10

# People and Prototypes

The author's own ideas about how to design interactions, with help from his friends and colleagues Jane Fulton Suri and Duane Bray

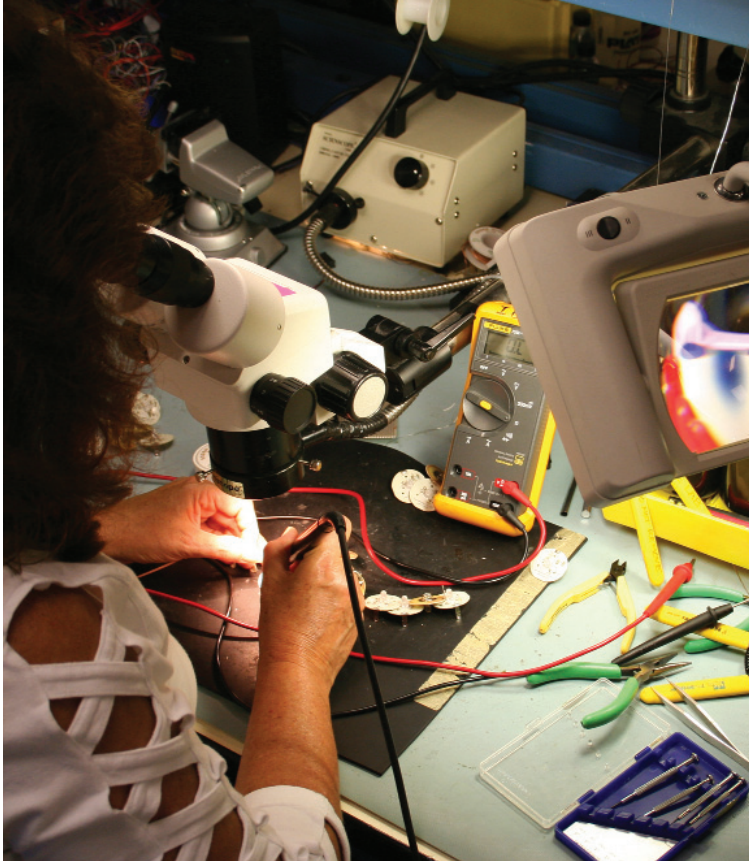
Prototype early and often, making each iterative step a little more realistic. At some point you are likely to experience that wonderful "Ah ha!" feeling that comes with a creative leap, but that is only an indication that you have moved forward in the detail of the aspect of the design that you are focusing on right then. You will only know that the design is good when you have tried it out with the people who will use it and found that they are pleased, excited, motivated, and satisfied with the result.

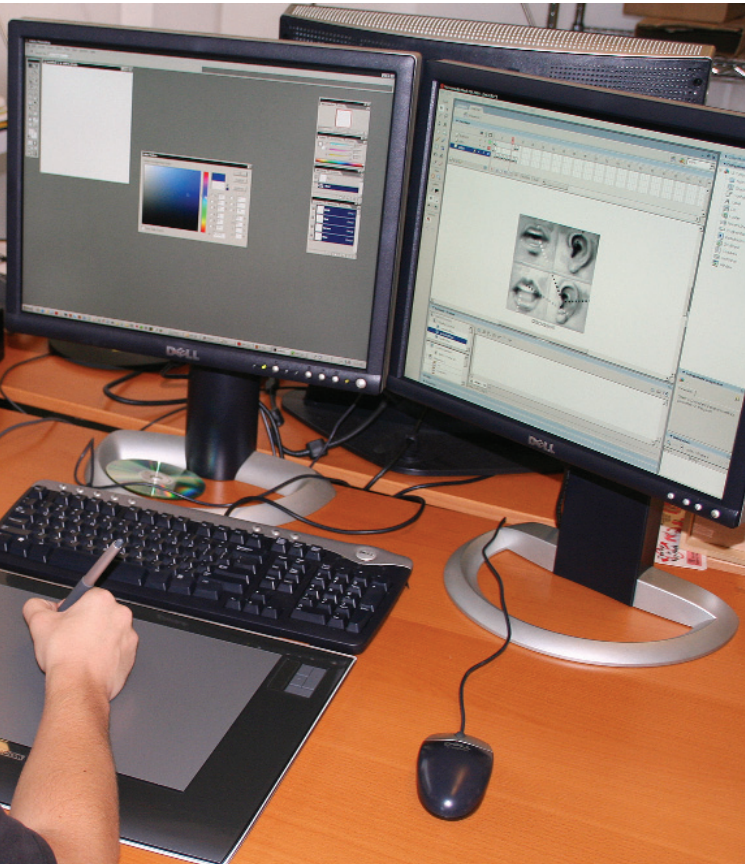
■ Various people

*Photos*
Courtesy of IDEO

In the Introduction I told two stories of personal experiences that lead me to start working toward a new design discipline, eventually called "interaction design," and said that I would talk more about my personal point of view in this chapter. One of my motivations for embarking on this book in the first place was the hope that I could move beyond being an interaction design practitioner and find a way of explaining it to other people. The title of this chapter gives me away; I believe that if we think first about people and then try, try, and try again to prototype our designs, we stand a good chance of creating innovative solutions that people will value and enjoy.

To "understand people and use prototypes for speed" is a good pragmatic summary of a way of answering a "how?" question, but it does not answer questions of what, why, or where. In the first section of this chapter I try to answer these other questions about interaction design from my perspective as a practitioner. Please forgive me if my theoretical explanations are limited; my excuse is that I am a designer. I can easily tell you what to do but find it difficult to articulate the rationale, and yes, I will try to give a reason for that difficulty.





I am helped by two of my friends and colleagues in telling a more complete story about people and prototypes. Jane Fulton Suri was the first human factors psychologist to join my design team. She has pioneered the integration of human factors into the design process at IDEO. She helps me explain the methods that we have developed to learn about people and derive insights from that knowledge to inspire design. She also explores the use of prototypes for understanding existing experiences, investigating design ideas, and communicating design concepts.

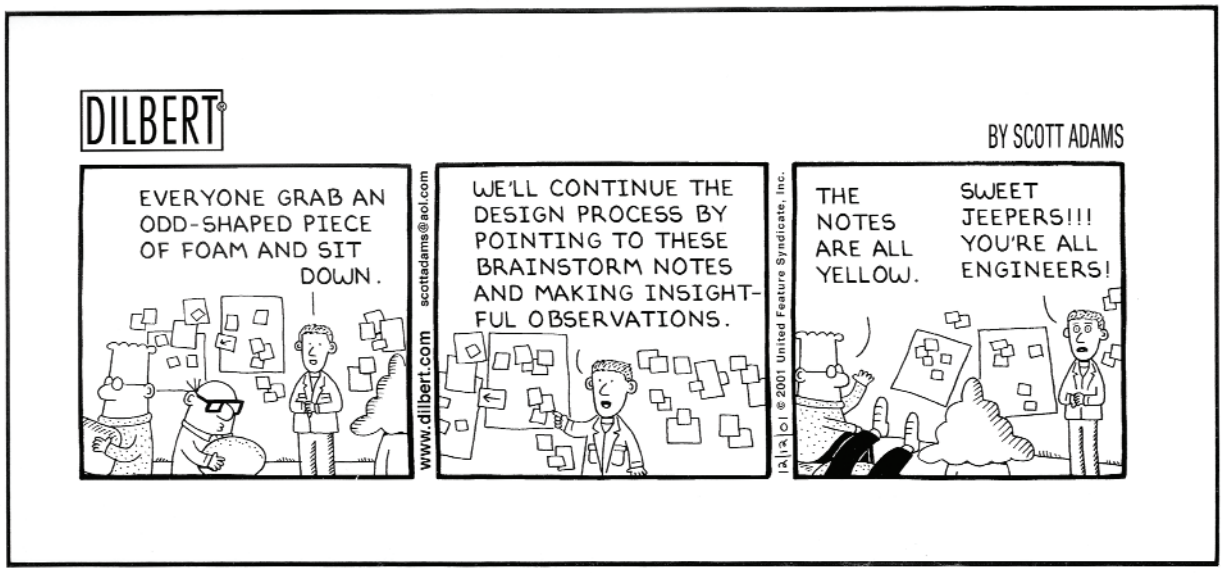
Duane Bray leads the interaction design discipline at IDEO, so I have enlisted his help for an analysis of prototyping techniques. We talk about how prototyping fits into the design process, explain the different sorts of prototypes that are useful for the various categories of interaction design, and try to predict how that will change the practice of design in the future.

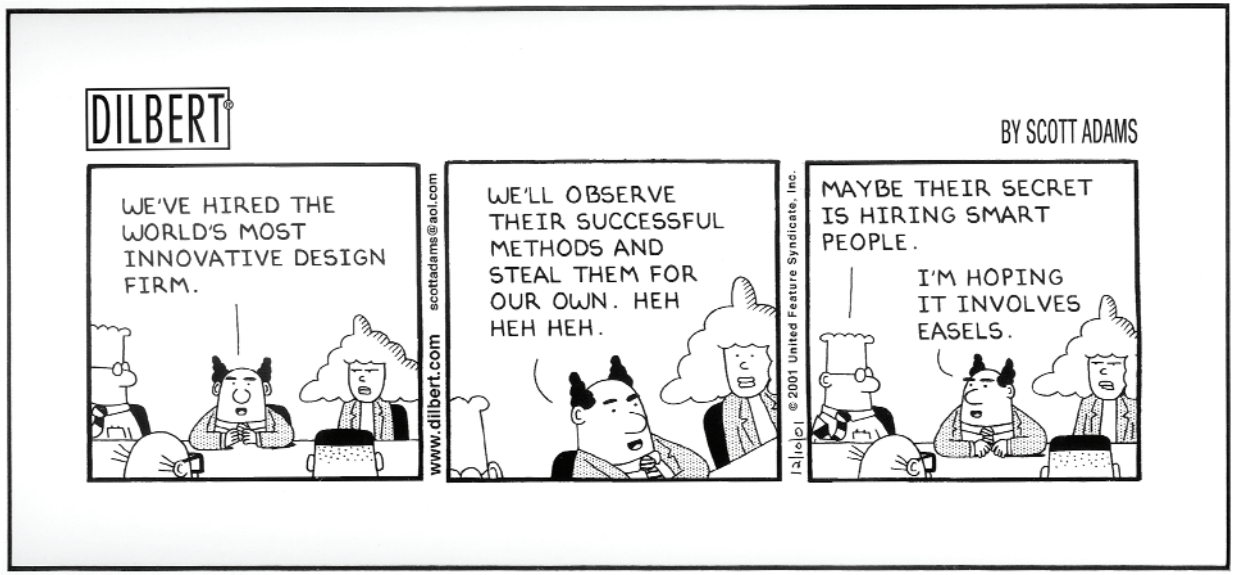
In the final section, about process, I characterize the difference between designing something new and designing a new version of something. With this contrast in mind, I propose a general method for designing interactions successfully.

Examples of prototyping environments at IDEO

*Clockwise from top left*

- Machine shop
  *Photo* Roberto Carra
- EE Lab
  *Photo* Author
- Video prototypes
  *Photo* Author
- Animation software
  *Photo* Craig Syverson



# Designing Interactions

Design is the conception and planning of the artificial.

Richard Buchanan, 1995

## What Is Design?

Dilbert cartoons about design

- *Top* Process
- *Bottom* Intellectual property

*Cartoons* by Scott Adams, courtesy of United Media

In the foreword, Gillian Crampton Smith summarized the design of interactions as being about shaping our everyday lives through digital artifacts—for work, play, and entertainment. Her essay gives us a very good understanding of the special attributes of interaction design when we compare it to other design disciplines, and if we are already fluent in our understanding and appreciation of design. However, it assumes that we know the answer to the question, What is design?

If you ask a designer for a definition of design, you are often answered with a smirk, a joke, or a change of subject, as design is notoriously difficult to define, and designers are much more at ease learning and knowing by doing than they are explaining. In 1995 the British Design Council put out a little book called "Definitions of Design," which was arrived at by asking fifty people—designers, children, and others—to give their personal definitions. The result was surprisingly uninformative, but entertaining. Here are four examples:

I believe design is an intention, purpose, plan: and that good design is therefore by inference, where such plan has been well conceived, well executed, and of benefit to someone.
Milner Gray, Designer

Design is all around us—either we control it—or it controls us.
Wally Olins, Chairman Wolff Olins

Design is the difference between doing it, and doing it right.
Mark Fisher MP, Co-chairman, All-Party Group on Design

With art—if you like, you can be really weird. But in design you have to think about what other people will like.
Ghisli, age 10

These quotes, like an impressionist painting, give you a sense of what is meant when you look from a distance, but they are not satisfying as definitions. This vagueness remains an accepted fact of life for design. If you visit the current Web site of the British Design Council,[1] the rhetorical question, What is design? is posed, and the answer given is, "Design is everywhere—and that's why looking for a definition may not help you grasp what it is."

The most satisfying definitive description of design that I have encountered is the statement by Charles Eames in conversation with Madame Amic. Here are the questions and answers from that conversation that seem particularly relevant to designing interactions:

Q. What is your definition of "Design?"
A. A plan for arranging elements in such a way as to best accomplish a particular purpose.

Q. Is design an expression of art (an art form)?
A. The design is an expression of the purpose. It may (if it is good enough) later be judged as art.

Q. Is design a craft for industrial purposes?
A. No—but design may be a solution to some industrial problems.

Q. What are the boundaries of design?
A. What are the boundaries of problems?

■ Charles and Ray Eames with a model of the exhibition "Mathematica," 1960

Q. Does the creation of design admit constraint?
A. Design depends largely on constraints.

Q. What constraints?
A. The sum of all constraints. Here is one of the few effective keys to the design problem—the ability of the designer to recognize as many of the constraints as possible—his willingness and enthusiasm for working within these constraints—the constraints of price, of size, of strength, balance, of surface, of time etc.; each problem has its own peculiar list.

