. We communicate with one another in text, imagery, motion, and sound. We even attempt to craft perfect experiences in retail settings and amusement parks. This intentional shaping of the world for mass consumption is often referred to as **design**.

Clearly, "design" is an incredibly broad term. Do choosing what color to paint your bedroom, sculpting the exterior of a car, and planning a complex application's technical architecture all have equal claims to the word? People outside of design professions have difficulty drawing the line, and there are so many philosophies and assumptions attached to it that even designers seldom agree on exactly what "design" is.

All of this explains why most design books begin with some definition of the word. For the purposes of this book, at least, **design is the craft of visualizing concrete solutions that serve human needs and goals within certain constraints.**

Visualizing concrete solutions is the essence of design. These solutions could be tangible products, such as buildings, software, consumer electronics, or advertisements, or they could be services that are intended to provide a specific sort of experience. The inherent aptitude—the drive, even—to imagine the desired end result and express it in a tangible way is what separates designers from non-designers. This doesn't mean that all designers must be good at illustration; I have known many fine designers whose drawing skills were limited. What designers must excel at is looking at a blank surface and filling it with believable representations of an end product,so that other people can see, understand, and eventually build it. Building it is a separate task; designers don't build products any more than architects build houses. Instead, they provide precise instructions so that builders can focus on accomplishing the end result.

Design is the craft of visualizing concrete solutions that serve human needs and goals within certain constraints.

Design is a **craft** because it is neither science nor art, but somewhere in between. Science is about understanding how the universe works and why it works that way. Design, while it is informed by scientific learning about human senses, cognition, and ergonomics, focuses on understanding only to the extent that it is necessary to solve the problem at hand. Art is about creating an end product that, above all, expresses the inner vision of the artist. Design is not about expressing the designer's point of view, but it is very much about creation.

In order for design to be design and not art, it must **serve human needs and goals**. All designed artifacts have a purpose. Good design helps humans accomplish something in an efficient, effective, safe, and enjoyable way. Designers draw on fields like ergonomics and HCI (the study of human-computer interaction) to increase efficiency and minimize the potential for injury. At the same time, designers strive to go beyond the simply functional, since pleasure and aesthetic satisfaction are also important human goals.

Finally, design always happens **within certain constraints**. There is no such thing as unconstrained design. Unconstrained classroom exercises may teach imagination, but they do not accurately represent the problem-solving nature of design. Time and cost are always factors on even the most ambitious projects. Designers are also constrained in some way by their materials; physical materials have immutable properties, and even the digital medium introduces limitations due to its very lack of a physical nature. Other common constraints include regulatory requirements, competitive pressures, and the various desires of the people bankrolling the project.

Mind you, this definition of design still encompasses a tremendous range of intentionally created artifacts, environments, and processes—types of things humans have been designing for a hundred years or more. Surely, we ought to have this design thing figured out by now. Perhaps this would be the case if it weren't for an assortment of technologies based on silicon chips. Our increasingly digital age has added a host of new challenges that traditional design, manufacturing, and business mind-sets simply are not equipped to address.

Digital Product and Service Design

This book focuses on the design of the products and services unique to the digital age, including any system or service enabled (at least in part) by a microprocessor. Digital systems include everything from a simple digital alarm clock to complex scientific equipment or supply

chain management software. Digitally enabled services might encompass anything from eBay (a service that lets people sell items online) to a comprehensive set of customer touch points for an airline, including its Web site, automated telephone systems, human customer service, and airport check-in.

Although I emphasize the digital realm, the methods described in this book have been applied with equal success to non-digital problems. Over the years, I've even heard from non-designers who have used the basic principles to develop everything from church social events to employee benefits programs.

Some people refer to human-centered product and service design as **experience design**, but I would argue that this term is presumptuous; we can design every aspect of the environment to encourage an optimal experience, but since each person brings her own attitudes, behaviors, and perceptions to any situation, no designer can determine exactly what experience someone has. For this reason, I refer to product and service design—or simply product design, as a service is still the end product of the design effort—throughout the book.

Designing complex products and services requires the talents of several closely related design disciplines, usually some combination of interaction design, graphic and information design, and industrial design. The graphic and industrial design professions are long established, so I won't define them here, but interaction design is still new enough that degrees in the discipline only started becoming available in the 1990s.

