- Put the knowledge required to operate the technology in the world. Don't require that all the knowledge must be in the head. Allow for efficient operation when people have learned all the requirements, when they are experts who can perform without the knowledge in the world, but make it possible for non-experts to use the knowledge in the world. This will also help experts who need to perform a rare, infrequently performed operation or return to the technology after a prolonged absence.
- Use the power of natural and artificial constraints: physical, logical, semantic, and cultural. Exploit the power of forcing functions and natural mappings.
- Bridge the two gulfs, the Gulf of Execution and the Gulf of Evaluation. Make things visible, both for execution and evaluation. On the execution side, provide feedforward information: make the options readily available. On the evaluation side, provide feedback: make the results of each action apparent. Make it possible to determine the system's status readily, easily, accurately, and in a form consistent with the person's goals, plans, and expectations.

We should deal with error by embracing it, by seeking to understand the causes and ensuring they do not happen again. We need to assist rather than punish or scold.

CHAPTER SIX

DESIGN THINKING

One of my rules in consulting is simple: never solve the problem I am asked to solve. Why such a counterintuitive rule? Because, invariably, the problem I am asked to solve is not the real, fundamental, root problem. It is usually a symptom. Just as in Chapter 5, where the solution to accidents and errors was to determine the real, underlying cause of the events, in design, the secret to success is to understand what the real problem is.

It is amazing how often people solve the problem before them without bothering to question it. In my classes of graduate students in both engineering and business, I like to give them a problem to solve on the first day of class and then listen the next week to their wonderful solutions. They have masterful analyses, drawings, and illustrations. The MBA students show spreadsheets in which they have analyzed the demographics of the potential customer base. They show lots of numbers: costs, sales, margins, and profits. The engineers show detailed drawings and specifications. It is all well done, brilliantly presented.

When all the presentations are over, I congratulate them, but ask: "How do you know you solved the correct problem?" They are puzzled. Engineers and business people are trained to solve problems. Why would anyone ever give them the wrong problem? "Where do you think the problems come from?" I ask. The real world is not like the university. In the university, professors make up artificial problems. In the real world, the problems do not come in nice, neat packages. They have to be discovered. It is all too easy to see only the surface problems and never dig deeper to address the real issues.

Solving the Correct Problem

Engineers and businesspeople are trained to solve problems. Designers are trained to discover the real problems. A brilliant solution to the wrong problem can be worse than no solution at all: solve the correct problem.

Good designers never start by trying to solve the problem given to them: they start by trying to understand what the real issues are. As a result, rather than converge upon a solution, they diverge, studying people and what they are trying to accomplish, generating idea after idea after idea. It drives managers crazy. Managers want to see progress: designers seem to be going backward when they are given a precise problem and instead of getting to work, they ignore it and generate new issues to consider, new directions to explore. And not just one, but many. What is going on?

The key emphasis of this book is the importance of developing products that fit the needs and capabilities of people. Design can be driven by many different concerns. Sometimes it is driven by technology, sometimes by competitive pressures or by aesthetics. Some designs explore the limits of technological possibilities; some explore the range of imagination, of society, of art or fashion. Engineering design tends to emphasize reliability, cost, and efficiency. The focus of this book, and of the discipline called human-centered design, is to ensure that the result fits human desires, needs, and capabilities. After all, why do we make products? We make them for people to use.

Designers have developed a number of techniques to avoid being captured by too facile a solution. They take the original problem as a suggestion, not as a final statement, then think broadly about what the issues underlying this problem statement might really be (as was done through the "Five Whys" approach to getting at the root cause, described in Chapter 5). Most important of all is that the process be iterative and expansive. Designers resist the temptation to jump immediately to a solution for the stated problem. Instead, they first spend time determining what basic, fundamental (root) issue needs to be addressed. They don't try to search for a solution until they have determined the real problem, and even then, instead of solving that problem, they stop to consider a wide range of potential solutions. Only then will they finally converge upon their proposal. This process is called *design thinking.*

Design thinking is not an exclusive property of designers—all great innovators have practiced this, even if unknowingly, regardless of whether they were artists or poets, writers or scientists, engineers or businesspeople. But because designers pride themselves on their ability to innovate, to find creative solutions to fundamental problems, design thinking has become the hallmark of the modern design firm. Two of the powerful tools of design thinking are human-centered design and the double-diamond diverge-converge model of design.

Human-centered design (HCD) is the process of ensuring that people's needs are met, that the resulting product is understandable and usable, that it accomplishes the desired tasks, and that the experience of use is positive and enjoyable. Effective design needs to satisfy a large number of constraints and concerns, including shape and form, cost and efficiency, reliability and effectiveness, understandability and usability, the pleasure of the appearance, the pride of ownership, and the joy of actual use. HCD is a procedure for addressing these requirements, but with an emphasis on two things: solving the right problem, and doing so in a way that meets human needs and capabilities.

Over time, the many different people and industries that have been involved in design have settled upon a common set of methods for doing HCD. Everyone has his or her own favorite method, but all are variants on the common theme: iterate through the four stages of observation, generation, prototyping, and testing. But even before this, there is one overriding principle: solve the right problem.

These two components of design—finding the right problem and meeting human needs and capabilities—give rise to two phases of the design process. The first phase is to find the right problem, the second is to find the right solution. Both phases use the HCD process. This double-phase approach to design led the British Design Council to describe it as a "double diamond." So that is where we start the story.

The Double-Diamond Model of Design

Designers often start by questioning the problem given to them: they expand the scope of the problem, diverging to examine all the fundamental issues that underlie it. Then they converge upon a single problem statement. During the solution phase of their studies, they first expand the space of possible solutions, the divergence phase. Finally, they converge upon a proposed solution (Figure 6.1). This double diverge-converge pattern was first introduced in 2005 by the British Design Council, which called it the *double-diamond design process model.* The Design Council divided the design process into four stages: "discover" and "define"—for the divergence and convergence phases of finding the right problem,

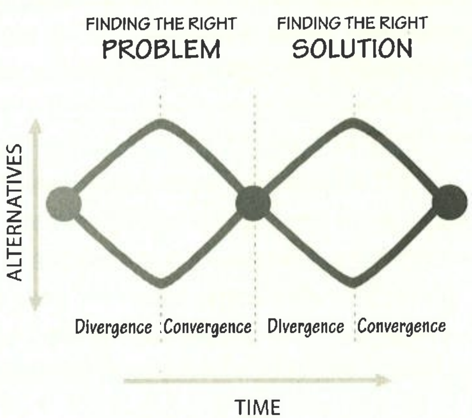

FIGURE 6.1. The Double-Diamond Model of Design. Start with an idea, and through the initial design research, expand the thinking to explore the fundamental issues. Only then is it time to converge upon the real, underlying problem. Similarly, use design research tools to explore a wide variety of solutions before converging upon one. (Slightly modified from the work of the British Design Council, 2005.)

and "develop" and "deliver"—for the divergence and convergence phases of finding the right solution.

The double diverge-converge process is quite effective at freeing designers from unnecessary restrictions to the problem and solution spaces. But you can sympathize with a product manager who, having given the designers a problem to solve, finds them questioning the assignment and insisting on traveling all over the world to seek deeper understanding. Even when the designers start focusing upon the problem, they do not seem to make progress, but instead develop a wide variety of ideas and thoughts, many only half-formed, many clearly impractical. All this can be rather unsettling to the product manager who, concerned about meeting the schedule, wants to see immediate convergence. To add to the frustration of the product manager, as the designers start to converge upon a solution, they may realize that they have inappropriately formulated the problem, so the entire process must be repeated (although it can go more quickly this time).

This repeated divergence and convergence is important in properly determining the right problem to be solved and then the best way to solve it. It looks chaotic and ill-structured, but it actually follows well-established principles and procedures. How does the product manager keep the entire team on schedule despite the apparent random and divergent methods of designers? Encourage their free exploration, but hold them to the schedule (and budget) constraints. There is nothing like a firm deadline to get creative minds to reach convergence.