Q. Does design obey laws?
A. Aren't constraints enough?[2]

## Core Skills of Design

Charles Eames was right about constraints; they are key to understanding design. Scientific disciplines rely on the ability of the practitioner to become expert in a narrow field, learning how to focus by excluding extraneous information and thus learning more and more about less and less. Here are five core skills of design:[3]

1 To synthesize a solution from all of the relevant constraints, understanding everything that will make a difference to the result
2 To frame, or reframe, the problem and objective
3 To create and envision alternatives
4 To select from those alternatives, knowing intuitively how to choose the best approach
5 To visualize and prototype the intended solution

I describe a process that includes this list at the end of the chapter. The five skills can be applied in the listed order, but the process is iterative rather than linear and does not necessarily follow a sequence. The most productive approach is often apparently unstructured, where members of the design team may suddenly dive into a prototype, renew some research activity, look

Good design comes from the successful synthesis of a solution that recognizes all the relevant constraints, and the nature of the constraints defines the difference between design disciplines.


- Pinball machine
- The mind is like an iceberg

at people afresh, reexamine some of the constraints, or create new alternative concepts. The process does not look like a linear system diagram, nor even a revolving wheel of iterations, but is more like playing with a pinball machine, where one bounces rapidly in unexpected directions.

## Tacit knowledge

Design thinking harnesses tacit knowledge rather than the explicit knowledge of logically expressed thoughts. Designers operate at a level of complexity in the synthesis of constraints where it is more effective to learn by doing, allowing the subconscious mind to inform intuitions that guide actions.

Perhaps the mind is like an iceberg, with just a small proportion of the overall amount protruding above the water, into consciousness. If we operate above the water line, we only have a small volume to use, but if we allow ourselves to use the whole submerged mass, we have a lot more to work with.

If a problem has a large number of constraints, the conscious mind starts to get confused, but the subconscious mind has a much larger capacity.[4] Designers have the ability and the training to harness the tacit knowledge of the unconscious mind, rather than being limited to working with explicit knowledge. This makes them good at synthesizing complex problems with large numbers of constraints; it also makes them bad at explaining or defining what they are doing or thinking. They will describe process and results because they are not consciously aware of their own rationale.

This is the reason that design education relies on a project-based approach of "learning by doing." The normal academic structure of learning is based on the conscious mind. You learn by understanding, with information that can be explained, elucidated and justified. A PhD is earned by contributing to the body of knowledge, by which we mean explicit knowledge. Designers learn by an atelier process, working on projects; the teacher advises on process as the designs are developed and criticizes the result, but neither teachers nor students are asked to explain the reasoning. When a problem is complex, with lots of constrains, it is much easier to recognize a good solution than to explain it.

These are typical evaluation criteria for a student design project:

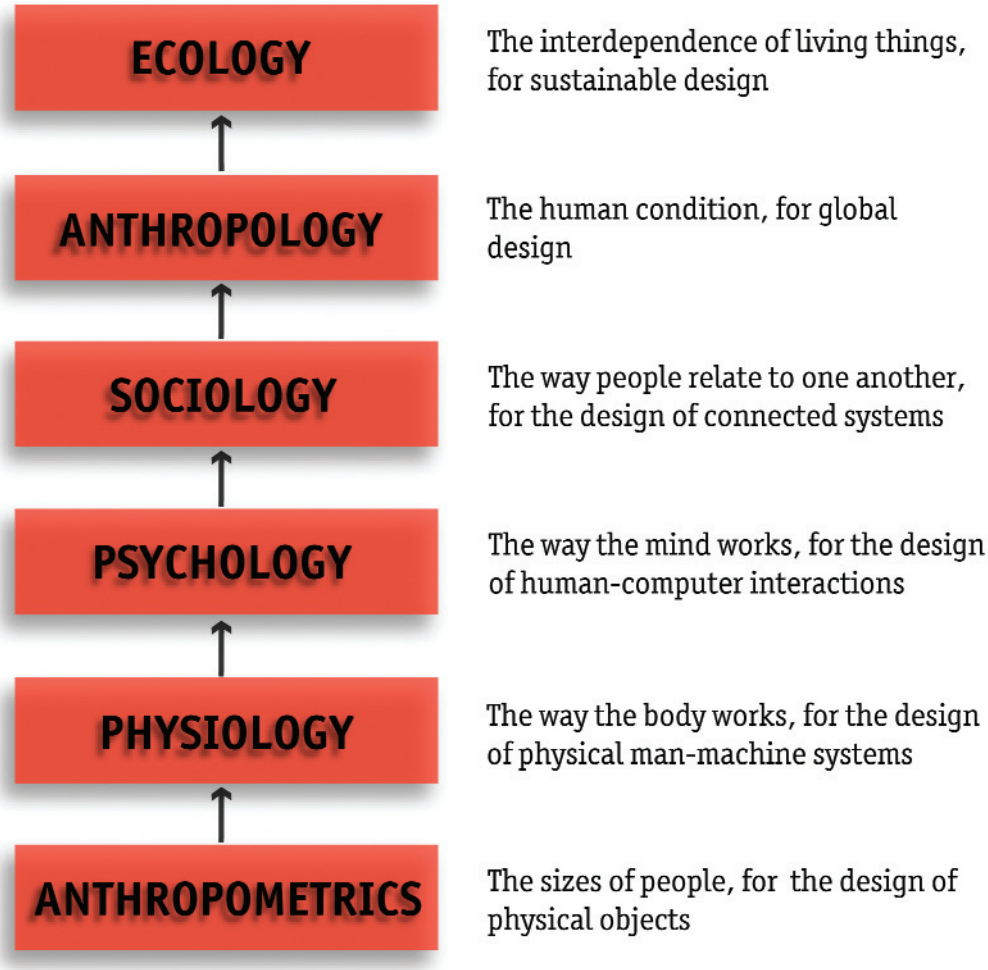
1 Creativity/innovation
2 Aesthetics/quality
3 Human factors/values
4 Performance/technology
5 Completeness/presentation

### Design disciplines

The nature of the constraints defines the design discipline. If you ask why people choose to be product designers, graphic designers, or architects, the answer will be less about their abilities and talent than about the kinds of constraints that they like to work with. Did they like everyday things, two-dimensional images and typography, or the built environment? Once you have been through a long educational process of projects and moved on to expand your experience in practice, it becomes more and more natural and normal for you to collect and understand the appropriate constraints for your design problem. Product designers know about how people relate to physical objects and how to manipulate metals and plastics. Graphic designers learn about how we see images and understand information and how to manipulate marks on paper. Architects become expert in the way we relate to space and learn how to develop structures for people to inhabit.

Designers are both enabled and controlled by the constraints that they learn about and come to understand; they are fluent with their tacit knowledge, in their own media, and in the contexts that they are familiar with and understand. This makes it difficult to develop a new design discipline in response to new kinds of constraints, but design problems are changing all the time. In the introduction I told the story of my own experience facing the complexities of designing a laptop computer and how this triggered my effort to start a new design discipline that we ended up calling "interaction design."

I think it is worth looking at the need for this new discipline as a step on a hierarchy, which forms a continuing trend of increasing complexity.




*This hierarchy shows the increasing complexity of the relevant constraints, if you consider each type of design problem from the point of view of the user. The hierarchy is based on the type of human factors that is relevant to the design context in each level of complexity, starting with the simplest at the bottom.*

# A Hierarchy of Complexity

When I graduated from college as an industrial designer in 1965, I expected to spend my life designing mass-produced objects to be manufactured in metals and plastics. Thinking about what people want from an object was a predominant consideration for the design, but there was an assumption that the most complex aspect would be to think about the subjective and qualitative values that would help the designer to create an appropriate aesthetic, so most of the research into what people wanted was aimed at discovering those subtle values that could inform an intuitive design process. The overall complexity came from synthesizing this understanding with all of the functional attributes of the design, such as performance, assembly, manufacturing, price, distribution, marketing and so on. These constraints demanded collaboration between experts in all of the fields that make up a multidisciplinary team, but with the roles clearly understood, individuals could operate successfully in separate disciplines, as long as they were willing to work together, even though failures were often encountered in companies when communications between discipline based departments were not strong enough. Designers were expected to be fluent with anthropometrics, as that was needed for the design of objects.

## Anthropometrics—the sizes of people

For the design of physical objects

The constraints are complex enough to demand the core skills of design, but the problems are well understood and have been evolving slowly since industrial design emerged as a new discipline in response to the Industrial Revolution. Designers have to understand basic human factors, but it is reasonable to expect that anthropometrics, or the sizes of people, are the most relevant. Thanks to the human factors work at the office of Henry Dreyfuss, anthropometric information for the designer is easy to find, by referring to the book *The Measure of Man*,[5] or the reference cards in *Humanscale*,[6] which present the salient dimensions of people of different statures, gender, age, and ethnic background.

## Physiology—the way the body works

For the design of physical man-machine systems

The next level of complexity comes when you need to consider actions as well as objects. If the design context includes what the person is doing as well as the things that they are using, the constraints need to include the way the human body works, or physiology, as well as the sizes of people. When you are designing a chair for work, you must consider the danger that long periods of sitting may cause back strain, which demands that you understand the structure of the human spine and the muscles that support it; this is not a constraint when you are designing a casual couch or a bar stool. When you are designing a racing bicycle, you need to know about the way the frame can be fitted to the body to yield the maximum power. When you are designing a keyboard for long hours of typing, you need to understand about tactile feedback for the keystrokes, and repetitive stress for the carpal tunnels.

Once we delve into the specifics of an active context like this, the designer may find that the issues are too complicated to understand and act on intuitively; this is when the partnership between designer and a human factors specialist, in these examples a physiologist, becomes essential. The basic complexity of design constraints still demands subconscious synthesis as well as collaboration between everyone in the multidisciplinary team, but a special connection is needed between designer and physiologist, to allow them to be innovative in the human aspect of the solution.

## Cognitive psychology—the way the mind works

For the design of human-computer interactions

Enter the chip! Electronics started with computers, gradually invaded everyday things and places, and are now almost everywhere. This is where we pick up on my stories of designing the laptop and the digital watch, as it is more and more difficult for the designer to understand intuitively about people and what they need and want, as the context is no longer just physical and biomechanical.

When you are concerned about the constraints that will matter to people when you are designing computers and things that are enhanced by electronic behaviors, you need a much more rigorous understanding of the way the mind works. When the design context includes machine intelligence as well as human intelligence, the design team will benefit from the expertise of a cognitive psychologist and will also need designers who are skilled at designing interactions. At this point in the hierarchy, we have arrived at the contents covered in the first five chapters.

### Sociology—the way people relate to each other

FOR THE DESIGN OF CONNECTED SYSTEMS

Connecting everything together caused the next leap in complexity, when the Internet made connectivity a part of many design problems and solutions. Communications technologies like telephones and broadcast media have been with us for long enough to settle down and become familiar, but the sudden explosion of the Internet added the potential of connectivity to objects and services. Sociologists can help members of a design team understand the implications of this and to operate in the more abstract realm of designing services, where you are affected more by relationships among people as well as between users and objects or interfaces. Although services have been around for a long time, the sudden expansion of technology enabled services—and hence service design as a discipline—is still very new. We can see this by the freshness of the ideas expressed in chapter 6, "Services."

Forget your ego, and leave your discipline behind. Let's do this together!

The addition of the expertise of sociologists to a design team is especially important when the nature of the constraints is systemic. When we are designing connected systems of products, services, and spaces, which are used in real time, the brain of any designer who tries to absorb all of the constraints is likely to explode. We are better equipped to face the complexity as an interdisciplinary team, with a collective consciousness, and the ability to create designs as a group or team rather than as individuals.

## Cultural anthropology—the human condition

For global design

Any designer who has developed a product for a global market has had to face the complexities that come from cultural variations. Some people eat with chopsticks and others with cutlery. And colors have strong symbolic meanings that are specific to particular societies. Cultural anthropologists can help people in a development organization understand the nature of cultural differences, which probably will not be intuitively obvious to them without some direct experience of the variations. There are also variations of culture within a single market, as different groups of people have unique anthropological characteristics, based on their occupation, background, or interests.

## Ecology—the interdependence of living things

For sustainable design

At the top of the hierarchy is ecology, where designers need to understand the issues that will affect the environmental condition of our planet as well as the interconnected social and economic systems that we need to sustain. At first thought, sustainable design seems to be in direct opposition to the nature of the consumer society that industrial designers and interaction designers strive to enhance, and is thus a challenging subject for designers to come to grips with. Organizations and processes are emerging that allow the design team to understand and analyze the implications of their designs on sustainability, including the use of materials, energy, and the full lifecycle from "cradle to cradle."[7] This knowledge is still immature, making design for a sustainable planet an intuitive rather than exact science so far. The designer can intuitively synthesize a complex set of requirements, but the right information has to be there to draw on. Sustainability is still at the level of complexity where the science is not yet well established.