Interaction design is a discipline focused on defining the form and behavior of interactive products, services, and systems. Interaction design answers questions such as:

- What activities does the product or service support, and how?
- What workflow provides the best way for users to accomplish their goals?
- What information do users need at each point in that process?
- What information does the system need from users?
- How will users move from one activity to another?
- How is functionality segmented and manifested?

Because interaction design is focused on what people want to do as well as how they can best accomplish it, it's common for interaction design to affect **product definition**, which is about *what functionality a product has* (as opposed to defining how that functionality is manifest, which is what most people see as the role of design).

Interaction design is often confused with related disciplines known as HCI, human factors or, informally, usability. Training in these fields emphasizes evaluative techniques rather than creative problem-solving skills or methods for generating solutions, which are the focus in design. The line between these professions and interaction design is fuzzy because many people have found their way to interaction design from these fields. Although interaction designers must be versed in the principles of HCI, most interaction designers find more in common with graphic designers and industrial designers than with evaluation-focused HCI professionals. The two approaches result in a difference in worldview much like the one between software engineering and quality assurance: complementary, but not at all interchangeable.

Interaction design may also be confused with Web site information architecture (IA). This field is partially rooted in library science, a discipline

Goal-Directed Design assumes the best way to design a successful product is to focus on achieving goals.

that has long focused on how to categorize and organize information for easy retrieval. Some information architects may disagree, but I argue that for all practical purposes, IA is a specialized subset of interaction design. The methods described in this volume work very well for information architecture, though there's no harm in supplementing them with card sorting and other IA techniques.

Goal-Directed Design

Goal-Directed Design[1] is the approach to product and service design developed at Cooper, a leading design consultancy. Its fundamental premise is that the best way to design a successful product is to focus on achieving goals. Although the rhetorical emphasis is on user goals, the method also incorporates the goals of the customers (people who purchase but don't use a system) and of the business creating the product or service. Goal-Directed Design encompasses the design of a product's behavior, visual form, and physical form. Methods for all three are covered throughout the book.

Origins of Goal-Directed Design

The firm's founder, software inventor Alan Cooper, began to develop the kernel of the method when he first started using a sort of proto-persona in 1983. Based on what he learned from informal interviews with seven or eight users, Alan would mentally walk through different interactions he was coding by pretending to be an end user. He would ask himself why he would be performing a certain task in the first place, what he would know at the beginning of the task, and what he was more likely to figure out as he went along. Alan continued to use this mental play-acting approach for more than a decade. Because he was inventing software that he would sell as a finished product, this purely internal approach worked quite well until Alan began consulting with companies about the design of their own products.

In 1995, on a project with Sagent Technologies, Alan and designer Wayne Greenwood found they needed a way to communicate with the client about user behaviors and needs. Alan's experience with his mental constructs based on real users and Wayne's previous experience at T/Maker, where he often invoked the entirely fictitious "Aunt Edna" to encourage engineers to think about less-skilled users, led them to create a small cast of characters to represent various types of people

1. Goal-Directed Design® is a registered trademark of Cooper (www.cooper.com), used with permission.

(see Figure 1.1). Thanks to a prior acquaintance with Alan, the client was willing to suspend judgment about this peculiar communication method. It wasn't long before the team at Sagent adopted these new models, making conversations about product design and functionality much easier than they had been. The product was a tremendous success. It became obvious to Alan and Wayne that these *dramatis personae* should play a role on other projects, too, and so personas were born.

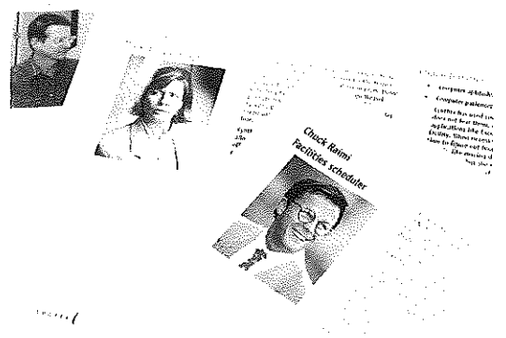

Figure 1.1. Rob, Cynthia, and Chuck were the first real personas.