The Human-Centered Design Process

The double-diamond describes the two phases of design: finding the right problem and fulfilling human needs. But how are these actually done? This is where the human-centered design process comes into play: it takes place within the double-diamond diverge-converge process.

There are four different activities in the human-centered design process (Figure 6.2):

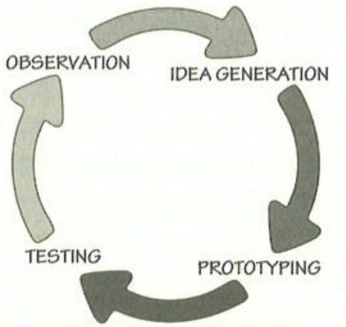

FIGURE 6.2. **The Iterative Cycle of Human-Centered Design.** Make observations on the intended target population, generate ideas, produce prototypes and test them. Repeat until satisfied. This is often called the *spiral method* (rather than the circle depicted here), to emphasize that each iteration through the stages makes progress.

1. Observation
2. Idea generation (ideation)
3. Prototyping
4. Testing

These four activities are iterated; that is, they are repeated over and over, with each cycle yielding more insights and getting closer to the desired solution. Now let us examine each activity separately.

OBSERVATION

The initial research to understand the nature of the problem itself is part of the discipline of design research. Note that this is research about the customer and the people who will use the products under consideration. It is not the kind of research that scientists do in their laboratories, trying to find new laws of nature. The design researcher will go to the potential customers, observing their activities, attempting to understand their interests, motives, and true needs. The problem definition for the product design will come from this deep understanding of the goals the people are trying to accomplish and the impediments they experience. One of its most critical techniques is to observe the would-be customers in their natural environment, in their normal lives, wherever the product or service being designed will actually be used. Watch them in their homes, schools, and offices. Watch them commute, at parties, at mealtime, and with friends at the local bar. Follow them into the shower if necessary, because it is essential to understand the real situations that they encounter, not some pure isolated experience. This technique is called *applied ethnography*, a method adapted from the field of anthropology. Applied ethnography differs from the slower, more methodical, research-oriented practice of academic anthropologists because the goals are different.

For one, design researchers have the goal of determining human needs that can be addressed through new products. For another, product cycles are driven by schedule and budget, both of which require more rapid assessment than is typical in academic studies that might go on for years.

It's important that the people being observed match those of the intended audience. Note that traditional measures of people, such as age, education, and income, are not always important: what matters most are the activities to be performed. Even when we look at widely different cultures, the activities are often surprisingly similar. As a result, the studies can focus upon the activities and how they get done, while being sensitive to how the local environment and culture might modify those activities. In some cases, such as the products widely used in business, the activity dominates. Thus, automobiles, computers, and phones are pretty standardized across the world because their designs reflect the activities being supported.

In some cases, detailed analyses of the intended group are necessary. Japanese teenage girls are quite different from Japanese women, and in turn, very different from German teenage girls. If a product is intended for subcultures like these, the exact population must be studied. Another way of putting it is that different products serve different needs. Some products are also symbols of status or group membership. Here, although they perform useful functions, they are also fashion statements. This is where teenagers in one culture differ from those of another, and even from younger children and older adults of the same culture. Design researchers must carefully adjust the focus of their observations to the intended market and people for whom the product is intended.

Will the product be used in some country other than where it is being designed? There is only one way to find out: go there (and always include natives in the team). Don't take a shortcut and stay home, talking to students or visitors from that country while remaining in your own: what you will learn is seldom an accurate reflection of the target population or of the ways in which the proposed product will actually be used. There is no substitute for direct observation of and interaction with the people who will be using the product.

Design research supports both diamonds of the design process. The first diamond, finding the right problem, requires a deep understanding of the true needs of people. Once the problem has been defined, finding an appropriate solution again requires deep understanding of the intended population, how those people perform their activities, their capabilities and prior experience, and what cultural issues might be impacted.

DESIGN RESEARCH VERSUS MARKET RESEARCH

Design and marketing are two important parts of the product development group. The two fields are complementary, but each has a different focus. Design wants to know what people really need and how they actually will use the product or service under consideration. Marketing wants to know what people will buy, which includes learning how they make their purchasing decisions. These different aims lead the two groups to develop different methods of inquiry. Designers tend to use qualitative observational methods by which they can study people in depth, understanding how they do their activities and the environmental factors that come into play. These methods are very time consuming, so designers typically only examine small numbers of people, often numbering in the tens.

Marketing is concerned with customers. Who might possibly purchase the item? What factors might entice them to consider and purchase a product? Marketing traditionally uses large-scale, quantitative studies, with heavy reliance on focus groups, surveys, and questionnaires. In marketing, it is not uncommon to converse with hundreds of people in focus groups, and to question tens of thousands of people by means of questionnaires and surveys.

The advent of the Internet and the ability to assess huge amounts of data have given rise to new methods of formal, quantitative market analysis. "Big data," it is called, or sometimes "market analytics." For popular websites, A/B testing is possible in which two potential variants of an offering are tested by giving

some randomly selected fraction of visitors (perhaps 10 percent) one set of web pages (the A set); and another randomly selected set of visitors, the other alternative (the B set). In a few hours, hundreds of thousands of visitors may have been exposed to each test set, making it easy to see which yields better results. Moreover, the website can capture a wealth of information about people and their behavior: age, income, home and work addresses, previous purchases, and other websites visited. The virtues of the use of big data for market research are frequently touted. The deficiencies are seldom noted, except for concerns about invasions of personal privacy. In addition to privacy issues, the real problem is that numerical correlations say nothing of people's real needs, of their desires, and of the reasons for their activities. As a result, these numerical data can give a false impression of people. But the use of big data and market analytics is seductive: no travel, little expense, and huge numbers, sexy charts, and impressive statistics, all very persuasive to the executive team trying to decide which new products to develop. After all, what would you trust—neatly presented, colorful charts, statistics, and significance levels based on millions of observations, or the subjective impressions of a motley crew of design researchers who worked, slept, and ate in remote villages, with minimal sanitary facilities and poor infrastructure?

The different methods have different goals and produce very different results. Designers complain that the methods used by marketing don't get at real behavior: what people say they do and want does not correspond with their actual behavior or desires. People in marketing complain that although design research methods yield deep insights, the small number of people observed is a concern. Designers counter with the observation that traditional marketing methods provide shallow insight into a large number of people.

The debate is not useful. All groups are necessary. Customer research is a tradeoff: deep insights on real needs from a tiny set of people, versus broad, reliable purchasing data from a wide range and large number of people. We need both. Designers understand what people really need. Marketing understands what people actually buy. These are not the same things, which is why both approaches are required: marketing and design researchers should work together in complementary teams.

What are the requirements for a successful product? First, if nobody buys the product, then all else is irrelevant. The product design has to provide support for all the factors people use in making purchase decisions. Second, once the product has been purchased and is put into use, it must support real needs so that people can use, understand, and take pleasure from it. The design specifications must include both factors: marketing and design, buying and using.

IDEA GENERATION

Once the design requirements are determined, the next step for a design team is to generate potential solutions. This process is called *idea generation*, or *ideation*. This exercise might be done for both of the double diamonds: during the phase of finding the correct problem, then during the problem solution phase.

This is the fun part of design: it is where creativity is critical. There are many ways of generating ideas: many of these methods fall under the heading of "brainstorming." Whatever the method used, two major rules are usually followed:

- **Generate numerous ideas.** It is dangerous to become fixated upon one or two ideas too early in the process.
- **Be creative without regard for constraints.** Avoid criticizing ideas, whether your own or those of others. Even crazy ideas, often obviously wrong, can contain creative insights that can later be extracted and put to good use in the final idea selection. Avoid premature dismissal of ideas.

I like to add a third rule:

- **Question everything.** I am particularly fond of "stupid" questions. A stupid question asks about things so fundamental that everyone assumes the answer is obvious. But when the question is taken seriously, it often turns out to be profound: the obvious often is not obvious at all. What we assume to be obvious is simply the way things have always been done, but now that it is questioned, we don't actually know the reasons. Quite often the solution to problems is discovered through stupid questions, through questioning the obvious.