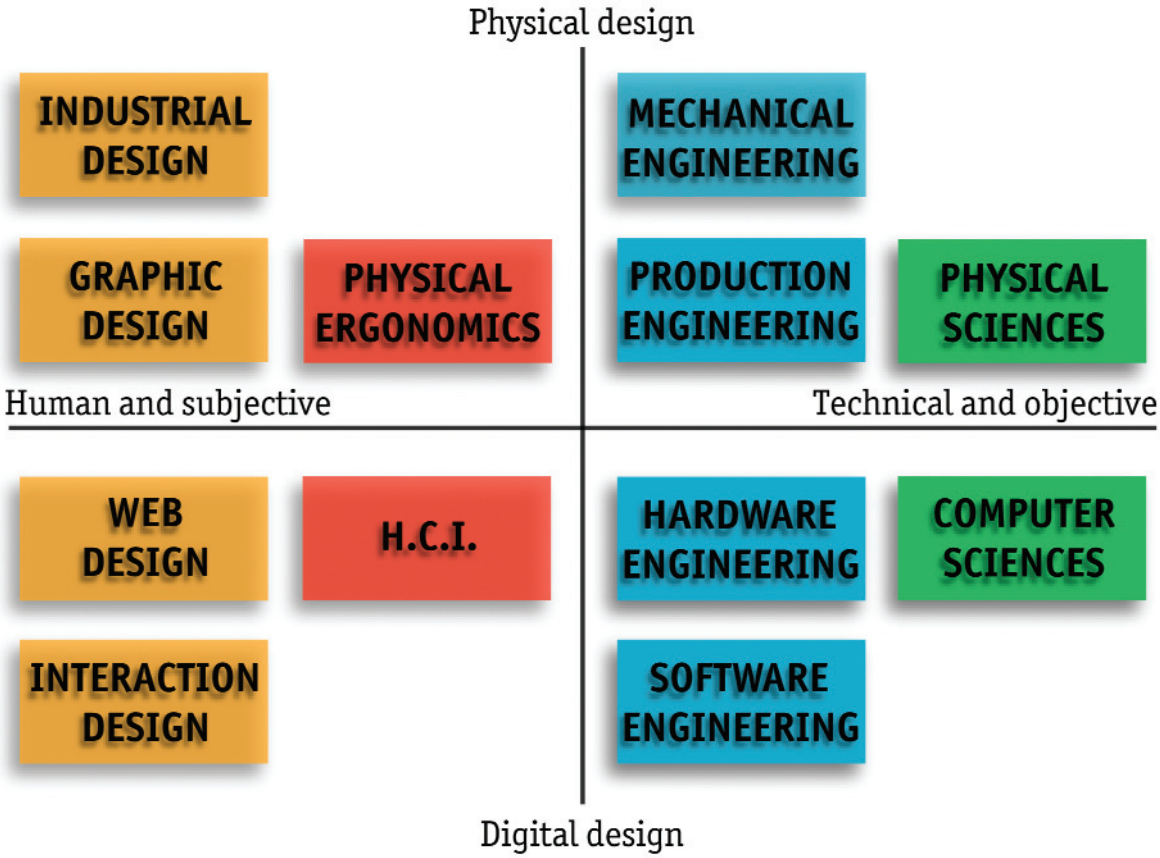
# Why a Design Discipline?

When you go to design a house, you talk to an architect first, not an engineer. Why is this? Because the criteria for what makes a good building fall substantially outside the domain of what engineering deals with. You want the bedrooms where it will be quiet so people can sleep, and you want the dining room to be near the kitchen. The fact that the kitchen and dining room should be proximate to each other emerges from knowing first that the purpose of the kitchen is to prepare food and the dining room to consume it, and second that rooms with related purposes ought to be closely related in space. This is not a fact, nor a technical item of knowledge, but a piece of design wisdom.

Mitch Kapor, "A Software Design Manifesto" (1990)[8]

Mitch Kapor ■

Designers get the love because we control the part that people want. I feel privileged to be a designer, because people like what I do (if I do it well). I also feel a little embarrassed to accept the accolades and the appreciation, as I know that I rely on all the other people who contribute the information that makes the successful synthesis possible. Mitch Kapor is right that designing a dream house is the output of design wisdom rather than technical knowledge. You enjoy your favorite article of clothing because it seems to be designed just for you, and it makes you feel good to wear it. I have a relationship with my favorite objects based on their aesthetic qualities more than their utility or price/performance ratio.




*The diagram shows four quadrants, with the horizontal axis dividing human and subjective qualities from those that are technical and objective, and the vertical axis separating physical design contexts from those in the digital realm. We can position the development disciplines in four columns: the design disciplines, human sciences, engineering disciplines, and technical sciences. We see the need for interaction design as a discipline that can create solutions with human and subjective qualities in a digital context.*

Designers rely on all the other disciplines, in that everything else has to work before design has a chance. First you have to be able to afford it, whether it is a house, a car, or a piece of software. Then it must perform: the building withstands the wind and rain; the car takes you where you want to go; and the software is stable. It must be useful, sheltering us, transporting us, and achieving the goals of the program. It must be usable, with stairs of the right tread sizes, steering wheel of a familiar size, and software that we can understand. Only when all of these attributes are already satisfied, after much effort from all of the contributors from the various disciplines, will the question of delight become important. Design wisdom has the power to please, but only in a context where the demands of all of the constraints are obeyed.

It can all go horribly wrong, of course, because the risks of living in the subconscious are so high. Operating at a subjective level, it is difficult to tell whether we are synthesizing the right set of constraints or whether the information is accurate. Your architect may be so interested in the magnificent material for the kitchen counter that the wisdom of space is lost, a difficulty that makes it risky to commission the design of your dream house, rather than buying one that already exists. At least with a car you can take it for a test drive as well as read the consumer reports. Interaction design is so new that there is very little established wisdom so far, and the chances are still low that you will be able to find a solution that satisfies all of those functional constraints, let alone give you aesthetic pleasure.

## Where Does Interaction Design Fit?

A NARROW DEFINITION of interaction design is: "The design of the subjective and qualitative aspects of everything that is both digital and interactive, creating designs that are useful, desirable, and accessible." The designer is working in the artificial context of bits, pixels, input devices, users' conceptual models, and organizing metaphors. This is the version of interaction design that I

practice—that I needed to learn in order to move up the hierarchy of complexity. This is a narrow version of interaction design, related to the experience and background of other design disciplines that deal in aesthetics and qualitative values, like industrial design, graphic design, and architecture. It is the equivalent of these disciplines in that the first concern of the designer is the values of the people who will use the design—the aesthetics, subjective and qualitative values, and human factors. The designer creates a solution to give pleasure and lasting satisfaction, and hence to fit the market, business, and social requirements.

There is another, much broader, view of interaction design: "The design of everything that is both digital and interactive." It includes the design of all the interactions that are enabled by digital technology, whether by computers, chips embedded in products or environments, services, or the Internet. This broad view of interaction design includes the work of human-computer interaction (HCI) professionals, computer scientists, software engineers, cognitive psychologists, sociologists, cultural anthropologists, and designers. It includes everyone who has the knowledge and tools that allow them to "create or contrive for a particular purpose or effect"[9] in this digital context, sometimes as an individual, but usually as part of an interdisciplinary team.

A narrow definition of interaction design is: "The design of the subjective and qualitative aspects of everything that is both digital and interactive."

A broad definition of interaction design is: 'The design of everything that is both digital and interactive."

It is natural for people outside the design and development disciplines to see this broad view, as they react to the resulting designs in terms of the experiences they have as users of interactive software, devices, and services. If they think about the design at all, they are likely to see the whole result, as they don't understand the individual roles of particular disciplines. This book is structured around this inclusive vision of what interaction design can be and has presented the thinking of many experts in the field from very varied backgrounds, chosen for their contributions rather than their closeness to my personal view of the discipline.

# Is Interaction Design Here to Stay?

The decades ahead will be a period of comprehending biotech, mastering nature, and realizing extraterrestrial travel, with DNA computers, microrobots, and nanotechnologies the main characters on the technological stage. Computers as we know them today will (a) be boring, and (b) disappear into things that are first and foremost something else: smart nails, self-cleaning shirts, driverless cars, therapeutic Barbie dolls, intelligent doorknobs that let the Federal Express man in and Fido out, but not 10 other dogs back in. Computers will be a sweeping yet invisible part of our everyday lives: We'll live in them, wear them, even eat them. . . . Yes, we are now in a digital age, to whatever degree our culture, infrastructure, and economy (in that order) allow us.

Nicholas Negroponte, MIT Media Lab, 1998[10]

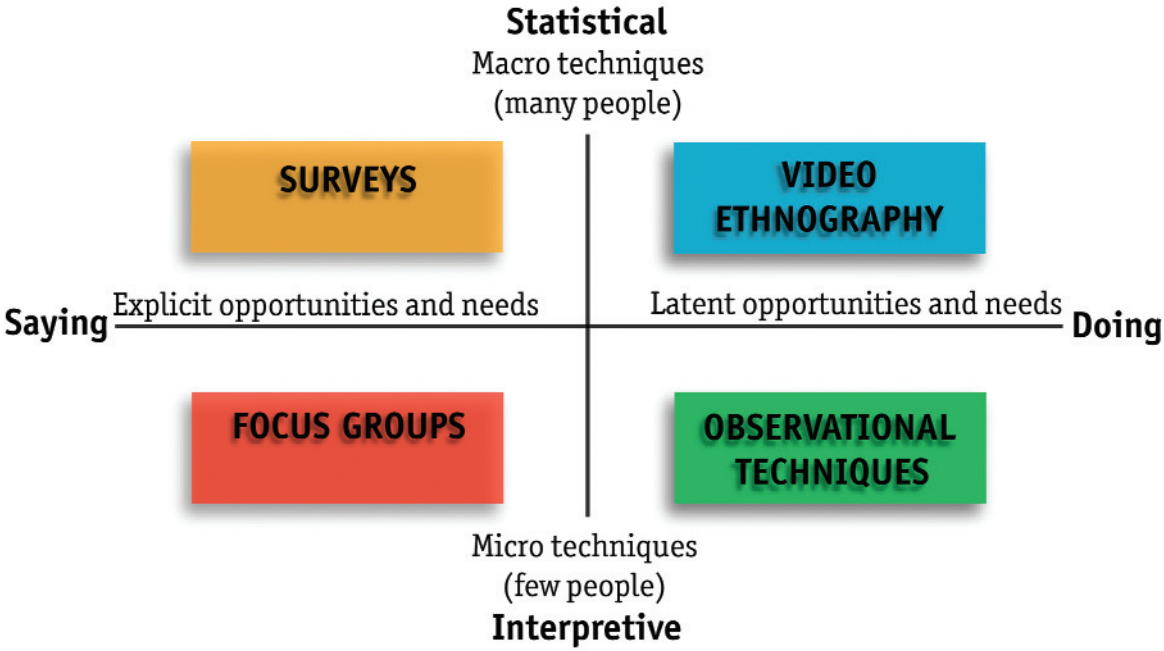
Nicholas Negroponte ■

We seem to be well on the way toward fulfilling these predictions that Nicholas Negroponte describes with such colorful images. Even if you doubt that we are already in a digital age, it is clear that we are marching relentlessly toward a condition where everything that can be digital will be digital.[11] What does this mean for interaction design?

In June of 2002 I was in London at the time of the display of work of the graduating master's students at the Royal College of Art, and I was looking at the projects from the interaction design department. I was impressed by the fact that most of them were both digital and physical; the students were designing smart

objects rather than computer-based software. I was moving from the work of one student to the next, looking in some detail at the individual designs. Suddenly I looked up at the whole room, and discovered to my surprise that I had drifted into the area occupied by the projects from the industrial design department, never noticing a difference in the nature of the work. Just as the interaction designers were designing smart objects, the industrial designers were designing objects that were smart, finding it natural to include electronically enabled behaviors. It made me wonder if this was evidence of the beginning of the end of interaction design as a separate discipline.

Practitioners in the technical design disciplines adopt new technologies earlier than their counterparts in the human disciplines, as is explained by David Liddle in chapter 4, "Adopting Technology." This would lead one to expect that a similar migration might have already happened in engineering. Computer science emerged first and gave rise to new disciplines for the design of hardware and software. Eventually every engineer expected to use electronics and software in the natural course of development, so engineering education included learning about circuits and programming languages. However, this did not mean that the new disciplines of hardware and software design merged back into the traditional engineering design disciplines, but rather that all aspects of engineering design make use of technology, and all engineering designers can operate to some extent in the digital realm. It seems likely that a parallel to this will exist in the human disciplines, with all designers thinking it natural to include digital solutions as aspects of their designs, accepting the constraints and opportunities offered by new technologies. At the same time there will continue to be interaction designers who have a more in-depth knowledge and expertise about designing interactions and remain the experts in the field. I think interaction design is here to stay.




*The diagram lays out different kinds of research methods, showing a horizontal scale that characterizes design opportunities and user needs, from explicit (left) to latent (right). The vertical scale indicates the difference in techniques from macro (top) to micro (bottom). Traditionally, market research was developed to find out what people want by asking them directly through large-scale surveys or more in-depth focus groups; these methods work very well to find out what people say they want.*

*If your goal is innovative design, your product or service has not even been thought of, so by definition it cannot be explained to research participants. This is where methods are needed to discover latent needs and desires that will help the members of the design team define potential opportunities. The examples on the diagram are video ethnography techniques on the macro scale, where stop frame video is set up to watch a space or task to reveal patterns of use. On the micro scale, the example is observational techniques, where members of the design team go to wherever the design context exists to see what people really do, as opposed to what they say they do.*

# People

It is essential to the success of interaction design that designers find a way to understand the perceptions, circumstances, habits, needs, and desires of the ultimate users.

Jane Fulton Suri, 2005

## Latent Needs and Desires

In the mid eighties I was struggling to come to grips with what interaction design could and should be and how we could learn to bring our expertise in subjective and qualitative values to the realm of electronics. My first principle in design is to think first about the people part of the design: Who are the users? What do they want from the experience? What will give them satisfaction and enjoyment? In the seventies I had developed a tradition of going to look at what really happens in the context of each design problem. When I or one of the other designers in my practice was designing a marine radio for fishing boats, we went out on the boats to see how they used existing marine radios and talked to them about what was important for them. When we were designing a device used in hip surgery, we put on the green gowns and masks, went into the actual surgery, and watched what happened to try to think of ways that a new design could improve the situation.

"Observation" was the label we used for the best way to learn about people in the context of a particular design problem, implying that you needed to look at what people really do in a situation, rather than rely on the conventional technique of asking them about what they think and do. When you are trying to

understand the latent needs and desires of potential users before a design is created, it is important to learn about their existing habits and context of use—things they are rarely able to tell you about explicitly. You will gather clearer and more vivid knowledge of these by experiencing them firsthand.

Over the years, the human factors people at IDEO have evolved many new techniques beyond the simple observations that we started with, and now we have amassed a set of fifty-one methods, published as a deck of cards.