Over the next five years or so, Alan and about a dozen of us working at Cooper took this powerful idea and used it as the basis to develop a rigorous method, which encompasses everything from planning and conducting research to generating, iterating, and communicating detailed design. The advancement of the method was an informal collective effort; we all paid close attention to which approaches were most (and least) successful, then shared those with our colleagues.

The most fundamental aspects of the method were well established by 2000 or so, at which point we began to formalize our techniques in training courses for new hires and for our clients. The cliché about teaching something being the best way to learn it is true; by articulating the thought processes of our most successful designers and trying to teach them to our newest staff members, we all became even more consistent in our ability to produce great results on a predictable timeline. As our thought processes and rationale became increasingly clear to our clients, it also became easier to persuade them that a particular design direction was the right one. Our clients became more accepting of our designs and more successful in building them.

Of course, any good method is a living thing that continues to evolve and grow. By now, dozens of Cooper designers have left their mark on the method in one way or another, and there's no doubt that dozens more will leave their mark in years to come. We've used these methods to design such a wide array of products and services that we seldom encounter situations the method doesn't have tools for, but that doesn't mean there aren't plenty of them left. When we do see a new problem, we always try new approaches, see what works and doesn't work, and incorporate the successful approaches into the "official" method.

Components of Goal-Directed Design

The Goal-Directed method is a set of tools and best practices developed entirely through practice in the real world. The method is not intended to be a set of rules and constraints; rather, it provides a framework within which skilled designers can do what they do best—generate great solutions—with the confidence that the method will help them get it right. No method can eliminate the need for the knowledge and skill of the designer. At Cooper, we hire skilled, experienced designers and put them through classes and an apprenticeship. On average, it takes them about a year to master the fundamental techniques and two or more years before they can take full advantage of the method's potential.

The method consists of four components: principles, patterns, process, and practices. Successful implementation of the method also requires people with the right skills (see the discussion of team roles in Chapter 2). This book focuses on process and practices, but it's worth briefly discussing principles and patterns to illustrate how these four components fit together.

PRINCIPLES

Principles are guidelines for creating good solutions under specific circumstances. For example, it is generally better for a computer to act immediately on a user's command but provide an option to undo it than it is to require confirmation of every action, but there are exceptions, such as when that action might do truly irreversible damage.

This book does not focus on principles, although Chapters 15, 17, and 21 do include some of the principles that are most useful for certain aspects of design. However, there are plenty of good references available. The canon at Cooper naturally includes the principles articulated in *About Face* 3.[2] I also recommend *Designing Visual Interfaces*.[3] Regardless of where you look for design principles, however, there are two essential things to keep in mind: not all principles apply in all contexts, and not all principles are created equal.

Principles appropriate to one context may not apply to another. A widget that works beautifully with a mouse may get in the way if a user's task is primarily free-form keyboard entry. Approaches to visual design on a monitor differ from those on a television screen or handheld device. However, most desktop interface design principles are equally applicable to Web sites.

When I say that all principles are not created equal, I mean that some supposed principles—including some of those expounded by various design or usability gurus—are simply unfounded opinion. For example, a number of years ago there was a much-touted rule about how many seconds it should take your Web page to load. In 2001, Christine Perfetti and Lori Landesman disproved this assertion when their study[4] showed that perception of page load times has little to do with objective reality and much more to do with whether someone can accomplish her goals on the site.

When evaluating whether a supposed principle is both true and applicable to the problem in front of you, ask yourself whether it passes the following tests:

1. **Does it help your users accomplish their goals?** Not every valid principle is about goals, but any principle that's both true and applicable won't work against them.
2. **Will it help users minimize work?** Work can be cognitive (thinking about whether to press yes, no, or cancel), visual (reading light gray text on a white background), memory (remembering all those complicated passwords), or motor (using an iPod click wheel to traverse from A to Z in a huge music collection). Note that there are rare cases, such as in video games, when introducing a bit of work is good as long as it's done in an engaging way.