PROTOTYPING

The only way to really know whether an idea is reasonable is to test it. Build a quick prototype or mock-up of each potential solution. In the early stages of this process, the mock-ups can be pencil sketches, foam and cardboard models, or simple images made with simple drawing tools. I have made mock-ups with spreadsheets, PowerPoint slides, and with sketches on index cards or sticky notes. Sometimes ideas are best conveyed by skits, especially if you're developing services or automated systems that are difficult to prototype.

One popular prototype technique is called "Wizard of Oz," after the wizard in L. Frank Baum's classic book (and the classic movie) *The Wonderful Wizard of Oz*. The wizard was actually just an ordinary person but, through the use of smoke and mirrors, he managed to appear mysterious and omnipotent. In other words, it was all a fake: the wizard had no special powers.

The Wizard of Oz method can be used to mimic a huge, powerful system long before it can be built. It can be remarkably effective in the early stages of product development. I once used this method to test a system for making airline reservations that had been designed by a research group at the Xerox Corporation's Palo Alto Research Center (today it is simply the Palo Alto Research Center, or PARC). We brought people into my laboratory in San Diego one at a time, seated them in a small, isolated room, and had them type their travel requirements into a computer. They thought they were interacting with an automated travel assistance program, but in fact, one of my graduate students was sitting in an adjacent room, reading the typed queries and typing back responses (looking up real travel schedules where appropriate). This simulation taught us a lot about the requirements for such a system. We learned, for example, that people's sentences were very different from the ones we had designed the system to handle. Example: One of the people we tested requested a round-trip ticket between San Diego and San Francisco. After the system had determined the desired flight to San Francisco, it asked, "When would you like to return?" The person responded, "I would like to leave on the following Tuesday, but I have to be back before my first class at 9 AM." We soon learned that it wasn't sufficient to understand the sentences: we also had to do problem-solving, using considerable knowledge about such things as airport and meeting locations, traffic patterns, delays for getting baggage and rental cars, and of course, parking—more than our system was capable of doing. Our initial goal was to understand language. The studies demonstrated that the goal was too limited: we needed to understand human activities.

Prototyping during the problem specification phase is done mainly to ensure that the problem is well understood. If the target population is already using something related to the new product, that can be considered a prototype. During the problem solution phase of design, then real prototypes of the proposed solution are invoked.

TESTING

Gather a small group of people who correspond as closely as possible to the target population—those for whom the product is intended. Have them use the prototypes as nearly as possible to the way they would actually use them. If the device is normally used by one person, test one person at a time. If it is normally used by a group, test a group. The only exception is that even if the normal usage is by a single person, it is useful to ask a pair of people to use it together, one person operating the prototype, the other guiding the actions and interpreting the results (aloud). Using pairs in this way causes them to discuss their ideas, hypotheses, and frustrations openly and naturally. The research team should be observing, either by sitting behind those being tested (so as not to distract them) or by watching through video in another room (but having the video camera visible and after describing the procedure). Video recordings of the tests are often quite valuable, both for later showings to team members who could not be present and for review.

When the study is over, get more detailed information about the people's thought processes by retracing their steps, reminding them of their actions, and questioning them. Sometimes it helps to show them video recordings of their activities as reminders.

How many people should be studied? Opinions vary, but my associate, Jakob Nielsen, has long championed the number five: five people studied individually. Then, study the results, refine them, and do another iteration, testing five different people. Five is usually enough to give major findings. And if you really want to test many more people, it is far more effective to do one test of five, use the results to improve the system, and then keep iterating the test-design cycle until you have tested the desired number of people. This gives multiple iterations of improvement, rather than just one.

Like prototyping, testing is done in the problem specification phase to ensure that the problem is well understood, then done again in the problem solution phase to ensure that the new design meets the needs and abilities of those who will use it.

ITERATION

The role of iteration in human-centered design is to enable continual refinement and enhancement. The goal is rapid prototyping and testing, or in the words of David Kelley, Stanford professor and cofounder of the design firm IDEO, "Fail frequently, fail fast."

Many rational executives (and government officials) never quite understand this aspect of the design process. Why would you want to fail? They seem to think that all that is necessary is to determine the requirements, then build to those requirements. Tests, they believe, are only necessary to ensure that the requirements are met. It is this philosophy that leads to so many unusable systems. Deliberate tests and modifications make things better. Failures are to be encouraged—actually, they shouldn't be called failures: they should be thought of as learning experiences. If everything works perfectly, little is learned. Learning occurs when there are difficulties.

The hardest part of design is getting the requirements right, which means ensuring that the right problem is being solved, as well as that the solution is appropriate. Requirements made in the abstract are invariably wrong. Requirements produced by asking people what they need are invariably wrong. Requirements are developed by watching people in their natural environment.

When people are asked what they need, they primarily think of the everyday problems they face, seldom noticing larger failures, larger needs. They don't question the major methods they use. Moreover, even if they carefully explain how they do their tasks and then agree that you got it right when you present it back to them, when you watch them, they will often deviate from their own description. "Why?" you ask. "Oh, I had to do this one differently," they might reply; "this was a special case." It turns out that most cases are "special." Any system that does not allow for special cases will fail.

Getting the requirements right involves repeated study and testing: iteration. Observe and study: decide what the problem might be, and use the results of tests to determine which parts of the design work, which don't. Then iterate through all four processes once again. Collect more design research if necessary, create more ideas, develop the prototypes, and test them.

With each cycle, the tests and observations can be more targeted and more efficient. With each cycle of the iteration, the ideas become clearer, the specifications better defined, and the prototypes closer approximations to the target, the actual product. After the first few iterations, it is time to start converging upon a solution. The several different prototype ideas can be collapsed into one.

When does the process end? That is up to the product manager, who needs to deliver the highest-possible quality while meeting the schedule. In product development, schedule and cost provide very strong constraints, so it is up to the design team to meet these requirements while getting to an acceptable, high-quality design. No matter how much time the design team has been allocated, the final results only seem to appear in the last twenty-four hours before the deadline. (It's like writing: no matter how much time you are given, it's finished only hours before the deadline.)

ACTIVITY-CENTERED VERSUS HUMAN-CENTERED DESIGN

The intense focus on individuals is one of the hallmarks of human-centered design, ensuring that products do fit real needs, that they are usable and understandable. But what if the product is intended for people all across the world? Many manufacturers make essentially the same product for everyone. Although automobiles are slightly modified for the requirements of a country, they are all basically the same the world round. The same is true for cameras, computers, telephones, tablets, television sets, and refrigerators. Yes, there are some regional differences, but remarkably little. Even products specifically designed for one culture—rice cookers, for example—get adopted by other cultures elsewhere.

How can we pretend to accommodate all of these very different, very disparate people? The answer is to focus on activities, not the individual person. I call this *activity-centered design*. Let the activity define the product and its structure. Let the conceptual model of the product be built around the conceptual model of the activity.

Why does this work? Because people's activities across the world tend to be similar. Moreover, although people are unwilling to learn systems that appear to have arbitrary, incomprehensible requirements, they are quite willing to learn things that appear to be essential to the activity. Does this violate the principles of human-centered design? Not at all: consider it an enhancement of HCD. After all, the activities are done by and for people. Activity-centered approaches are human-centered approaches, far better suited for large, nonhomogeneous populations.

Take another look at the automobile, basically identical all across the world. It requires numerous actions, many of which make little sense outside of the activity and that add to the complexity of driving and to the rather long period it takes to become an accomplished, skilled driver. There is the need to master foot pedals, to steer, use turn signals, control the lights, and watch the road, all while being aware of events on either side of and behind the vehicle, and perhaps while maintaining conversations with the other people in the auto. In addition, instruments on the panel need to be watched, especially the speed indicator, as well as the water temperature, oil pressure, and fuel level. The locations of the rear- and side-view mirrors require the eyes to be off the road ahead for considerable time.