## 51 Ways of Learning about People

I GOT TO KNOW Jane Fulton Suri[12] in 1986 when she came to San Francisco to study for a year at UC Berkeley. This was a sabbatical for her, after a decade of human factors research and consulting in Britain, where she worked with a group that specialized in consumer product safety. She was interested in being more actively involved in design:

■ Jane Fulton Suri

*Photo*
Skylar Reeves

> I'm always too late! I want to do some good in the world, but I'm only getting to influence bad designs after the damage is already done. For example, I've learned a lot about a lot of people who have had toes cut off by rotary lawn mowers. I would much rather have been working with the people who designed the mower, so I could have helped to make it safer in the first place, and saved the toes.

I thought this was a chance to integrate human factors expertise into my specialist team of designers, so I asked Jane if she would be willing to join the group in San Francisco when her course of study was complete. She accepted and started to work with the designers. At first they simply asked for her reaction to their ideas, but they gradually grew to appreciate the thoughtfulness and value of her contributions and started to ask her to help right at the beginning of each project and be involved throughout. By 1991 her contribution was clamored for by all of us, so we made a commitment to include a contribution from human factors specialists on every project, and expanded the


IDEO METHOD CARDS
IDEO METHOD CARDS
IDEO METHOD CARDS

human factors staff. As time went by, she and her human factors colleagues drew upon a vast range of methods for understanding people and their experiences. They soon evolved a substantial portfolio of tools and techniques. When the number of methods was approaching fifty, one of the team, Maura Shea, suggested that they represent them as a deck of cards.

The idea of the methods cards[13] is to make a large number of different techniques accessible to all members of a design team and to encourage a creative approach to the search for information and insights as their projects evolve. The intention is to provide a tool that can be used flexibly to sort, browse, search, spread out, or pin up. I find myself using the cards after a typical project briefing meeting, working my way through the pack as if in a game of patience and selecting the most useful set for that particular project in its various phases. When I meet with the team, I deal the set I've chosen and talk about why they might be useful in this context.

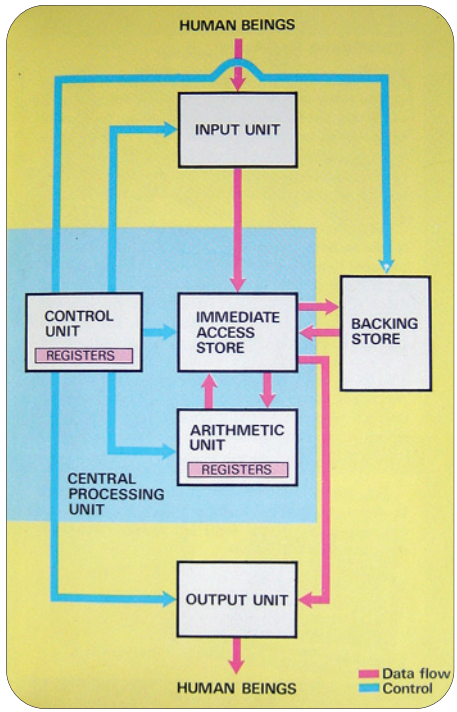
Each of the fifty-one cards contains explanatory text about how and when the method can be used and a brief example of its application to a real design project, with an illustrative and sometimes whimsical image on the other side. The cards are divided into four categories, ranging from the objective to more subjective—Learn, Look, Ask, and Try: "Learn" from the facts you gather, "Look" at what users do, "Ask" them to help, and "Try" it for yourself.

■ IDEO methods cards

*Photo*
Mark Serr

It is generally most valuable to apply, or sometimes modify, a range of different methods for any given project. The most useful set will depend on whether the purpose is primarily a *generative* one of defining design opportunities for particular kinds of users or a domain of activity, or an *evaluative* one of refining specific design ideas as they develop. In an evolutionary project, where the new design will be closely related to something that exists, techniques that yield explicit information about a particular product and usage may be valuable. If the project is revolutionary—the design will set a new precedent—methods that help the designer understand a broader domain of activity and related latent needs may be more appropriate.

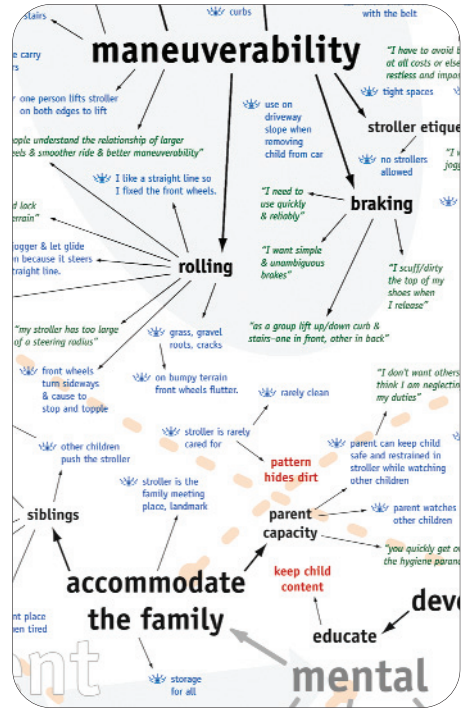
Here are four examples of the methods from each category:




FLOW ANALYSIS


COGNITIVE TASK ANALYSIS

HISTORICAL ANALYSIS




AFFINITY DIAGRAMS

# Learn

ANALYZE THE INFORMATION you've collected to identify patterns and insights.

### FLOW ANALYSIS

How — Represent the flow of information or activity through all phases of a system or process.

Why — This is useful for identifying bottlenecks and opportunities for functional alternatives.

Example — *Designing an online advice Web site, flow analysis helped the team to gain a clearer sense of how to make it easy to find your way around the site.*

### COGNITIVE TASK ANALYSIS

How — List and summarize all of a user's sensory inputs, decision points, and actions.

Why — This is good for understanding users' perceptual, attentional, and informational needs and for identifying bottlenecks where errors may occur.

Example — *Logging the commands that would be involved in controlling a remotely operated camera helped the team establish priorities among them.*

### HISTORICAL ANALYSIS

How — Compare features of an industry, organization, group, market segment or practice through various stages of development.

Why — This method helps to identify trends and cycles of product use and customer behavior and to project those patterns into the future.

Example — *A historical view of chair design helped to define a common language and reference points for the team members from the client and consultancy.*

### AFFINITY DIAGRAMS

How — Cluster design elements according to intuitive relationships, such as similarity, dependence, proximity, and so forth.

Why — This method is a useful way to identify connections among issues and to reveal opportunities for innovation.

Example — *This affinity diagram shows what's involved in transporting young children, and helps to identify the opportunities to improve the design of a stroller.*

FLY ON THE WALL

A DAY IN THE LIFE

SHADOWING


PERSONAL INVENTORY

# Look

OBSERVE PEOPLE TO discover what they really do—not what they say they do.

### FLY ON THE WALL

How — Observe and record behavior within its context, without interfering with people's activities.

Why — It is useful to see what people do in real contexts and time frames, rather than accept what they say they did after the fact.

Example — *By spending time in the operating room, the designers were able to observe and understand the information that the surgical team needed.*

### A DAY IN THE LIFE

How — Catalog the activities and contexts that users experience for an entire day.

Why — This is a useful way to reveal unanticipated issues inherent in the routines and circumstances people experience daily.

Example — *For the design of a portable communication device, the design team followed people throughout the day, observing moments at which they would like to be able to access information.*

### SHADOWING

How — Tag along with people to observe and understand their day-to-day routines, interactions, and contexts.

Why — This is a valuable way to reveal design opportunities and show how a product might affect or complement user's behavior.

Example — *The team accompanied truckers on their routes in order to understand how they might be affected by a device capable of detecting drowsiness.*

### PERSONAL INVENTORY

How — Document the things that people identify as important to them as a way of cataloging evidence of their lifestyles.

Why — This method is useful for revealing people's activities, perceptions, and values as well as patterns among them.

Example — *For a project to design a handheld electronic device, people were asked* to show *the contents of their purses and briefcases and explain how they use the objects that they carry around everyday.*


CONCEPTUAL LANDSCAPE


COLLAGE


FOREIGN CORRESPONDENTS

CARD SORT

# Ask

ENLIST PEOPLE'S PARTICIPATION to elicit information relevant to your project.

### CONCEPTUAL LANDSCAPE

How — Ask people to diagram, sketch, or map the aspects of abstract social and behavioral constructs or phenomena.

Why — This is a helpful way to understand people's mental models of the issues related to the design problem.

Example — *Designing an online university, the team illustrated the different motivations, activities, and values that prompt people to go back to school.*

### COLLAGE

How — Ask participants to build a collage from a provided collection of images and to explain the significance of the images and arrangements they choose.

Why — This illustrates participants' understanding and perceptions of issues and helps them verbalize complex or unimagined themes.

Example — *Participants were asked to create a collage around the theme of sustainability to help the team understand how new technologies might be applied to better support people's perceptions.*

### FOREIGN CORRESPONDENTS

How — Request input from coworkers and contacts in other countries and conduct a cross-cultural study to derive basic international design principles.

Why — This is a good way to illustrate the varied cultural and environmental contexts in which the products are used.

Example — *A global survey about personal privacy helped to quickly compile images and anecdotes from the experiences of the correspondents.*

### CARD SORT

How — On separate cards, name possible features, functions, or design attributes. Ask people to organize the cards spatially, in ways that make sense to them.

Why — This helps to expose people's mental models of a device or system. Their organization reveals expectations and priorities about the intended functions.

Example — *In a project to design a new digital phone service, a card-sorting exercise enabled potential users to influence the final menu structure and naming.*


EMPATHY TOOLS


SCENARIOS


THE WALL STREET JOURNAL.

Page B4.

Page B5.

MARKETPLACE

Reveille for Couch Potato

AOL Rolls Out Interactive TV With Time Warner Links; Rivals Complain of Bias

By Jared Sandberg

WALL STREET JOURN

© 2000 Dow Jones & Company, Inc. All Rights Reserved.

FRIDAY, NOVEMBER 3, 2002

INSIDE

's News—

World-Wide

BUSH AND GORE FISHED a shrinking pool of undecided voters in a swing-state dash.

The Republican sprinted toward the campaign's final weekend holding a national lead that has withstood a barrage of Democratic attacks. In terms of assembling an Electoral College majority, however, Gore still has reason for optimism. Meanwhile, Bush took a late blow as a 1976 drunken-driving conviction surfaced. Clinton campaigned in California, where a Gore lead has slipped, saying a vote for the vice president is "the next best thing" to a vote for himself. (Articles in Column 1 and on Pages A8 and A22)

Delaware's Sen. Roth, chairman of the Finance Committee, is in the re-election fight of his career. The Republican's age, 29, has become an issue in the race.

WEEKEND JOURNAL.

When Food Goes Postal

Gourmet goodies are a click away in the $1.1-billion-a-year online grocery business. But how good is the stuff that shows up on your doorstep?

To DVD—or Not?

It's this year's hot Christmas gift. We see if a bare-bones DVD player is as good as ones costing hundreds more.

Sleeping Your Way to Europe

With business-class seats that let you actually lie down and sleep, who needs the Concorde? Our Finicky Traveler's report.

Spoiler Fears Open A Generation Gap

Washing

A Special Wee
The Wall S
Capit

NEXT YEAR'S HEADLINES

INFORMANCE

# Try

Create simulations and prototypes to help empathize with people and to evaluate proposed designs.

### EMPATHY TOOLS

How — Use tools like clouded glasses and weighted gloves to experience processes as though you yourself have the abilities of different users.

Why — This is an easy way to prompt an empathic understanding for users with disabilities or special conditions.

Example — *Designers wore gloves to help them evaluate the suitability of cords and buttons for a home health monitor designed for people with reduced dexterity and tactile sensation.*

### SCENARIOS

How — Illustrate a character-rich storyline describing the context of use for a product or service.

Why — This process helps to communicate and test the essence of a design idea within its probable context of use. It is especially useful for the evaluation of service concepts.

Example — *Designing a community Web site, the team drew up scenarios to highlight the ways particular design ideas served different user needs.*

### NEXT YEAR'S HEADLINES

How — Invite employees to project their company into the future, identifying how they want to develop and sustain customer relations.

Why — Based on customer-focused research, these predictions can help to define which design issues to pursue for development.

Example — *While designing an Intranet site for information technologists, the team prompted the client to define and clarify their business targets for immediate and future launches.*

### INFORMANCE

How — Act out an "informative performance" scenario by role-playing insights or behaviors that you have witnessed or researched.

Why — This is a good way to communicate an insight and build a shared understanding of a concept and its implications.

Example — *A performance about a story of mobile communications shows the distress of a frustrated user.*

An article by Daniel Pink in *Fast Company*[14] captures one way the cards can be used at the outset of projects.

> Fast Company decided to give IDEO's Method Cards a workout. In a conference room at the company's Palo Alto headquarters, we presented an IDEO team with two scenarios to see how they would begin wrapping their minds around a design problem. We weren't looking for an end. We were looking for a beginning—the initial steps that would set the course of the eventual design. Here's what happened when IDEO let the cards out of the box.
>
> *First deal: A carmaker, recognizing that people are living longer and better, wants to develop a car that appeals uniquely to drivers over 65 years old. How can the carmaker better understand the concerns of this group of prospective customers?*
>
> Five IDEO staffers—Jane Fulton Suri, David Gilmore, Kristine Chan Lizardo, Annetta Papadopoulos, and Aaron Sklar—listen as I read the scenario aloud. Then they open their boxes and begin sorting and shuffling the cards. Some they toss aside. Others they lay faceup in front of them. Our first-floor conference room is flanked by a wall-sized window that looks out on a sidewalk. To the pedestrians passing by, it looks as if we're playing pinochle.
>
> Gilmore, a British expat who once designed coins for the Royal Mint, holds up a card from the Ask suit. It's called Unfocus Group. To grasp the underlying design issues, Gilmore would assemble a diverse collection of people to talk about cars. He'd include healthy and active senior citizens, seniors with health problems, seniors who love cars, and seniors who don't. Fulton Suri, another Brit transplanted to the West Coast, chimes in: Why not also include a driving instructor and a state trooper for their perspectives? "And maybe they can help build something," she adds. She fingers the Experience Prototype card from the Try suit. Perhaps the grandmas and the smokeys could suggest a prototype car feature that IDEO could quickly construct and let them test. Fulton Suri also selects Empathy Tools. To simulate what it's like to have limited mobility and dexterity while driving, IDEO designers could don clouded glasses, slip on heavy gloves, or bandage their legs before taking a test-drive. "Of course, not everybody over 65 has those problems," she says. But the carmaker could end up introducing some new features for one age group that everyone might value because of the simplicity and elegance of the design.