2. Cooper, A., Reimann, R., and Cronin, D. *About face 3: The essentials of interaction design*. John Wiley and Sons, 2007.
3. Mullet, K., and Sano, D. *Designing visual interfaces: Communication oriented techniques*. Prentice Hall, 1994.
4. Perfetti, C., and Landesman, L. "The Truth About Download Time," January 31, 2001. www.uie.com/articles/download_time/

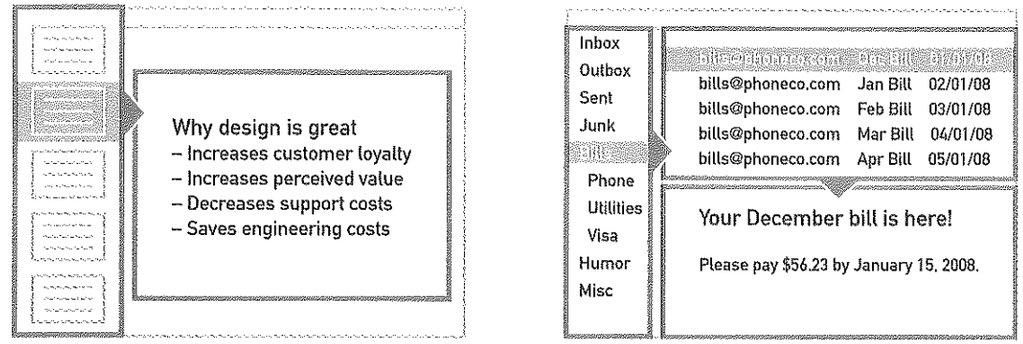

Figure 1.2. Many productivity applications are based on an organizer/workspace pattern.

PATTERNS

Patterns are types of solutions that tend to be useful for certain classes of problems. For example, Figure 1.2 illustrates a common pattern for navigating multi-document interfaces such as e-mail. A pane on the top or left of the screen contains a list of documents or groups of documents, while one or two panes below or to the right allow drilling down into an individual document's contents. This works well for electronic medical records, browser bookmarks, and a host of other design problems that involve looking at each of many documents for a short period of time.

Patterns are essential in design because they are the building blocks of a designer's vocabulary, as principles are the rules of grammar that govern how we use them. Seasoned designers can often work faster and come up with a wider range of ideas because they've had years in which to build up their vocabularies.

I cannot do patterns justice in this book, though Chapters 15, 17, and 21 provide a sampling of some of the most useful. There are a few other references you might find worthwhile. Many designers first became interested in patterns after reading architect Christopher Alexander's seminal book, *A Pattern Language*.[5] For a more specific approach, look at *Universal Principles of Design*.[6] This book muddles principles and patterns together, but is useful in that it straddles multiple design disciplines. Finally, Jenifer Tidwell's *Designing Interfaces*[7] offers a nice collection of interaction and visual interface design patterns.

PROCESS

This book focuses primarily on the **design process**: the steps and techniques involved in planning and conducting design research, using it to develop personas, scenarios, and requirements, then using those to develop and iterate a design solution. This process (outlined in Figure 1,3) can scale up or down depending on the time and budget available as well as the priorities.

5. Alexander, C. *A pattern language: Towns, buildings, construction*. Oxford University Press, 1977.
6. Lidwell, W., Holden, K., and Butler, J. *Universal principles of design: 100 ways to enhance usability, influence perception, increase appeal, make better design decisions, and teach through design*. Rockport Publishers, 2003.
7. Tidwell, J. *Designing interfaces: Patterns for effective interaction design*. O'Reilly, 2005.

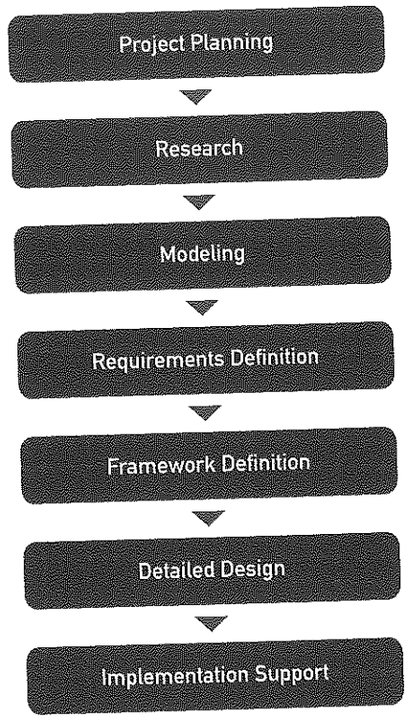

Figure 1.3. An overview of the Goal-Directed process.