People learn to drive cars quite successfully despite the need to master so many subcomponent tasks. Given the design of the car and the activity of driving, each task seems appropriate. Yes, we can make things better. Automatic transmissions eliminate the need for the third pedal, the clutch. Heads-up displays mean that critical instrument panel and navigation information can be displayed in the space in front of the driver, so no eye movements are required to monitor them (although it requires an attentional shift, which does take attention off the road). Someday we will replace the three different mirrors with one video display that shows objects on all sides of the car in one image, simplifying yet another action. How do we make things better? By careful study of the activities that go on during driving.

Support the activities while being sensitive to human capabilities, and people will accept the design and learn whatever is necessary.

ON THE DIFFERENCES BETWEEN TASKS AND ACTIVITIES

One comment: there is a difference between task and activity. I emphasize the need to design for activities: designing for tasks is usually too restrictive. An activity is a high-level structure, perhaps "go shopping." A task is a lower-level component of an activity, such as "drive to the market," "find a shopping basket," "use a shopping list to guide the purchases," and so forth.

An activity is a collected set of tasks, but all performed together toward a common high-level goal. A task is an organized, cohesive set of operations directed toward a single, low-level goal. Products have to provide support for both activities and the various tasks that are involved. Well-designed devices will package together the various tasks that are required to support an activity, making them work seamlessly with one another, making sure the work done for one does not interfere with the requirements for another.

Activities are hierarchical, so a high-level activity (going to work) will have under it numerous lower-level ones. In turn, low-level activities spawn "tasks," and tasks are eventually executed by basic "operations." The American psychologists Charles Carver and Michael Scheier suggest that goals have three fundamental levels that control activities. Be-goals are at the highest, most abstract level and govern a person's being: they determine why people act, are fundamental and long lasting, and determine one's self-image. Of far more practical concern for everyday activity is the next level down, the do-goal, which is more akin to the goal I discuss in the seven stages of activity. Do-goals determine the plans and actions to be performed for an activity. The lowest level of this hierarchy is the motor-goal, which specifies just how the actions are performed: this is more at the level of tasks and operations rather than activities. The German psychologist Marc Hassenzahl has shown how this three-level analysis can be used to guide in the development and analysis of a person's experience (the user experience, usually abbreviated UX) in interacting with products.

Focusing upon tasks is too limiting. Apple's success with its music player, the iPod, was because Apple supported the entire activity involved in listening to music: discovering it, purchasing it, getting it into the music player, developing playlists (that could be shared), and listening to the music. Apple also allowed other companies to add to the capabilities of the system with external speakers, microphones, all sorts of accessories. Apple made it possible to send the music throughout the home, to be listened to on those other companies' sound systems. Apple's success was due to its combination of two factors: brilliant design plus support for the entire activity of music enjoyment.

Design for individuals and the results may be wonderful for the particular people they were designed for, but a mismatch for others. Design for activities and the result will be usable by everyone. A major benefit is that if the design requirements are consistent with their activities, people will tolerate complexity and the requirements to learn something new: as long as the complexity and the new things to be learned feel appropriate to the task, they will feel natural and be viewed as reasonable.

ITERATIVE DESIGN VERSUS LINEAR STAGES

The traditional design process is linear, sometimes called the *waterfall method* because progress goes in a single direction, and once decisions have been made, it is difficult or impossible to go back. This is in contrast to the iterative method of human-centered design, where the process is circular, with continual refinement, continual change, and encouragement of backtracking, rethinking early decisions. Many software developers experiment with variations on the theme, variously called by such names as Scrum and Agile.

Linear, waterfall methods make logical sense. It makes sense that design research should precede design, design precede engineering development, engineering precede manufacturing, and so on. Iteration makes sense in helping to clarify the problem statement and requirements; but when projects are large, involving considerable people, time, and budget, it would be horribly expensive to allow iteration to last too long. On the other hand, proponents of iterative development have seen far too many project teams rush to develop requirements that later prove to be faulty, sometimes wasting huge amounts of money as a result. Numerous large projects have failed at a cost of multiple billions of dollars.

The most traditional waterfall methods are called *gated* methods because they have a linear set of phases or stages, with a gate blocking transition from one stage to the next. The gate is a management review during which progress is evaluated and the decision to proceed to the next stage is made.

Which method is superior? As is invariably the case where fierce debate is involved, both have virtues and both have deficits. In design, one of the most difficult activities is to get the specifications right: in other words, to determine that the correct problem is being solved. Iterative methods are designed to defer the formation of rigid specifications, to start off by diverging across a large set of possible requirements or problem statements before convergence, then again diverging across a large number of potential solutions before

converging. Early prototypes have to be tested through real interaction with the target population in order to refine the requirements.

The iterative method, however, is best suited for the early design phases of a product, not for the later stages. It also has difficulty scaling its procedures to handle large projects. It is extremely difficult to deploy successfully on projects that involve hundreds or even thousands of developers, take years to complete, and cost in the millions or billions of dollars. These large projects include complex consumer goods and large programming jobs, such as automobiles; operating systems for computers, tablets, and phones; and word processors and spreadsheets.

Decision gates give management much better control over the process than they have in the iterative methods. However, they are cumbersome. The management reviews at each of the gates can take considerable time, both in preparation for them and then in the decision time after the presentations. Weeks can be wasted because of the difficulty of scheduling all the senior executives from the different divisions of the company who wish to have a say.

Many groups are experimenting with different ways of managing the product development process. The best methods combine the benefits of both iteration and stage reviews. Iteration occurs inside the stages, between the gates. The goal is to have the best of both worlds: iterative experimentation to refine the problem and the solution, coupled with management reviews at the gates.

The trick is to delay precise specification of the product requirements until some iterative testing with rapidly deployed prototypes has been done, while still keeping tight control over schedule, budget, and quality. It may appear impossible to prototype some large projects (for example, large transportation systems), but even there a lot can be done. The prototypes might be scaled objects, constructed by model makers or 3-D printing methods. Even well-rendered drawings and videos of cartoons or simple animation sketches can be useful. Virtual reality computer aids allow people to envision themselves using the final product, and in the case of a building, to envision living or working within it. All of these methods can provide rapid feedback before much time or money has been expended.

The hardest part of the development of complex products is management: organizing and communicating and synchronizing the many different people, groups, and departmental divisions that are required to make it happen. Large projects are especially difficult, not only because of the problem of managing so many different people and groups, but also because the projects' long time horizon introduces new difficulties. In the many years it takes to go from project formulation to completion, the requirements and technologies will probably change, making some of the proposed work irrelevant and obsolete; the people who will make use of the results might very well change; and the people involved in executing the project definitely will change.

Some people will leave the project, perhaps because of illness or injury, retirement or promotion. Some will change companies and others will move on to other jobs in the same company. Whatever the reason, considerable time is lost finding replacements and then bringing them up to the full knowledge and skill level required. Sometimes this is not even possible because critical knowledge about project decisions and methods are in the form we call *implicit knowledge*; that is, within the heads of the workers. When workers leave, their implicit knowledge goes with them. The management of large projects is a difficult challenge.

What I Just Told You? It Doesn't Really Work That Way

The preceding sections describe the human-centered design process for product development. But there is an old joke about the difference between theory and practice:

> *In theory, there is no difference between theory and practice.*
> *In practice, there is.*

The HCD process describes the ideal. But the reality of life within a business often forces people to behave quite differently from that ideal. One disenchanted member of the design team for a consumer products company told me that although his company professes to believe in user experience and to follow human-centered design, in practice there are only two drivers of new products:

1. Adding features to match the competition
2. Adding some feature driven by a new technology

"Do we look for human needs?" he asked, rhetorically. "No," he answered himself.