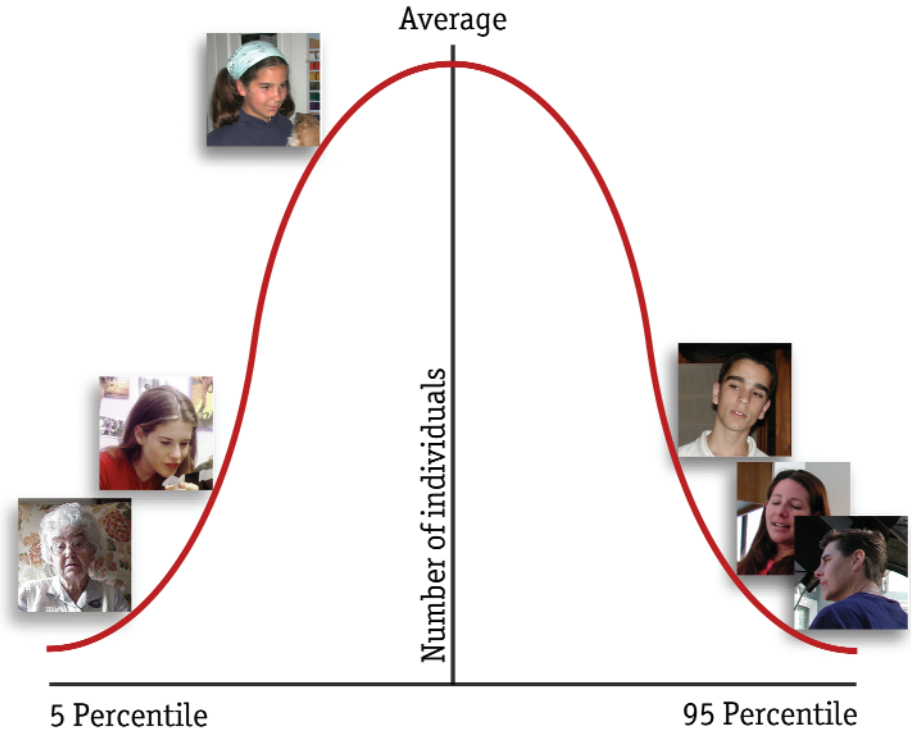
Gilmore emphasizes the Emotional Dimension card. Cars have "life trajectories," he says. Like furniture and certain pieces of clothing, they carry memories of a particular stage of a person's life. So he'd have seniors craft a personal history of the cars they've owned and what those vehicles have meant to them. Buying your first car is a rite of passage. But, Gilmore wonders, what does it feel like to buy what could be your last car?

*Second deal: A national television network seeks to reinvent its struggling nightly newscast and to update a format that has been largely untouched for a generation. What are some ways to uncover new approaches to the nightly news?*

Lizardo starts things off by shouting, "A Day in the Life!" A card from the Look suit, it asks the potential users to document everything they do in a given day. The goal is to discover how people actually spend their time—and how that affects when, where, and whether they watch the news. Fulton Suri, eyeing the four cards fanned out in her left hand as if she were playing poker, sees and raises Lizardo. She suggests pairing her approach with another card: Behavioral Sampling. IDEO would give subjects pagers and then contact them randomly throughout the day to ask what news and information is available to them at that moment and what they've encountered in the past five minutes. Surveys and focus groups don't yield this sort of texture nor do they set the problem in context. And in this room, as elsewhere at the firm, context is king.

So is serious engineering. Two of the six people in this room are mechanical engineers, each with four patents to her name. One is Lizardo. The other is Papadopoulos, who offers the Foreign Correspondents card. She would enlist IDEO staff in different countries to watch the nightly news where they are and contribute their observations. Along those lines, Sklar wants to broaden the inquiry by using Extreme User Interviews, a card from the Ask suit. He'd try to understand the center by interviewing those who occupy the edges: "someone who doesn't have a TV, someone who gets all their news from the National Enquirer, someone who watches TV constantly."

Minds click. Ideas fly. How about Affinity Diagrams? How about Word-Concept Association? Says Fulton Suri: "Just the fact that I've got them in my hands is making my brain think about all sorts of different approaches." A breakthrough, it seems, is in the cards.




### Remember the extremes

*People vary in many characteristics that might be relevant to our design: interests, experience, learning pace, lifestyles, wealth, work styles, living situations, and so on. We maximize our chances of success by considering the full range of people we are designing for.*

*Many such attributes are distributed according to this kind of bell curve, where a relatively large number of people cluster around its average value, with numbers tailing off gradually to a few people represented at each of its extremes. When we want to learn about people, it is important to include some who represent critical extreme values of the relevant characteristics, to avoid the trap of designing only for the average.*

*A very simple example—if we design an ATM interaction to time out after the average time people take to input their PIN number, we will inconvenience the full 50 percent of users who take longer than average! Far better to design the system to accommodate all but the slowest 2 percent.*

The kinds of techniques exemplified by the methods cards can be used to counteract our natural self-centeredness and better inform our intuition. It is very difficult to be a good designer without having a big ego. After all, we need to believe enough in our own ability to synthesize the right solution intuitively, while at the same time admitting that we may not readily have the most lucid explanation of the rationale. That may be why we find it much easier to design for ourselves than for other people. Good friends like Jane Fulton Suri, who offer all those methods for understanding users, their habits, and contexts, are invaluable for keeping us focused on other people, even inspired by our insights about them, so we don't slide back into the world we know and understand.

It is relatively easy when we are ourselves typical of the intended users. There are lots of examples of successful products that have been designed based only on the intuitive judgment of the designers thinking about themselves. The computer industry emerged in this way. Enthusiastic early adopters enabled the early phases, and they were thrilled to use the interactions designed by engineers for themselves. For twenty-five years Hewlett Packard was a company of engineers who worked in labs, designing equipment for engineers who work in labs. It was much more difficult for them when they started developing general purpose computers for a broader consumer market and needed to design interactions that would be easy to use for ordinary people. Nike attracts designers who are interested in athletic activities, so it becomes natural for them to design footwear for themselves.

Designers are often self-absorbed in this way. If you are sitting at a computer, working to complete a deadline for a project, it is easy to stay where you are, to design a solution to satisfy your own needs and aspirations, one that you find amusing or engaging and that fits your own idea of beauty. If you stay at your desk, your design is unlikely to represent the full range of people who will use the result of your work, but if you leave your desk and bravely open yourself and your design ideas to influence by potential users and usage contexts, you will produce work that more successfully reflects the needs and desires of a diverse set of people.


Gossamer Albatross
DUPONT

# Prototypes

What I hear, I forget.
What I see, I remember.
What I do, I understand.

Lao Tse, Chinese philosopher, b. 604 BC

Language is convincing.
Seeing is believing.
Touching is reality.

Alan Kay

## A New Prototype Every Day

- *Top*
  Gossamer Condor, 1977
- *Bottom*
  Gossamer Albatross, 1979

Paul MacCready is well known as a designer of innovative air and land vehicles, but it was the creation of the Gossamer Condor[15] that made him famous. In 1959 a British industrialist named Henry Kremer offered £50,000 for the first human-powered aircraft that could demonstrate the same degree of aerodynamic control as the early Wright fliers, by tracing a figure eight around two markers four-fifths of a kilometer apart. After frustrated attempts by various enthusiasts, in 1977 the Gossamer Condor was the first craft to succeed, pedaled by the cyclist Brian L. Allen. Paul MacCready led the design team, adopting a rapid prototyping approach to the development. The plane was built of a delicate skeleton of thin tubes and tensioning wires, covered by transparent plastic sheeting that was taped in place. This proved a very flexible kit of materials, allowing the design to be changed as often as once a day, and a new prototype to be built. This rapid prototyping technique was so effective that the craft was an easy winner in the competitive tests, and a later generation of the design went on to be flown by pedal power for great distances, including a crossing between England and France.

Larry Tesler told a similar story of creating new prototypes every day in chapter 2, "My PC." He and Bill Atkinson were designing the pull-down menu structure for the Apple Lisa, working around the clock. Bill would work nights and Larry would work days. During the night Bill would make prototypes of user interface concepts, written in a robust enough code to support some form of testing. Then Larry would run user tests during the day. Larry would give Bill a report at the end of the day and tell him what he had learned from the tests. Then they would brainstorm for ideas and decide what to try next, so that Bill could go off and spend the night programming. In the morning he would bring in a new prototype and then go home to bed. They used this method for several intense weeks until the specification was solid, and it was during this time that they designed the arrangement of pull-down menus across the top of the screen that is so familiar today.

In both these examples, the designers achieved dramatic breakthroughs by generating their ideas every day and advancing them by a combination of prototyping and user evaluation. In Paul MacCready's case, the ingenuity was in choosing materials that would allow daily changes in the design just by retaping the plastic or repositioning the structure of tubes and wires, so that the pedal power provider could try out another design. Bill and Larry generated new ideas by brainstorming every evening after the user tests of the day, relying on their experience and ingenuity to come up with the ideas for the next step; then Bill built a prototype, designing as he went along. Some nights may have been futile, but the tests on the following day showed if it was time to say goodbye to an idea. Other nights were spectacular breakthroughs, as Larry said in his interview[16] when describing the creation of the pull-down menu structure for Lisa. The user tests the following day proved the point.

There may not be many design problems where the development of a new prototype every day is possible, or even wise, but the surprising lesson that we can take away from these stories is the advantage of prototyping early and often, trying things out as quickly as possible and more often that we think in advance that we would dare. The risk of failure puts up a barrier

to trying the first ideas, as we know that they are almost bound to be bad or wrong in some way. If we can shrug off the fear of that risk and get into the habit of trying things out as soon as we can, we will fail frequently, but the reward is that we will succeed sooner.

## Interaction Design Prototypes

**Pro-to-type** *n.* 1. An original type, form, or instance that serves as a model on which later stages are based or judged.

American Heritage Dictionary[17]

The definition of a prototype in the *American Heritage Dictionary* seems particularly apt as a description of a design prototype in its broad inclusion of "type, form, or instance," and also in the inference of a cyclical process where later stages follow before judgment is passed. In the context of designing interactions, perhaps we can narrow the definition to "A representation of a design, made before the final solution exists."

This is still a very broad and includes a lot of possible types. When the design process is nearing completion, the prototype will need to represent a combination of the way the design looks, feels, behaves, and works. If the context includes the experience as well as the product, we will also want a representation for understanding, exploring, or communicating what it might be like to engage with the design. The "experience prototype" for Kodak, described by Jane Fulton Suri and Mat Hunter in chapter 4, "Adopting Technology," was the right approach to integrate and communicate the interaction architecture concept, but the team had worked on the problem for long enough to know exactly what to prototype. The uncertainties were already resolved by the research efforts and iterations that had gone before. Jane had articulated the framework that summed up the design opportunities for consumer digital photography. Mat had

developed the key design ideas of modes, filmstrip, and so on; thus the design ideas were already mature. The design team had already developed and discarded many prototypes before they were ready for the version that they described.

In a paper called "Experience Prototyping,"[18] Marion Buchenau and Jane Fulton Suri divide prototyping techniques into three categories: those that help you to understand existing user experiences and context, those that help you to explore and evaluate design ideas, and those that help you to communicate ideas to an audience. The paper starts by introducing the background to the thinking:

## Introduction

> Increasingly, as designers of interactive systems (spaces, processes and products for people), we find ourselves stretching the limits of prototyping tools to explore and communicate what it will be like to interact with the things we design.
>
> "Prototypes" are representations of a design made before final artifacts exist. They are created to inform both design process and design decisions. They range from sketches and different kinds of models at various levels—"looks like," "behaves like," "works like"—to explore and communicate propositions about the design and its context.
>
> As such, prototyping is a key activity within the design of interactive systems. Several groups of designers and researchers, perhaps most notably at Apple Computer, Xerox PARC, and Interval Research, have been active both in pushing the boundaries of prototyping beyond the range of traditional methods[19] and in developing understanding of the value of different forms of prototype. For example, Houde and Hill[20] discuss various functions for prototypes as being essentially about the "role" an artifact will play, its "look and feel" and how it will be implemented. Other work has explored issues such as different levels of fidelity,[21] prototypes for different audiences[22] and models or use in the context of participatory design.[23] Further, prototyping as a design practice is now promoted within the business community as a key element in innovation.[24] Building from this foundation, designers at IDEO are working to expand internal prototyping practices to embody the concept of

■ Marion Buchenau

"experience prototyping" as an integrated part of the design process. In this paper we will discuss what we mean by experience prototyping, why we think it is important and then look at its application within three key design activities—understanding, exploring and communicating—through examples from design projects.

## What is "experience prototyping"?

First, let's think for a moment about what we mean by "experience." With respect to prototyping, our understanding of "experience" is close to what Houde and Hill call the "look and feel" of a product or system, that is "the concrete sensory experience of using an artifact—what the user looks at, feels and hears while using it." But experience goes beyond the "concrete sensory." Inevitably we find ourselves asking questions about the "role" which Houde and Hill define as "the functions that an artifact serves in a user's life—the way in which it is useful to them." And even more than this, when we consider experience we must be aware of the important influences of contextual factors, such as social circumstances, time pressures, environmental conditions, etc.

By the term "experience prototype" we mean to emphasize the experiential aspect of whatever representations are needed to successfully (re)live or convey an experience with a product, space or system. So, for an operational definition we can say an Experience Prototype is any kind of representation, in any medium, that is designed to understand, explore or communicate what it might be like to engage with the product, space or system we are designing.