Some organizations (such as start-ups in search of additional funding) may believe getting a best-guess product out the door is more important than taking the time to understand their users and customers in any depth, so there are ways to jump to design with little or no time spent on research. Throughout the book, I'll describe each part of the process in its typical form first, then discuss how it can be compressed or expanded.

Project planning

Before dedicating resources to a project, most executives want at least a rough outline of how the project will be structured and what results they can expect. It is essential to identify the key stakeholders, determine from their input what the objectives of the project are, and draft a plan from there. It is almost never possible to predict exactly how long a completed design will take, since the problem is not yet sufficiently defined and there are likely several possible approaches to solving it. However, it should be possible for an experienced designer to propose an ideal approach and rough schedule, then discuss with stakeholders the trade-offs they could make to reduce the time or cost involved.

Planning is not complete once an initial project plan is done, however. Recruiting research participants and scheduling the interviews is far from trivial. Chapter 6 outlines approaches to this sometimes-daunting task.

Research

To solve a problem, you must first understand it. Good research helps you make the best product definition and design decisions later on. Beyond helping you understand the design problem, research also helps build consensus and move the design process along faster. Once you have objective data, it's no longer necessary to pit one person's opinion against another's.

Interviews with stakeholders, which are the first component of research, provide a clear view of the business objectives and technical parameters, while uncovering risks and assumptions you should examine. This topic is covered in Chapter 5. The next step, ethnographic research with potential users, gives you insight into goals, environments, communication needs, and other important factors. Chapters 7 and 8 tell you how to go about it. Finally, Chapter 9 describes additional methods that can be useful.

Modeling

Although raw research data can be eye-opening, analysis makes that data more useful both to the people who conducted the research and to other members of the product team. This analysis

involves identifying trends and developing models that explain what you observed. The most important model is a set of **personas**, which are user archetypes that help you make design decisions and communicate your rationale. Each persona represents a set of behavior patterns and goals. By designing for these archetypal users, you can satisfy the needs of the broader range of people they represent. Every product decision can be tied back to the personas. Other models may include representations of current workflows, the usage environment, or other important aspects of the problem. Chapters 10 and 11 describe how to develop personas and other models.

Requirements definition

The last step in analyzing the data is determining what it implies about the product's functionality and design. The personas' skills, environments, behaviors, and goals help determine their needs. Scenarios, which are stories about the personas using the future product or service, highlight additional needs. In the language of product development, these needs are expressed as requirements. Of course, business objectives and constraints are also incorporated in this list.

Design requirements do not represent a comprehensive list that the engineering team can use to build the product; this would be like handing a building contractor a list that says your house should contain some sort of kitchen and three bedrooms. Rather, these requirements are intended to give business stakeholders a chance to make critical trade-off decisions early in the process. A meeting with the full set of stakeholders at the end of this phase lets you discuss the personas and requirements. The end result should be consensus about the focus and parameters of your design efforts.

Chapters 12 and 13 explain how to develop and communicate requirements.

Framework definition

Once you have agreement on who the users and customers are and what the design must accomplish for them, you can begin laying out the basic framework for the form and behavior of the product. Personas and scenarios are the primary drivers of this process, but a vocabulary of design patterns and a solid grounding in principles are essential, as well. The phase begins with broad exploration of multiple solutions, though whether this exploration takes a few hours or several weeks depends on the project objectives.

As the options are narrowed, the **interaction framework** outlines how functionality is grouped and how the personas will accomplish the most critical tasks. The **visual framework** expresses the brand's qualities in concrete terms, typically using design language studies divorced from the interaction design. The **industrial design framework** consists of an approximate form factor and component architecture, physical expression of the brand developed in conjunction with the visual design language, and a description of any hardware controls that are essential to the interaction. Chapters 14 through 18 describe how to go about framework definition, while Chapter 19 outlines effective ways to communicate about each framework.

In all three cases, the design is articulated at a high level, deliberately omitting details for two reasons. First, stakeholders need something concrete to look at as quickly as possible, which means there simply isn't time to address the finer points. Second, starting design by thinking about the major underlying structures helps ensure that the structure is clear and simple, just as an outline aids clear writing. It also prevents rework later on.