This is typical: market-driven pressures plus an engineering-driven company yield ever-increasing features, complexity, and confusion. But even companies that do intend to search for human needs are thwarted by the severe challenges of the product development process, in particular, the challenges of insufficient time and insufficient money. In fact, having watched many products succumb to these challenges, I propose a "Law of Product Development":

DON NORMAN'S LAW OF PRODUCT DEVELOPMENT

> *The day a product development process starts, it is behind schedule and above budget.*

Product launches are always accompanied by schedules and budgets. Usually the schedule is driven by outside considerations, including holidays, special product announcement opportunities, and even factory schedules. One product I worked on was given the unrealistic timeline of four weeks because the factory in Spain would then go on vacation, and when the workers returned, it would be too late to get the product out in time for the Christmas buying season.

Moreover, product development takes time even to get started. People are never sitting around with nothing to do, waiting to be called for the product. No, they must be recruited, vetted, and then transitioned off their current jobs. This all takes time, time that is seldom scheduled.

So imagine a design team being told that it is about to work on a new product. "Wonderful," cries the team; "we'll immediately send out our design researchers to study target customers." "How long will that take?" asks the product manager. "Oh, we can do it quickly: a week or two to make the arrangements, and then two weeks in the field. Perhaps a week to distill the findings. Four or five weeks." "Sorry," says the product manager, "we don't have time. For that matter, we don't have the budget to send a team into the field for two weeks." "But it's essential if we really want to understand the customer," argues the design team. "You're absolutely right," says the product manager, "but we're behind schedule: we can't afford either the time or the money. Next time. Next time we will do it right." Except there is never a next time, because when the next time comes around, the same arguments get repeated: that product also starts behind schedule and over budget.

Product development involves an incredible mix of disciplines, from designers to engineers and programmers, manufacturing, packaging, sales, marketing, and service. And more. The product has to appeal to the current customer base as well as to expand beyond to new customers. Patents create a minefield for designers and engineers, for today it is almost impossible to design or build anything that doesn't conflict with patents, which means redesign to work one's way through the mines.

Each of the separate disciplines has a different view of the product, each has different but specific requirements to be met. Often the requirements posed by each discipline are contradictory or incompatible with those of the other disciplines. But all of them are correct when viewed from their respective perspective. In most companies, however, the disciplines work separately, design passing its results to engineering and programming, which modify the requirements to fit their needs. They then pass their results to manufacturing, which does further modification, then marketing requests changes. It's a mess.

What is the solution?

The way to handle the time crunch that eliminates the ability to do good up-front design research is to separate that process from the product team: have design researchers always out in the field, always studying potential products and customers. Then, when the product team is launched, the designers can say, "We already

examined this case, so here are our recommendations." The same argument applies to market researchers.

The clash of disciplines can be resolved by multidisciplinary teams whose participants learn to understand and respect the requirements of one another. Good product development teams work as harmonious groups, with representatives from all the relevant disciplines present at all times. If all the viewpoints and requirements can be understood by all participants, it is often possible to think of creative solutions that satisfy most of the issues. Note that working with these teams is also a challenge. Everyone speaks a different technical language. Each discipline thinks it is the most important part of the process. Quite often, each discipline thinks the others are stupid, that they are making inane requests. It takes a skilled product manager to create mutual understanding and respect. But it can be done.

The design practices described by the double-diamond and the human-centered design process are the ideal. Even though the ideal can seldom be met in practice, it is always good to aim for the ideal, but to be realistic about the time and budgetary challenges. These can be overcome, but only if they are recognized and designed into the process. Multidisciplinary teams allow for enhanced communication and collaboration, often saving both time and money.

The Design Challenge

It is difficult to do good design. That is why it is such a rich, engaging profession with results that can be powerful and effective. Designers are asked to figure out how to manage complex things, to manage the interaction of technology and people. Good designers are quick learners, for today they might be asked to design a camera; tomorrow, to design a transportation system or a company's organizational structure. How can one person work across so many different domains? Because the fundamental principles of designing for people are the same across all domains. People are the same, and so the design principles are the same.

Designers are only one part of the complex chain of processes and different professions involved in producing a product. Although the theme of this book is the importance of satisfying the needs of the people who will ultimately use the product, other aspects of the product are important; for example, its engineering effectiveness, which includes its capabilities, reliability, and serviceability; its cost; and its financial viability, which usually means profitability. Will people buy it? Each of these aspects poses its own set of requirements, sometimes ones that appear to be in opposition to those of the other aspects. Schedule and budget are often the two most severe constraints.

Designers try hard to determine people's real needs and to fulfill them, whereas marketing is concerned with determining what people will actually buy. What people need and what they buy are two different things, but both are important. It doesn't matter how great the product is if nobody buys it. Similarly, if a company's products are not profitable, the company might very well go out of business. In dysfunctional companies, each division of the company is skeptical of the value added to the product by the other divisions.

In a properly run organization, team members coming from all the various aspects of the product cycle get together to share their requirements and to work harmoniously to design and produce a product that satisfies them, or at least that does so with acceptable compromises. In dysfunctional companies, each team works in isolation, often arguing with the other teams, often watching its designs or specifications get changed by others in what each team considers an unreasonable way. Producing a good product requires a lot more than good technical skills: it requires a harmonious, smoothly functioning, cooperative and respectful organization.

The design process must address numerous constraints. In the sections that follow, I examine these other factors.

PRODUCTS HAVE MULTIPLE, CONFLICTING REQUIREMENTS

Designers must please their clients, who are not always the end users. Consider major household appliances, such as stoves, refrigerators, dishwashers, and clothes washers and dryers; and even faucets and thermostats for heating and air-conditioning systems.

They are often purchased by housing developers or landlords. In businesses, purchasing departments make decisions for large companies; and owners or managers, for small companies. In all these cases, the purchaser is probably interested primarily in price, perhaps in size or appearance, almost certainly not in usability. And once devices are purchased and installed, the purchaser has no further interest in them. The manufacturer has to attend to the requirements of these decision makers, because these are the people who actually buy the product. Yes, the needs of the eventual users are important, but to the business, they seem of secondary importance.

In some situations, cost dominates. Suppose, for example, you are part of a design team for office copiers. In large companies, copying machines are purchased by the Printing and Duplicating Center, then dispersed to the various departments. The copiers are purchased after a formal "request for proposals" has gone out to manufacturers and dealers of machines. The selection is almost always based on price plus a list of required features. Usability? Not considered. Training costs? Not considered. Maintenance? Not considered. There are no requirements regarding understandability or usability of the product, even though in the end those aspects of the product can end up costing the company a lot of money in wasted time, increased need for service calls and training, and even lowered staff morale and lower productivity.

The focus on sales price is one reason we get unusable copying machines and telephone systems in our places of employment. If people complained strongly enough, usability could become a requirement in the purchasing specifications, and that requirement could trickle back to the designers. But without this feedback, designers must often design the cheapest possible products because those are what sell. Designers need to understand their customers, and in many cases, the customer is the person who purchases the product, not the person who actually uses it. It is just as important to study those who do the purchasing as it is to study those who use it.

To make matters even more difficult, yet another set of people needs to be considered: the engineers, developers, manufacturing, services, sales, and marketing people who have to translate the ideas from the design team into reality, and then sell and support the product after it is shipped. These groups are users, too, not of the product itself, but of the output of the design team. Designers are used to accommodating the needs of the product users, but they seldom consider the needs of the other groups involved in the product process. But if their needs are not considered, then as the product development moves through the process from design to engineering, to marketing, to manufacturing, and so on, each new group will discover that it doesn't meet their needs, so they will change it. But piecemeal, after-the-fact changes invariably weaken the cohesion of the product. If all these requirements were known at the start of the design process, a much more satisfactory resolution could have been devised.

Usually the different company divisions have intelligent people trying to do what is best for the company. When they make changes to a design, it is because their requirements were not suitably served. Their concerns and needs are legitimate, but changes introduced in this way are almost always detrimental. The best way to counteract this is to ensure that representatives from all the divisions are present during the entire design process, starting with the decision to launch the product, continuing all the way through shipment to customers, service requirements, and repairs and returns. This way, all the concerns can be heard as soon as they are discovered. There must be a multidisciplinary team overseeing the entire design, engineering, and manufacturing process that shares all departmental issues and concerns from day one, so that everyone can design to satisfy them, and when conflicts arise, the group together can determine the most satisfactory solution. Sadly, it is the rare company that is organized this way.