Here are some quotes from the paper that describe the three categories, supported by an example of each taken from the paper, and others from the previous chapters. The first category is about understanding existing user experiences when you are collecting constraints and trying to understand everything that will make a difference to the result.


spyfish
front
50
m
240°
W

## Understanding existing user experiences and context

Experience prototyping here is applied to demonstrate context and to identify issues and design opportunities. One way to explore this is through direct experience of systems—the prototyping goal is to achieve a high fidelity simulation of something that exists, which can't be experienced directly because it is unsafe, unavailable, too expensive, etc.

The questions to ask in this stage are: What are the contextual, physical, temporal, sensory, social and cognitive factors we must consider as we embark on design? What is the essence of the existing user experience? What are essential factors that our design should preserve?

An example from the paper is for the design of the control interactions for a remotely operated underwater camera:

## The ROV Pilot experience

This example used a proxy device to provide the team with specific insight into an experience that was not readily available to them.

The project involved the design of a pilot's interface for an underwater remotely operated vehicle (ROV) and its cameras. It was important that designers grasp and deal with some of the cognitive confusion that would arise for the operator. There would be problems for operators steering a tethered vehicle with six degrees of freedom, as well as multiple cameras—which can be positioned independently from the ROV itself—while trying to find a target in a vast undifferentiated space with limited visibility.

■ H2Eye Spyfish underwater camera ROV, 2002

*Photo*
Jason Tozer

In the initial project phase, the design team created a task analysis, based on interviews with pilots and literature research which was useful to them, but did not communicate the realities of ROV operation very effectively. For the first experience prototyping exercise, one of the designers used a rolled-up sheet of paper to limit her peripheral view while searching for a target—a Post-it note in her work space.

To get to the more problematic cognitive and functional issues, the team developed a game in which one player, A, stood in a room which was empty except for multiple chairs (portraying underwater obstacles), and one of them held a chocolate bar, the target. Player A held a video camera connected by a long cable to a remote TV screen

where the live picture was viewed by player B. Player B gave verbal instructions to player A to move right/left, forward/back, and up/down and gave separate verbal commands to direct the camera.

After a few yards of cables wrapped round A's legs and the chairs as well as B's frustration at making mistakes, "Aargh! I meant camera right not move right," the design team and the client had personal insight about many important issues. For example, it was obvious that a critical need was clear feedback to support a mental picture of the vehicle's path through space, feedback about the tether condition, and the need for a clear distinction between controls for the vehicle and for the camera.

As a follow-up, the team asked a participating retired ROV pilot about the validity of the simulated experience which, to his surprise, portrayed a quite accurate picture. He provided additional information, mainly about contextual factors (e.g., different levels of experience, underwater conditions, support tools like maps) which might change or influence the portrayed experience. The ability to share this experience prototype provided verification and enrichment of the simulated experience with a real life event. This further enhanced the participants' understanding of the pilot's problems and created a shared reference point between all members of the design team as the work moved forward.

The success of the game with two players was dramatic because each of the team members who tried it experienced a prototype of the difficulties that formed the crux of the interaction design problem. It was also ingeniously quick and inexpensive to do, with the whole team going through the exercise in a single afternoon, but notice that the inspiration for design of the prototype itself came only after an unsuccessful try with the rolled-up paper.

### Other examples

There are several stories in the earlier chapters that belong in this category of understanding existing user experiences:

### Guided fantasy

In chapter 1, "The Mouse and the Desktop," Tim Mott and Larry Tesler explained their devising a "guided fantasy" process to learn what people would want for a text-editing system. They put editors in front of a blank display with a keyboard and a mouse and asked them to walk through an imaginary editing session using that hardware. They explained that the mouse could be used to position a pointer on the screen and that the text would be on the screen. The editors described the process that they used at that time with paper and pencil, and together they imagined typing in the text and creating a manuscript, and then editing that manuscript using the mouse and keyboard in the same way that they would use a pencil.

### Draw your money

Another example is in chapter 6, "Services." Fran Samalionis asked people to draw their money for the design of an online bank. After observational research, the participants were asked to sit down and make a drawing of their financial situation. This helped to expose their feelings and their mental models. As they started to draw, they talked about themselves openly, providing a clear view into the way they felt and thought about money, and hence the kind of services and support an online bank could offer them. This technique could be valuable in a more general service design context, as it helps to describe a complicated system and to establish a framework.

- Children's picture communicator for Maypole project
- Prototypes operated by equipment in backpacks

## Exploring and evaluating design ideas

The second category from the paper is useful during the design development process, when alternatives are being created and envisioned:

> The main purpose of experience prototyping in this activity is in facilitating the exploration of possible solutions and directing the design team toward a more informed development of the user experience and the tangible components which create it. At this point, the experience is already focused around specific artifacts, elements, or functions. Through experience prototypes of these artifacts and their interactive behavior we are able to evaluate a variety of ideas—by ourselves, with design colleagues, users or clients—and through successive iterations mold the user experience.

An example from the paper is for the design of a picture communicator for children, where children used working prototypes for an extended period of time. Despite heavy backpacks containing batteries and drivers for the prototypes, the children were happy to integrate picture sending and receiving into their daily activity.

## Children's picture communicator

> Part of the process of design exploration involves checking out ideas with potential users. For example, in the EEC funded "Maypole" project's exploration of community communications[25] the goal was to create prototypes which would give children an experience as close as possible to that invoked by the intended design solution. Usually, user tests focus on fairly specific functional performance issues. Such tests also generally involve conditions that are not typical of the ultimate use situation, for example they frequently involve outsiders (e.g., as observers or "Wizards of Oz" when some functions need to be simulated by a person). This makes it difficult to answer questions about experience such as: How will people feel about the system we are designing? Will it change the way people behave or think about an activity? Is it compelling to them in their own context? A true experience prototype for users—providing a really relevant experience—seems to require a level of resolution and functionality such that it can be "let loose" into an everyday context and more fully integrated into people's lives.

> For the Maypole project, Nokia built working sets of picture communicators that the design team was able to distribute to children who could take them away and play with them unsupervised for days at a time.[26]
>
> These prototypes required a power pack and transceiver unit that the children had to carry around in a backpack, yet the experience of being able to take pictures and send and receive them to and from friends proved so compelling that the users almost forget about that inconvenience.
>
> As an observer of user evaluations, one knows very quickly if the designed experience is a good one. If it is, people get so involved in the experience that they forget about the limitations of the prototype (e.g., a tether to the computer running the software, or an extreme weight or size hindrance because of limiting prototyping components).

Kids are notoriously tolerant of heavy backpacks, but they are also good at rejecting offerings that have no value to them, so this experience prototype showed the potential of the main proposition of sending and receiving images, as well as allowing the designers to try out the interactions.

## Scrollbars

The exploration and evaluation of design ideas is central to the iterative design process, so it has come up again and again in earlier chapters. As one of the many examples, think of Bill Atkinson in chapter 2, "My PC." In collaboration with Larry Tesler, he honed the art of combining prototype designs with experiments and user trials. He proved that a combination of trial and error and a lot of experiments could lead to successful user interface design. He frequently built new prototypes and tried them on different people.

He gave the example of the directional ambivalence of the arrows on a scrollbar. Which way should the arrows go, and where should they be? When you scroll toward the bottom of a document, the document moves up, so there's some reason to think of a down arrow, and some reason to think of an up arrow. What do people expect? When people see an arrow, which way

do they think it will move? He found, by prototyping and testing, that it mattered much more where the arrow was, than whether it went down or up. If the arrow was at the top, people expected to see more of what was above, whereas if it was at the bottom, they expected to see more of what was below.

### HyperCard

Bill Atkinson was also the creator of HyperCard, which became one of the most valued tools for prototyping interactions in the eighties. As he explained in his interview:

> The most important thing is to start with the user interface, so we use what I call "String and baling wire" prototypes. These are software programs that have no depth to them, which would easily crash if you did anything other than the prescribed course of actions, but with which you can feel what it would be like to use this program.
>
> The process of going from one of these mockups to a rigorously crafted software application that can withstand users banging on it and trying all sorts of weird things is a big jump. It is a different medium. The most important thing with that first medium is to be able to try different ideas and iterate quickly. Prototyping environments, like high-level authoring systems or Smalltalk, are very useful because you can put things together quickly. I found that when people made a HyperCard stack, they could put a prototype together in an afternoon and get it to do what they wanted. If they wanted to craft this result into a robust application that a lot of people would use, they could use the HyperCard as an example, and they could sit down with their C compiler and write the software in a different medium.

### Testing mouse designs

HyperCard was the program used by Bill Verplank to test and evaluate alternative designs for the Microsoft mouse, also described in chapter 2, "My PC." He devised five tasks to test a range of experienced and naive users trying out the various prototypes and existing mice. A tapping task measured the tradeoff between speed and accuracy for the most common mouse usage, that is move and click, by asking the user to click

twenty times back and forth on pairs of targets ranging in size. Next, subjects were asked to trace their way through a maze, revealing the ability to steer. A task to test precision asked the user to position the cursor exactly in the center of arrays of dots, and then to click. Writing the word "TAXABLES," with one character in each of eight boxes, required repeated short strokes with the button down and then up. The final task required typing, then pointing, then typing, then pointing—again and again—measuring homing time as you move from keyboard to pointing device. These human factors trials allowed quick evaluations of a wide range of design concepts, and the combination of user testing and rapid prototyping was a key to the success of the project.

### Faking

Bill was able to break the evaluation of the mouse designs into simple individual tasks, but sometimes the exploration and evaluation of design concepts is in a much more intricate context. Take for example the service design problems described by the team from Live|Work in chapter 6, "Services." They say that service touch-points can be faked with quick and informal techniques. Experiences can be enacted, but the communication works much better if the actors are using props that form evidence in the mind of the onlooker. This evidence can be faked using quick and informal techniques to simulate an advert, a piece of CCTV footage, or an invoice. This type of "archaeology of the future" allows qualitative judgments to be made early in the development process about the implications of a design. It allows customers and collaborators to "play back" their own assumptions as concrete experiences rather then abstract evaluations.

### Service experience model

They use prototyping to get a very intimate and subjective idea about what the experience of using a service could be. The prototype simulates the experience of using services, which are intangible, take place over time, and have multiple touch-points, media, and modes. The concepts of "faking evidence" and

"experience modeling" come together in a "service experience model." This is the way in which a proposal for the design of a new service can be explored, evaluated, and improved through iterative development.

## Communicating ideas

The third category in the paper from Marion and Jane is more about communicating results:

> The role of experience prototyping here is to let a client, a design colleague, or a user understand the subjective value of a design idea by directly experiencing it. This is usually done with the intention of persuading the audience, for example, that an idea is compelling or that a chosen design direction is correct.

An example from the paper is for the design of a Kiss Communicator. This was a research project, looking for ways in which emotional content could be exchanged through an intermediating technology. The pair of prototypes let people have the hands-on experience of creating, sending, and receiving subtle sensual messages.

## The Kiss Communicator

> In this example, "getting into the mood" became a significant set-up task for successfully communicating the proposed experience.
>
> The Kiss Communicator was a concept prototype built to explore ways of using technology to communicate with another person in a subtle, sensual way. The intention was to keep the nature of the physical object as simple as possible, so the interaction was more about the experience of the message.
>
> Designed to facilitate the exchange of emotional content between people separated by physical distance, the Kiss Communicator used wireless technology to transmit the digital equivalent of a personal gesture, such as a wave, wink, or a kiss. Each Communicator connects only with a specific corresponding module, resulting in a secure and intimate one-on-one exchange. To let a partner know that you are thinking of her or him, you squeeze the Communicator gently. It responds with a slight glow to invite you to blow into it and create your "message" in the form of an animated light sequence as the

■ Kiss Communicator

device responds to your breath. The "message" shows while you blow and if you are happy with it, you simply relax your grip and it is sent to the corresponding Communicator. On the other end, the partner Kiss Communicator indicates that there is a message but waits until its owner squeezes it to play back the light sequence.

There are some important conditions necessary to really appreciate the experiencing of this prototype: an intimate relationship, two distant people, sending a gesture, etc. Now imagine sharing this concept with clients in their business suits in a conference room. To help set the scene for the experience in this formal context the designers now usually preface the hands-on experience of the prototype with a short video sequence which shows a pair of the devices being used by a dreamy couple who are working apart. Using conventional devices like soft focus and a romantic soundtrack, the video creates, at least temporarily, an atmosphere that is more appropriate. This situation exemplifies how traditional and more passive communication techniques (like video) and experience prototypes can work hand-in-hand, with the goal of sharing a new user experience with an audience.

The design process, like other creative endeavors, leaps from synthesis to idea without rationale. Inspiration is hard to trust for the recipient, so the designer relies on showing the result to convince the audience. The proof is in the prototype. Seeing may be believing, but "touching is reality," as Alan Kay said.

## The demo that changed the world

We see a prime example of the power of the prototype to communicate in the very first interview in the book. When Doug Engelbart demonstrated his interactive system in real time, in front of an audience at the Fall Joint Computer Conference in 1968, the computer science community moved from skepticism to standing ovation in an hour and a half, and the ideas of direct manipulation of a graphical user interface became lodged in the communal consciousness.

Doug sat at a console in the middle of the stage, with the twenty-foot screen behind him showing the view from the video feeds. The overhead camera showed his right hand using a mouse to point and select with, a standard typewriter keyboard in the

center, and a five-key command pad under his left hand. The prototype of the graphical user interface was communicated and the dominant design was in place.

### Inventing toys and games

We see another example in chapter 5, "Play," when Brendan Boyle describes his process for inventing a toy or a game. A team gets good at producing results if they are dedicated to understanding play and focus on coming up with new ideas that have a chance of succeeding in the marketplace. They spend a lot of time brainstorming, which yields a stream of new concepts. They build prototypes of the designs that are close to the intended result, so that they can try them out on kids and see if they like them. Having the kids play with the prototypes is essential to inform the iterative processes of prototyping. When they have confidence in the design, they make new and improved versions to communicate to the companies that they hope will license the design, presenting them as working prototypes, supported by videos that show young players enjoying them.