Discussion of the framework with the complete set of stakeholders is another important step in refining the product's focus and parameters. Once people begin to see a solution, they can better assess how critical it is to develop certain aspects

now versus later. If the design can be somewhat unconstrained at this stage, stakeholders are more able to see the possibilities and make more informed trade-off decisions.

Detailed design

Once the scope of the product is clearly defined, it is usually possible to develop a detailed project plan for filling out and refining the design. Increasingly detailed scenarios continue to drive the interaction design aspect of the process. However, the role of design principles, design evaluation, and engineering feasibility increases as the level of detail in the design increases. Chapters 20 through 23 describe this process.

This phase ideally involves extensive collaboration with subject matter experts and engineers. Subject matter experts can provide a greater level of detail about best practices and edge cases than you can glean in field research. Engineers can ask questions that will help them build the design and point out aspects of it that may be difficult to implement (though this does not necessarily mean you should use a poor design alternative just because it's easier to build). When the engineers are not certain what it will take to build a specific aspect of the design, they can build some throwaway code or hardware, then feed what they learn back to the design team.

It's best if you can determine how the product looks and works down to the contents of every list box and the colors of the pixels in every icon, then document the results in a detailed specification (see Chapter 24). Arriving at a final specification that is truly ready for construction typically requires two passes through the detailed design; the design team provides as much detail as they can in a first draft, then the engineers and subject matter experts review the design in painstaking detail, looking for areas where they have questions or concerns, and the design team revises things accordingly. The first draft specification also provides an excellent opportunity for usability testing, though of course you may opt to test earlier if you have concerns about specific issues. To learn more about how testing fits into the design process, see Chapter 23.

Implementation support

Any building architect will tell you that her work is not done until construction is complete. The same can be said for designers of digital products. No matter how good the engineering team or how thorough the specification, unexpected issues or questions inevitably crop up. If the engineers must begin making design decisions in your absence, things may head downhill from there, with some engineers varying from the specification for the sake of easier implementation. Engineers are less likely to take matters into their own hands if they have a good relationship with the design team and know you're available for a day or so each week. Chapter 25 addresses this kind of ongoing support.

PRACTICES

The design process does not stand alone; its effectiveness depends in large part upon the project management practices that support it. The structure of the team, the timing and content of communication, and the way collaboration works within the design team and outside of it all affect the outcome of a project. The Goal-Directed method is optimized to be as efficient as possible without sacrificing effectiveness.

Among other things, this means that the optimal design team is small but has frequent contact with members of a larger product team. Most projects require no more than five people on the design team, two or three of whom are part-time. Some projects require fewer. Of course, such a small team requires rigorous hiring and training practices to ensure that each team member has the required skills. See Chapter 2 for more on the various roles involved and the skills required of each.

Design serves as a process catalyst by making ideas concrete. A handful of formal meetings move the process along either by forcing decisions to happen or by helping people identify what additional information they need. The degree of formality and thoroughness in these discussions varies according to the size and geographical distribution of the product team, the culture of the company, and the need for materials to reference later on. Chapters 13, 19, and 24 describe formal communication at the most critical points in the process.

Not all organizations are able to take full advantage of design's strategic value. Even the best design process carried out by the best designers may not succeed in an environment where the business people are afraid of the engineers, the engineers are not very skilled, or decision-making is dysfunctional in some way, such as when no one has clear responsibility or the commensurate authority to make something happen. Assuming a reasonably healthy organization to start, though, it is entirely possible to introduce design and eventually make it a central part of how products are conceived and developed. Such transformations won't happen overnight, however; they take at least several years, provided you have a coherent plan and executive support. They also rely on designers having fully developed their own skills. Chapter 26 provides some ideas for developing individual and organizational design capabilities.

Summary

An effective design method supports designers in doing what they do best: visualizing concrete solutions to human problems. Goal-Directed Design helps skilled designers ensure thoroughness, timely execution, and consistently high quality of output. It also helps ensure that the design effort is not in vain by making the thought process transparent to the rest of the product team.

The remainder of this book focuses on the process and practices that have helped Cooper teams and other designers we've trained deliver great work on a deadline, from planning the research effort to seeing the product out the door. Of course, these processes and practices are only as good as the people applying them, so Chapter 2 outlines the skills and roles you'll want on your team.