Design is a complex activity. But the only way this complex process comes together is if all the relevant parties work together as a team. It isn't design against engineering, against marketing, against manufacturing: it is design together with all these other players. Design must take into account sales and marketing, servicing and help desks, engineering and manufacturing, costs and

schedules. That's why it's so challenging. That's why it's so much fun and rewarding when it all comes together to create a successful product.

DESIGNING FOR SPECIAL PEOPLE

There is no such thing as the average person. This poses a particular problem for the designer, who usually must come up with a single design for everyone. The designer can consult handbooks with tables that show average arm reach and seated height, how far the average person can stretch backward while seated, and how much room is needed for average hips, knees, and elbows. *Physical anthropometry* is what the field is called. With data, the designer can try to meet the size requirements for almost everyone, say for the 90th, 95th, or even the 99th percentile. Suppose the product is designed to accommodate the 95th percentile, that is, for everyone except the 5 percent of people who are smaller or larger. That leaves out a lot of people. The United States has approximately 300 million people, so 5 percent is 15 million. Even if the design aims at the 99th percentile it would still leave out 3 million people. And this is just for the United States: the world has 7 billion people. Design for the 99th percentile of the world and 70 million people are left out.

Some problems are not solved by adjustments or averages: Average a left-hander with a right-hander and what do you get? Sometimes it is simply impossible to build one product that accommodates everyone, so the answer is to build different versions of the product. After all, we would not be happy with a store that sells only one size and type of clothing: we expect clothing that fits our bodies, and people come in a very wide range of sizes. We don't expect the large variety of goods found in a clothing store to apply to all people or activities; we expect a wide variety of cooking appliances, automobiles, and tools so we can select the ones that precisely match our requirements. One device simply cannot work for everyone. Even such simple tools as pencils need to be designed differently for different activities and types of people.

Consider the special problems of the aged and infirm, the handicapped, the blind or near blind, the deaf or hard of hearing, the very short or very tall, or people who speak other languages. Design for interests and skill levels. Don't be trapped by overly general, inaccurate stereotypes. I return to these groups in the next section.

THE STIGMA PROBLEM

> *"I don't want to go into a care facility. I'd have to be around all those old people."* (Comment by a 95-year-old man.)

Many devices designed to aid people with particular difficulties fail. They may be well designed, they may solve the problem, but they are rejected by their intended users. Why? Most people do not wish to advertise their infirmities. Actually, many people do not wish to admit having infirmities, even to themselves.

When Sam Farber wanted to develop a set of household tools that his arthritic wife could use, he worked hard to find a solution that was good for everyone. The result was a series of tools that revolutionized this field. For example, vegetable peelers used to be an inexpensive, simple metal tool, often of the form shown on the left in Figure 6.3. These were awkward to use, painful to hold, and not even that effective at peeling, but everyone assumed that this was how they had to be.

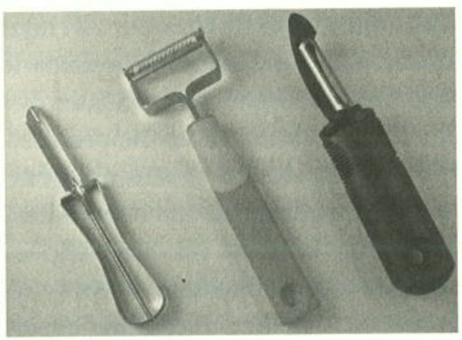

FIGURE 6.3. Three Vegetable Peelers. The traditional metal vegetable peeler is shown on the left: inexpensive, but uncomfortable. The OXO peeler that revolutionized the industry is shown on the right. The result of this revolution is shown in the middle, a peeler from the Swiss company Kuhn Rikon: colorful and comfortable.

After considerable research, Farber settled upon the peeler shown on the right in Figure 6.3 and built a company, OXO, to manufacture and distribute it. Even though the peeler was designed for someone with arthritis, it was advertised as a better peeler for everyone. It was. Even though the design was more expensive than the regular peeler, it was so successful that today, many companies make variations on this theme. You may have trouble seeing the OXO peeler as revolutionary because today, many have followed in these footsteps. Design has become a major theme for even simple tools such as peelers, as demonstrated by the center peeler of Figure 6.3.

Consider the two things special about the OXO peeler: cost and design for someone with an infirmity. Cost? The original peeler was very inexpensive, so a peeler that is many times the cost of the inexpensive one is still inexpensive. What about the special design for people with arthritis? The virtues for them were never mentioned, so how did they find it? OXO did the right thing and let the world know that this was a better product. And the world took note and made it successful. As for people who needed the better handle? It didn't take long for the word to spread. Today, many companies have followed the OXO route, producing peelers that work extremely well, are comfortable, and are colorful. See Figure 6.3.

Would you use a walker, wheelchair, crutches, or a cane? Many people avoid these, even though they need them, because of the negative image they cast: the stigma. Why? Years ago, a cane was fashionable: people who didn't need them would use them anyway, twirling them, pointing with them, hiding brandy or whisky, knives or guns inside their handles. Just look at any movie depicting nineteenth-century London. Why can't devices for those who need them be as sophisticated and fashionable today?

Of all the devices intended to aid the elderly, perhaps the most shunned is the walker. Most of these devices are ugly. They cry out, "Disability here." Why not transform them into products to be proud of? Fashion statements, perhaps. This thinking has already begun with some medical appliances. Some companies are making hearing aids and glasses for children and adolescents with special colors and styles that appeal to these age groups. Fashion accessories. Why not?

Those of you who are young, do not smirk. Physical disabilities may begin early, starting in the midtwenties. By their midforties, most people's eyes can no longer adjust sufficiently to focus over the entire range of distances, so something is necessary to compensate, whether reading glasses, bifocals, special contact lenses, or even surgical correction.

Many people in their eighties and nineties are still in good mental and physical shape, and the accumulated wisdom of their years leads to superior performance in many tasks. But physical strength and agility do decrease, reaction time slows, and vision and hearing show impairments, along with decreased ability to divide attention or switch rapidly among competing tasks.

For anyone who is considering growing old, I remind you that although physical abilities diminish with age, many mental capacities continue to improve, especially those dependent upon an expert accumulation of experience, deep reflection, and enhanced knowledge. Younger people are more agile, more willing to experiment and take risks. Older people have more knowledge and wisdom. The world benefits from having a mix and so do design teams.

Designing for people with special needs is often called *inclusive* or *universal design.* Those names are fitting, for it is often the case that everyone benefits. Make the lettering larger, with high-contrast type, and everyone can read it better. In dim light, even the people with the world's best eyesight will benefit from such lettering. Make things adjustable, and you will find that more people can use it, and even people who liked it before may now like it better. Just as I invoke the so-called error message of Figure 4.6 as my normal way of exiting a program because it is easier than the so-called correct way, special features made for people with special needs often turn out to be useful for a wide variety of people.

The best solution to the problem of designing for everyone is flexibility: flexibility in the size of the images on computer screens, in the sizes, heights, and angles of tables and chairs. Allow people to adjust their own seats, tables, and working devices. Allow them to adjust lighting, font size, and contrast. Flexibility on our highways might mean ensuring that there are alternative routes with different speed limits. Fixed solutions will invariably fail with some

people; flexible solutions at least offer a chance for those with different needs.

Complexity Is Good; It Is Confusion That Is Bad

The everyday kitchen is complex. We have multiple instruments just for serving and eating food. The typical kitchen contains all sorts of cutting utensils, heating units, and cooking apparatus. The easiest way to understand the complexity is to try to cook in an unfamiliar kitchen. Even excellent cooks have trouble working in a new environment.

Someone else's kitchen looks complicated and confusing, but your own kitchen does not. The same can probably be said for every room in the home. Notice that this feeling of confusion is really one of knowledge. My kitchen looks confusing to you, but not to me. In turn, your kitchen looks confusing to me, but not to you. So the confusion is not in the kitchen: it is in the mind. "Why can't things be made simple?" goes the cry. Well, one reason is that life is complex, as are the tasks we encounter. Our tools must match the tasks.