### Dynamic documents

Joy Mountford described a dramatic example of the persuasive communicating power of a prototype in her interview in chapter 8, "Multisensory and Multimedia," in the story of bringing QuickTime to the personal computer. She was at the stage when her team had a series of illustrative prototypes running demonstrations of what looked like little movies inside documents. She managed to waylay Jean-Louis Gassée, president of the Apple Products Division at the time, and show him the prototypes. He saw a couple of movies playing, with some text all around them. He stopped, and said, "Now I know why my grandmother wants a computer!" From that moment forward he was convinced of the value of dynamic documents.

### Pick-and-drop

The Interaction Lab at Sony's Computer Science Laboratory in Tokyo is full of working prototypes, which Jun Rekimoto uses to communicate his vision of the future of interactive technology. For example, when he wants to communicate the value of being able to move information between a PDA and wall screen or desktop computer, he shows that you can simply pick an object up in one computer and drop it into another. The demonstration of pick-and-drop in the Interaction Lab allows you to use a stylus to select an icon on one computer, which then disappears from the screen. When you tap on the screen of another computer, the icon appears on that screen and the file has been transferred. It works like "cut" and "paste," but instead of imagining a virtual clipboard, it feels more like the file was virtually attached to the stylus. This "drag and drop" approach is much more direct than anything we use today between separate devices. Even a memory stick requires several steps to transfer the data.

Jun believes that in the near future people will be using combinations of multiple devices, posing a challenge for interaction design, as most interfaces so far have been designed in the context of a single computer. He thinks that it will be possible to build on the concept of direct manipulation across platforms, leveraging the familiarity with drag-and-drop that people have experienced with the graphical user interface. His prototypes communicate the potential.

# Prototyping Techniques

The categories that Jane Fulton Suri and Marion Buchenau devised help us think about why we want a prototype, so the purpose is the first consideration to drive the choice of what kind of prototypes to use. It is striking that there is such a wide variety of types of prototype, fueled by the increasing complexity of design contexts. I therefore thought it worth looking at the different types of prototyping technique as a separate topic and mentioning a few that I have found particularly useful.

Prototyping techniques are much more likely to change over time than user research methods, because they depend on our rapidly evolving technology rather than on our slowly evolving selves. The tools that we use to build a prototype are constantly changing, with new possibilities emerging all the time. I therefore look first at techniques, as people can choose their currently preferred tool.

I have enlisted help from Duane Bray to put together this discussion. He is another of my friends and colleagues based in the San Francisco location of IDEO, so first let me introduce him.

■ Duane Bray

*Photo*
Skylar Reeves

## Duane Bray

Duane Bray studied to be a print graphic designer at the University of Florida. When he was designing posters and books, he discovered that he was much less interested in paper and inks than he was in navigational structures; how do you find your way around a book or perceive a poster? Then he got a job as a "digital designer" and found himself confronted with the challenge of designing screens for early online applications, where he could control the experience that unfolds for the user by manipulating the underlying structure. This was interesting enough to make him want to go back to study, and he returned to the University of Florida for a master's in "electronic intermedia."

He joined IDEO in San Francisco just as the group of interaction designers there was starting to flourish and rose to become the head of discipline for the whole company. In this role he is responsible for helping the community of practitioners to develop shared skills and a common philosophy about their work.

He has found that the strongest common bond that the members of this community bring to their work is a passionate interest in designing experiences, particularly temporal ones. He structures interdisciplinary teams for projects to allow the interaction designers to work with the other disciplines in a flexible way, sometimes contributing an understanding of navigation and flow, or perhaps an expertise in creating beautiful interface animations and behaviors. He is used to explaining the "what, why, and how" of designing interactions, and he thinks of interaction design in three main categories:

1. Screen-based experiences

   The earliest to emerge was screen graphics, or pixel-based experiences, where the designer manipulates pixels to express software interactions. This is similar to the more recent skill needed to design for the Internet, as Web sites are also designed as screen graphics.

2. Interactive products

   The second version is where the physical object is integrated with the electronic hardware and software. If a screen is embedded, the designer must consider the relationship to physical controls and the overall form factor. If there is no screen, the design relies on ambient feedback, using light, sound, or movement.

3. Services

   The third is in the design of services, where the interactivity occurs between a company and the broader relationship with the customer, blending time-based interactions with multiple channels—spaces, products, the Web, and so on. This blurrs the boundaries between interaction design and organizational psychology.

When we look at the prototyping techniques that are needed for each of Duane's three categories, we find that the menu of possible techniques is almost complete by the time we have discussed screen-based experiences, as the other two categories usually also use pixel-based visual displays for output. The other categories add some requirements for additional techniques, as the means that the designer can use to enhance the users' experience are more than screen-based.

The categories are discussed below with some examples.

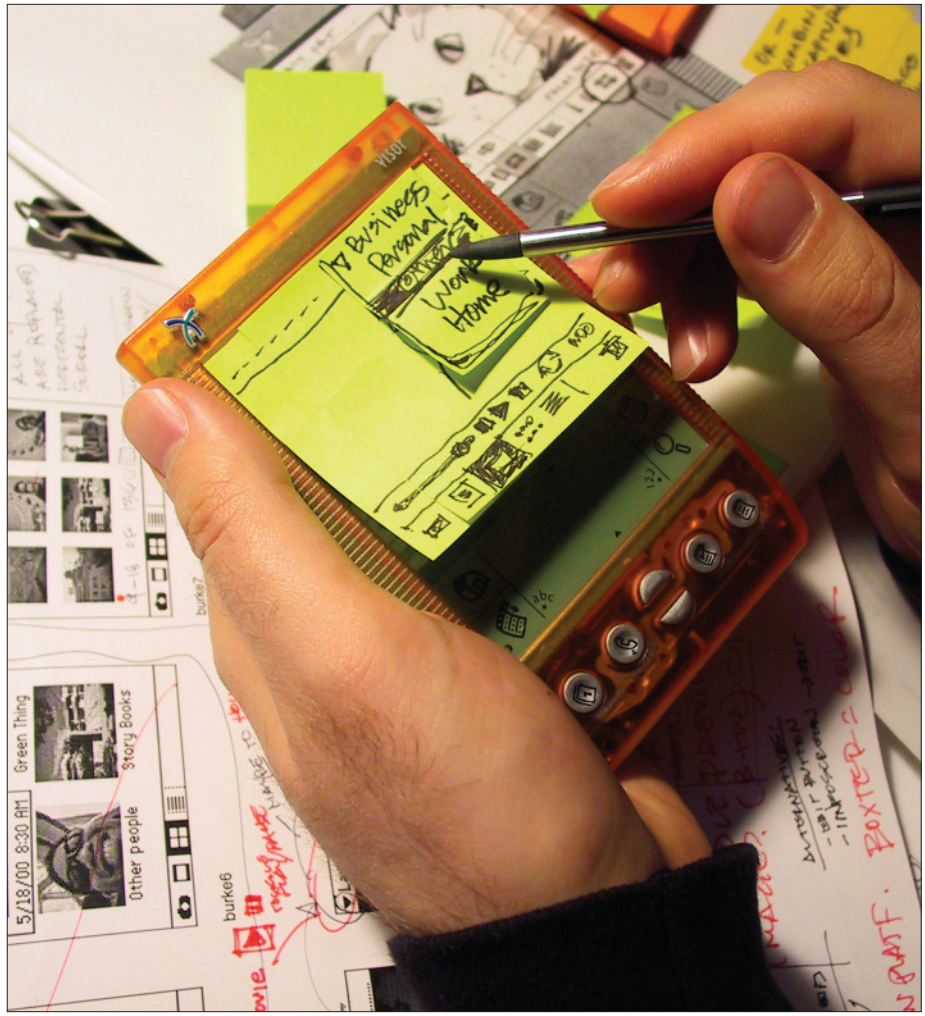
# 1. Screen-Based Experiences

In the early days of graphical user interfaces, the visual skills of a professional designer were usually applied to the design of icons, or the visual representations of behaviors such as window structures or tool bars, or perhaps at a more basic level to design typefaces. In these cases the pixels were a disappointing limitation, as seventy-two little squares per inch gives much less opportunity for artistic expression than the fluid forms of metal type, or the free flow of three hundred dots per inch. Designers, like Duane and I, who are attracted to this pixel-based world, find ourselves interested in the underlying behaviors combined with the limited pixels, so that we can create time-based user experiences as well as static graphics.

When the use of the Internet exploded, and every company realized that it needed a Web site, much of the early work was done by graphic designers. For the first iteration, they often translated a company's page-based print collateral material directly to the Web, just to establish a presence on the Internet. The limited resolution degraded the graphics and did little to exploit the behavioral advantages of the Web. Soon companies like Razorfish emerged, specializing in designing solutions for the new economy that were more than paper solutions applied to screens. Web sites started to be designed to deliver experiences that were more sophisticated, taking advantage of animation and the behavioral possibilities.

Initially design for software was quite distinct from design for the Web, but once they both started to understand how to design behaviors as well as pixels, they blended together much more. Duane describes a more current trend:

> The other thing I think the Web is starting to do is blend the boundary between the Web and interactive media, because the Web now can be a much more dynamic medium than it was before, especially with things like Flash and dynamic HTML. It has given people the ability to think about creating narratives, more like film and video. Instead of being a very static, print-based experience, as it was in the mid nineties, the Web has become animated. It can be designed more like software or interactive media.




## EXAMPLE

*Paper prototyping—in very early, sketch-based prototypes like these, designers can still simulate the user experience (note the use of the smaller post-it to represent a pull-down menu), but ineffective solutions can simply be placed in the recycling bin.*

Here are some prototyping techniques that can be used in the design of screen-based experiences, first in the early phases of exploration, and then in the design of the images and behaviors on displays.

## 1.1 Screen-based experiences: Early exploration

In the early stages of understanding existing screen-based experiences and exploring ideas, the techniques that are useful are shared across all of Duane's categories. They are chosen to help members of a design team generate and share concepts as quickly as possible, and are driven more by the behavior of people than the specifically screen-based nature of the output. This group can therefore apply to all three categories.

The techniques in the "Try" category of the methods cards fit into this group, as do the examples used to clarify the first group in the "Experience Prototyping" paper. They were the game to understand the interaction context for the underwater remote operated camera, the "guided fantasy" devised by Tim Mott for the design of a screen-based publishing system, and the exercise that had people draw their money during the early phases of the design of an online bank.

### Don't forget paper

You can do such a lot with paper. Designers are often too quick to start working in the final medium of their project. For a design where they know the result will be on a pixel-based screen of a particular resolution, they often leap straight into that medium. They can then find themselves sucked down into the details of "pixel pushing," when they should still be experimenting with the basic constraints of usability and perceived value. Paper is accessible and so versatile. You can quickly sketch, lay out, and evaluate interaction design concepts for basic usability, making it possible to rapidly organize, articulate, and visualize interaction design concepts.

You can also demonstrate the structure and behavior of a screen interface using paper mockups, separating the different components of the interactivity. Elements of the interface that will be separate on the screen can be attached to separate pieces of paper, like for example a pop-up or flying menu. You can


*EXAMPLE*

*Designing a handheld media device, the design team used scenario cards to prototype early concepts, asking potential users to evaluate them.*

sketch out ideas, by hand or with a graphics program, and then assemble them on a background of appropriate size and content. This allows you to demonstrate the possible behaviors by hand manipulations, with no need for time consuming scripting for animations, and because these lo-fi prototypes are not precious, they invite more honest comment and critique by users than "perfect looking" screens rendered in Photoshop. You can talk this through with colleagues, or show it to stakeholders, and you will find that the speed with which you can make changes and the informality of style help communication. If there is a particular piece of animation that you want to describe, you can make up a little "flip book" to communicate changes over time.

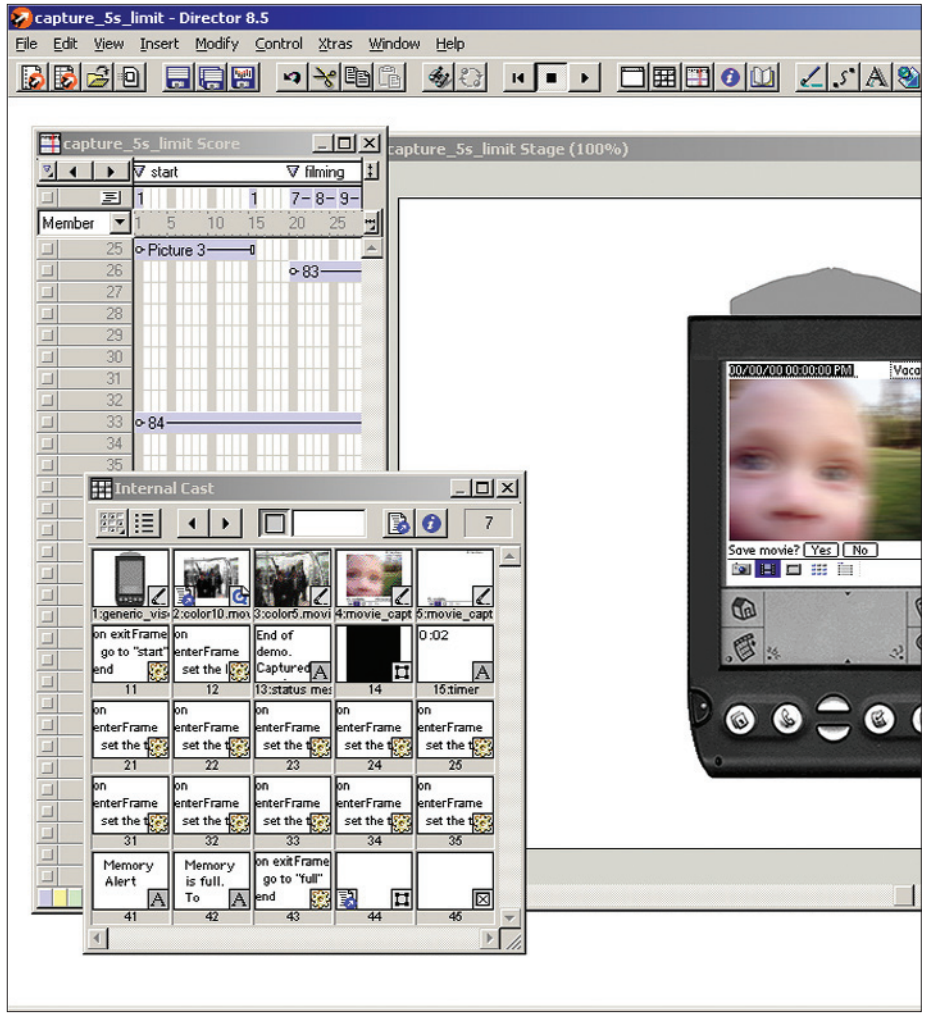
Remember the whole journey

The context of a particular design solution is often too narrow. Designers tend to explain their work within a limited framework that has been imposed from somewhere else. Perhaps they have focused on the constraints within the screen design, without understanding what happened before or after, or in a more systemic context. The idea of the whole journey is very useful to challenge the limits of a context. When we go back to the basics of the experience that a person has as a whole and think about what happens from start to finish, we often find areas of design opportunity.