I feel so strongly about this that I wrote an entire book on the topic, *Living with Complexity,* in which I argued that complexity is essential: it is confusion that is undesirable. I distinguished between "complexity," which we need to match the activities we take part in, and "complicated," which I defined to mean "confusing." How do we avoid confusion? Ah, here is where the designer's skills come into play.

The most important principle for taming complexity is to provide a good conceptual model, which has already been well covered in this book. Remember the kitchen's apparent complexity? The people who use it understand why each item is stored where it is: there is usually structure to the apparent randomness. Even exceptions fit: even if the reason is something like, "It was too big to fit in the proper drawer and I didn't know where else to put it," that is reason enough to give structure and understanding to the person who stored the item there. Complex things are no longer complicated once they are understood.

Standardization and Technology

If we examine the history of advances in all technological fields, we see that some improvements come naturally through the technology itself, others come through standardization. The early history of the automobile is a good example. The first cars were very difficult to operate. They required strength and skill beyond the abilities of many. Some problems were solved through automation: the choke, the spark advance, and the starter engine. Other aspects of cars and driving were standardized through the long process of international standards committees:

- On which side of the road to drive (constant within a country, but variable across countries)
- On which side of the car the driver sits (depends upon which side of the road the car is driven)
- The location of essential components: steering wheel, brake, clutch, and accelerator (the same, whether on the left- or right-hand side of the car)

Standardization is one type of cultural constraint. With standardization, once you have learned to drive one car, you feel justifiably confident that you can drive any car, anyplace in the world. Standardization provides a major breakthrough in usability.

ESTABLISHING STANDARDS

I have enough friends on national and international standards committees to realize that the process of determining an internationally accepted standard is laborious. Even when all parties agree on the merits of standardization, the task of selecting standards becomes a lengthy, politicized issue. A small company can standardize its products without too much difficulty, but it is much more difficult for an industrial, national, or international body to

agree to standards. There even exists a standardized procedure for establishing national and international standards. A set of national and international organizations works on standards; when a new standard is proposed, it must work its way through the organizational hierarchy. Each step is complex, for if there are three ways of doing something, then there are sure to be strong proponents of each of the three ways, plus people who will argue that it is too early to standardize.

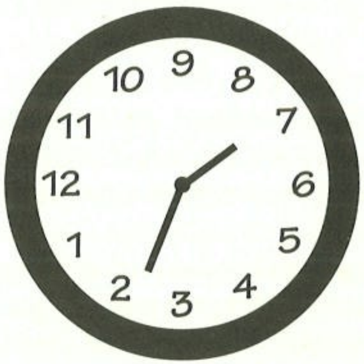

FIGURE 6.4. The Nonstandard Clock. What time is it? This clock is just as logical as the standard one, except the hands move in the opposite direction and "12" is not in its usual place. Same logic, though. So why is it so difficult to read? What time is being displayed? 7:11, of course.

Each proposal is debated at the standards committee meeting where it is presented, then taken back to the sponsoring organization—which is sometimes a company, sometimes a professional society—where objections and counter-objections are collected. Then the standards committee meets again to discuss the objections. And again and again and again. Any company that is already marketing a product that meets the proposed standard will have a huge economic advantage, and the debates are therefore often affected as much by the economics and politics of the issues as by real technological substance. The process is almost guaranteed to take five years, and quite often longer.

The resulting standard is usually a compromise among the various competing positions, oftentimes an inferior compromise. Sometimes the answer is to agree on several incompatible standards. Witness the existence of both metric and English units; of left-hand- and right-hand-drive automobiles. There are several international standards for the voltages and frequencies of electricity, and several different kinds of electrical plugs and sockets—which cannot be interchanged.

WHY STANDARDS ARE NECESSARY: A SIMPLE ILLUSTRATION

With all these difficulties and with the continual advances in technology, are standards really necessary? Yes, they are. Take the everyday clock. It's standardized. Consider how much trouble you would have telling time with a backward clock, where the hands revolved "counterclockwise." A few such clocks exist, primarily as humorous conversation pieces. When a clock truly violates standards, such as the one in Figure 6.4 on the previous page, it is difficult to determine what time is being displayed. Why? The logic behind the time display is identical to that of conventional clocks: there are only two differences—the hands rotate in the opposite direction (counterclockwise) and the location of "12," usually at the top, has been moved. This clock is just as logical as the standard one. It bothers us because we have standardized on a different scheme, on the very definition of the term *clockwise*. Without such standardization, clock reading would be more difficult: you'd always have to figure out the mapping.

A STANDARD THAT TOOK SO LONG, TECHNOLOGY OVERRAN IT

I myself participated at the very end of the incredibly long, complex political process of establishing the US standards for high-definition television. In the 1970s, the Japanese developed a national television system that had much higher resolution than the standards then in use: they called it "high-definition television."

In 1995, two decades later, the television industry in the United States proposed its own high-definition TV standard (HDTV) to the Federal Communications Commission (FCC). But the computer industry pointed out that the proposals were not compatible with the way that computers displayed images, so the FCC objected to the proposed standards. Apple mobilized other members of the industry and, as vice president of advanced technology, I was selected to be the spokesperson for Apple. (In the following description, ignore the jargon—it doesn't matter.) The TV industry proposed a

wide variety of permissible formats, including ones with rectangular pixels and interlaced scan. Because of the technical limitations in the 1990s, it was suggested that the highest-quality picture have 1,080 interlaced lines (1080i). We wanted only progressive scan, so we insisted upon 720 lines, progressively displayed (720p), arguing that the progressive nature of the scan made up for the lesser number of lines.

The battle was heated. The FCC told all the competing parties to lock themselves into a room and not to come out until they had reached agreement. As a result, I spent many hours in lawyers' offices. We ended up with a crazy agreement that recognized multiple variations of the standard, with resolutions of 480i and 480p (called *standard definition*), 720p and 1080i (called *high-definition*), and two different aspect ratios for the screens (the ratio of width to height), 4:3 (= 1.3)—the old standard—and 16:9 (= 1.8)—the new standard. In addition, a large number of frame rates were supported (basically, how many times per second the image was transmitted). Yes, it was a standard, or more accurately a large number of standards. In fact, one of the allowed methods of transmission was to use any method (as long as it carried its own specifications along with the signal). It was a mess, but we did reach agreement. After the standard was made official in 1996, it took roughly ten more years for HDTV to become accepted, helped, finally, by a new generation of television displays that were large, thin, and inexpensive. The whole process took roughly thirty-five years from the first broadcasts by the Japanese.

Was it worth the fight? Yes and no. In the thirty-five years that it took to reach the standard, the technology continued to evolve, so the resulting standard was far superior to the first one proposed so many years before. Moreover, the HDTV of today is a huge improvement over what we had before (now called "standard definition"). But the minutiae of details that were the focus of the fight between the computer and TV companies was silly. My technical experts continually tried to demonstrate to me the superiority of 720p images over 1080i, but it took me hours of viewing special scenes under expert guidance to see the deficiencies of the interlaced images (the differences only show up with complex moving images). So why did we care?

Television displays and compression techniques have improved so much that interlacing is no longer needed. Images at 1080p, once thought to be impossible, are now commonplace. Sophisticated algorithms and high-speed processors make it possible to transform one standard into another; even rectangular pixels are no longer a problem.

As I write these words, the main problem is the discrepancy in aspect ratios. Movies come in many different aspect ratios (none of them the new standard) so when TV screens show movies, they either have to cut off part of the image or leave parts of the screen black. Why was the HDTV aspect ratio set at 16:9 (or 1.8) if no movies used that ratio? Because engineers liked it: square the old aspect ratio of 4:3 and you get the new one, 16:9.

Today we are about to embark on yet another standards fight over TV. First, there is three-dimensional TV: 3-D. Then there are proposals for ultra-high definition: 2,160 lines (and a doubling of the horizontal resolution as well): four times the resolution of our best TV today (1080p). One company wants eight times the resolution, and one is proposing an aspect ratio of 21:9 (= 2.3). I have seen these images and they are marvelous, although they only matter with large screens (at least 60 inches, or 1.5 meters, in diagonal length), and when the viewer is close to the display.

Standards can take so long to be established that by the time they do come into wide practice, they can be irrelevant. Nonetheless, standards are necessary. They simplify our lives and make it possible for different brands of equipment to work together in harmony.

A STANDARD THAT NEVER CAUGHT ON: DIGITAL TIME

Standardize and you simplify lives: everyone learns the system only once. But don't standardize too soon; you may be locked into a primitive technology, or you may have introduced rules that turn out to be grossly inefficient, even error-inducing. Standardize too late, and there may already be so many ways of doing things that no international standard can be agreed on. If there is agreement on an old-fashioned technology, it may be too expensive for everyone to change to the new standard. The metric system is a good example: it is a far simpler and more usable scheme for representing distance, weight, volume, and temperature than the older English system of feet, pounds, seconds, and degrees on the Fahrenheit scale. But industrial nations with a heavy commitment to the old measurement standard claim they cannot afford the massive costs and confusion of conversion. So we are stuck with two standards, at least for a few more decades.

Would you consider changing how we specify time? The current system is arbitrary. The day is divided into twenty-four rather arbitrary but standard units—hours. But we tell time in units of twelve, not twenty-four, so there have to be two cycles of twelve hours each, plus the special convention of a.m. and p.m. so we know which cycle we are talking about. Then we divide each hour into sixty minutes and each minute into sixty seconds.

What if we switched to metric divisions: seconds divided into tenths, milliseconds, and microseconds? We would have days, millidays, and microdays. There would have to be a new hour, minute, and second: call them the digital hour, the digital minute, and the digital second. It would be easy: ten digital hours to the day, one hundred digital minutes to the digital hour, one hundred digital seconds to the digital minute.

Each digital hour would last exactly 2.4 times an old hour: 144 old minutes. So the old one-hour period of the schoolroom or television program would be replaced with a half-digital hour period, or 50 digital minutes—only 20 percent longer than the current hour. We could adapt to the differences in durations with relative ease.

What do I think of it? I much prefer it. After all, the decimal system, the basis of most of the world's use of numbers and arithmetic, uses base 10 arithmetic and, as a result, arithmetic operations are much simpler in the metric system. Many societies have used other systems, 12 and 60 being common. Hence twelve for the number of items in a dozen, inches in a foot, hours in a day, and months in a year; sixty for the number of seconds in a minute, seconds in a degree, and minutes in an hour.

The French proposed that time be made into a decimal system in 1792, during the French Revolution, when the major shift to the metric system took place. The metric system for weights and lengths took hold, but not for time. Decimal time was used long enough for decimal clocks to be manufactured, but it eventually was discarded. Too bad. It is very difficult to change well-established habits. We still use the QWERTY keyboard, and the United States still measures things in inches and feet, yards and miles, Fahrenheit, ounces, and pounds. The world still measures time in units of 12 and 60, and divides the circle into 360 degrees.

In 1998, Swatch, the Swiss watch company, made its own attempt to introduce decimal time through what it called "Swatch International Time." Swatch divided the day into 1,000 ".beats," each .beat being slightly less than 90 seconds (each .beat corresponds to one digital minute). This system did not use time zones, so people the world over would be in synchrony with their watches. This does not simplify the problem of synchronizing scheduled conversations, however, because it would be difficult to get the sun to behave properly. People would still wish to wake up around sunrise, and this would occur at different Swatch times around the world. As a result, even though people would have their watches synchronized, it would still be necessary to know when they woke up, ate, went to and from work, and went to sleep, and these times would vary around the world. It isn't clear whether Swatch was serious with its proposal or whether it was one huge advertising stunt. After a few years of publicity, during which the company manufactured digital watches that told the time in .beats, it all fizzled away.

Speaking of standardization, Swatch called its basic time unit a ".beat" with the first character being a period. This nonstandard spelling wreaks havoc on spelling correction systems that aren't set up to handle words that begin with punctuation marks.

Deliberately Making Things Difficult

How can good design (design that is usable and understandable) be balanced with the need for "secrecy" or privacy, or protection? That is, some applications of design involve areas that are sensitive and necessitate strict control over who uses and understands them. Perhaps we don't want any user-in-the-street to understand enough of a system to compromise its security. Couldn't it be argued that some things shouldn't be designed well? Can't things be left cryptic, so that only those who have clearance, extended education, or whatever, can make use of the system? Sure, we have passwords, keys, and other types of security checks, but this can become wearisome for the privileged user. It appears that if good design is not ignored in some contexts, the purpose for the existence of the system will be nullified. (A computer mail question sent to me by a student, Dina Kurktchi. It is just the right question.)

In Stapleford, England, I came across a school door that was very difficult to open, requiring simultaneous operation of two latches, one at the very top of the door, the other down low. The latches were difficult to find, to reach, and to use. But the difficulties were deliberate. This was good design. The door was at a school for handicapped children, and the school didn't want the children to be able to get out to the street without an adult. Only adults were large enough to operate the two latches. Violating the rules of ease of use is just what was needed.

Most things are intended to be easy to use, but aren't. But some things are deliberately difficult to use—and ought to be. The number of things that should be difficult to use is surprisingly large:

- Any door designed to keep people in or out.
- Security systems, designed so that only authorized people will be able to use them.
- Dangerous equipment, which should be restricted.
- Dangerous operations that might lead to death or injury if done accidentally or in error.

- Secret doors, cabinets, and safes: you don't want the average person even to know that they are there, let alone to be able to work them.
- Cases deliberately intended to disrupt the normal routine action (as discussed in Chapter 5). Examples include the acknowledgment required before permanently deleting a file from a computer, safeties on pistols and rifles, and pins in fire extinguishers.
- Controls that require two simultaneous actions before the system will operate, with the controls separated so that it takes two people to work them, preventing a single person from doing an unauthorized action (used in security systems or safety-critical operations).
- Cabinets and bottles for medications and dangerous substances deliberately made difficult to open to keep them secure from children.
- Games, a category in which designers deliberately flout the laws of understandability and usability. Games are meant to be difficult; in some games, part of the challenge is to figure out what is to be done, and how.

Even where a lack of usability or understandability is deliberate, it is still important to know the rules of understandable and usable design, for two reasons. First, even deliberately difficult designs aren't entirely difficult. Usually there is one difficult part, designed to keep unauthorized people from using the device; the rest of it should follow the normal principles of good design. Second, even if your job is to make something difficult to do, you need to know how to go about doing it. In this case, the rules are useful, for they state in reverse just how to go about the task. You could systematically violate the rules like this:

- Hide critical components: make things invisible.
- Use unnatural mappings for the execution side of the action cycle, so that the relationship of the controls to the things being controlled is inappropriate or haphazard.
- Make the actions physically difficult to do.
- Require precise timing and physical manipulation.
- Do not give any feedback.

- Use unnatural mappings for the evaluation side of the action cycle, so that system state is difficult to interpret.

Safety systems pose a special problem in design. Oftentimes, the design feature added to ensure safety eliminates one danger, only to create a secondary one. When workers dig a hole in a street, they must put up barriers to prevent cars and people from falling into the hole. The barriers solve one problem, but they themselves pose another danger, often mitigated by adding signs and flashing lights to warn of the barriers. Emergency doors, lights, and alarms must often be accompanied by warning signs or barriers that control when and how they can be used.

Design: Developing Technology for People

Design is a marvelous discipline, bringing together technology and people, business and politics, culture and commerce. The different pressures on design are severe, presenting huge challenges to the designer. At the same time, the designers must always keep foremost in mind that the products are to be used by people. This is what makes design such a rewarding discipline: On the one hand, woefully complex constraints to overcome; on the other hand, the opportunity to develop things that assist and enrich the lives of people, that bring benefits and enjoyment.