Prototype techniques that are useful for this more expansive approach tend to push beyond the screen and are therefore more related to role and enactment than specific screens, so we will return to them in the group of techniques associated with designing services. Even within screen-based experience design, however, the journey notion can help to extend a context over time and force us to consider the different situations that will occur.

### 1.2 Exploring, evaluating, and communicating design ideas

The prototyping techniques that you need for designing screens have to combine two-dimensional graphic representations with changes over time. The graphics are usually structured as pixels, but there are some exceptions, such as liquid crystal displays, where there are fixed image elements that are on or off. These




*EXAMPLE*

*This interactive simulation is being created using MacroMedia Director, another iteration in the design development of the camera interface for a PDA. The prototypes were used to develop the design and test users, redesigning the approach until the right balance had been reached between the interactions that were consistent with the Palm operating system resident on the PDA and the interactions that would be familiar to users of digital cameras.*

simple requirements have been met by a large number of prototyping tools aimed at designers, including HyperCard, Supercard, Visual Basic, VideoWorks, and Macromind Director.

Director became the clear choice for designers as soon as it was launched, and the company, now called Macromedia, went on to develop a range of tools for prototyping, including more recently tools like Dreamweaver and Flash, for prototyping Web sites. Tim Mott was instrumental in the early success of the company, after he left Electronic Arts, and he talked about his contribution in his interview:

> After I left Electronic Arts I ran a small multimedia company called Macromind. Over a number of years we turned that into a larger multimedia company called Macromedia. The mainstay of the product line was a design tool called Director. When I got to the company the mantra was, "We've got to make Director easier to use, and less expensive, and then we'll be able to turn every secretary in every office into a multimedia designer and producer."
>
> It seemed like a good idea at first, but we realized that, independent of the tool, not every secretary in every office had the design sensibility to become a multimedia producer. In fact, the users of that product were, and needed to be, professional multimedia designers.
>
> We reversed the strategy. We went out and talked to the users of the product, and found that they didn't care that it was difficult to use because it was so much better than anything else they had. What they cared about was additional functionality. I spent a lot of my career trying to figure out how to make things easier for everyday use, and then I found myself back into the business of building rockets for rocket scientists!
>
> Subsequently the company worked on products much more limited in functionality, and that were much easier to use, and could address the needs of a broader audience. You see some of that today in PowerPoint for instance; there are limited media capabilities in PowerPoint, and it's pretty easy for anyone to learn to use, but you wouldn't find a professional multimedia designer using PowerPoint because it doesn't have anywhere near enough capability.

Director established Macromedia as the primary company to deliver prototyping tools for interaction designers. Today that


## EXAMPLE

*An axonometric view of the layers in the screen design can help to explain the structure, clarifying the difference between the control layer at the front, the content layer at the back and the system layer in between. This example from 2001 shows a Web-based user interface from an IT application service provider (ASP) aggregator.*

trend continues, but more and more interaction designers are using Flash, rather than Director, as their prototyping tool of choice. PowerPoint or Keynote are enough for early sketches and simple transitions.

### Live prototyping

There is a recent trend for prototyping tools and methods for interaction designers to converge with the development tools and techniques used by software engineers who are writing code. Most of the techniques described so far have been useful for prototyping the design of the experience and have not been directly connected to the world of software development. Whether the interaction designer was using paper, Supercard, or even Macromedia Director, a translation was needed for the development engineers to make sense of the design and start to develop algorithms and code. The translation was usually in the form of a specification document of some sort, to transition from design to development, but this tended to leave plenty of opportunity for errors and misunderstandings, so the final implemented product would often fail to match the early design. Duane explains the trend that is called "live prototyping":

> Prototyping tools have been moving closer and closer to real-world code. An example of that is Flash, a tool that has a scripting language associated with it, but unlike the "designer friendly" English language based scripting language, it is something that is much closer in syntax to a true programming language. For example, someone who knows Java can easily use Flash. Tools have been created for "rich Internet applications"—for example, Macromedia Flex and Lazlo—that allow people with software development experience to create Flash applications, without ever actually using the nice-looking graphical interface, just working at the XML code level.
>
> Director and Flash used to use a sequential timeline of the screens, as well as describing what happens on individual screens. As the coding has become more sophisticated, what you end up with is this long movie format, with one frame and a whole bunch of code that updates that frame as you interact, so it is much closer to a real software application.


*Photo* Credit

*EXAMPLE*

*Prototyping tools such as Dreamweaver or Flash can be used to try out screen behaviors for Web sites, incorporating animations, sound and video, in a realistic equivalent of the final result. A Web site like this from Isse Miyake (www.isseymiyake.com—2003) could be prototyped and tested with these tools.*

> You see examples on the Web, where people are able to take Flash tools, or Flash applications, and embed them within a Website, forming a product that can actually be shipped. You also hear more and more about companies that are willing to accept a "Flash front end" that accesses a database and that becomes your prototyping tool. The boundary between prototyping tools and the development world is blurring.

This chance to access real databases using prototyping tools is a huge leap forward for interaction designers. "Live prototyping" can give you the opportunity to run user tests with large numbers of participants as you develop the design, so that each attempt teaches the lessons of reality and gives the designer the chance to try again with the knowledge of what worked for people.

Terry Winograd pointed to an example of this in chapter 7, "Services," when he took a sabbatical from Stanford to spend a year at Google. He was excited to be there, because this was the first company that he had consulted with that was having a major impact on the market place. He enjoyed the direct connection to the needs and demands of serving hundreds of millions of people, trying to understand what kinds of things they use and do, and marveling at the ease with which it was possible to try out prototypes with enormous numbers of users.

Live prototyping and the evolution of prototyping tools are all about moving closer and closer to the implemented result. The culture of software development, where people share sample code or comment on their code as a way to build communities, has parallels in the prototyping world. You can easily go online and find people sharing code as a way to help one another out, so people are learning prototyping tools in a similar way to how they used to learn new programming languages.


## EXAMPLE

*A cell phone is an example of a product where screen interactions are integrated with physical and tactile behaviors. This Sony CMD-J5 Dualband Mobile Phone from 2001 shows a combination of lively screen graphics with various controls, but the audible and vibration aspects of the interface cannot be seen from the photographs.*

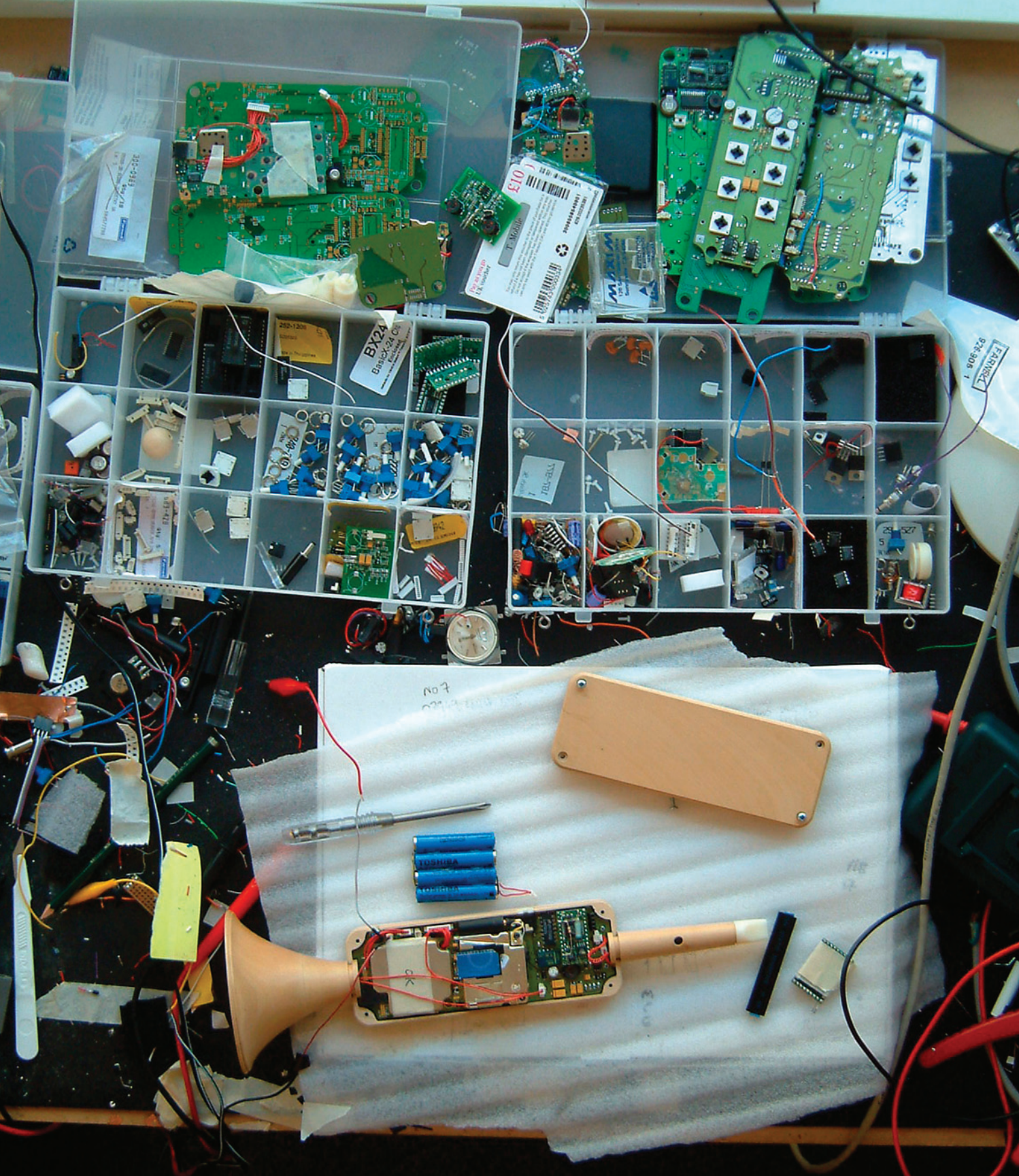
# 2. Interactive Products

The second of the three categories of interaction design that Duane Bray defines is in the context of products. The need for a new kind of design is triggered by the ability of everyday objects to "behave," enabled by interactive technology. This is the version of interaction design where the physical object is integrated with the electronic hardware and software, and people are interacting with ubiquitous computing devices, sometimes not thinking of them as computers at all but rather as products that are responsive or intelligent in some way.

To prototype the behavior of an object, you need tools and techniques that can simulate behaviors of the physical things that are both digital and interactive. For generic computers, whether desktop or laptop, the physical objects are input devices—keyboards, mice, track pads, tablets, and so on—where the traditional prototyping techniques of product design work well enough as long as user testing is cognizant of the interactive tasks that will be required, as with the tests for the Microsoft mouse described above.

The new challenge for interactive products occurs when the physical object and the electronic behavior are integrated. The cell phone has a screen and push buttons, but it also has rotary controls, or perhaps a jog shuttle, slider switches, acoustic feedback, ringing tones, and a vibration behavior for the silent mode. How do you prototype all that? The techniques are similar to those used for mechanical engineering and electronic hardware and do not need to be specific to designing interactions. Simple mockups are useful in the early stages of the design, using any tools that are at hand, perhaps random objects taped together, mockups constructed from foamcore board, or Lego, using the MindStorm robotic invention system to add some behaviors.

Sometimes it is important to engage team members in radical design ideas before they are fully resolved. For example, in an exploration of television remote controls, the designer wanted to explore the specific experience of switching channels, while


BX24
TOSHIBA

ignoring other aspects of functionality or look and feel. He was especially interested in exploring the implications of a more intuitive and multisensory design solution. He created what he called "behavioral sketches" which were simple electronic circuits containing a few lines of code (Basic Stamps™), encased in off-the-shelf soap dishes. Two experience prototypes were controlled by a tilting gesture, switching channels up or down, one with visual feedback with moving bands of light and the other with tactile feedback using vibrations. By tinkering with the simple software program, he was able to quickly develop and test subtle iterations of product behaviors and user experiences.

The photograph opposite of the workbench associated with the "musical mobile" shows a typical collection of hardware components for prototyping simple electronic circuits, combined with the prototype parts for the physical form of the device. The integration of these prototyping techniques with the screen-based methods requires ingenuity, but nothing completely new for the experienced prototyper. The advent of programmable chips like the "Basic Stamp" has made the prototyping of interactive products accessible to interaction designers.

*EXAMPLE*

*This "musical mobile" was one of the Social Mobiles in a research project, which commented on the social problems that are caused by mobile phones. The phone requires you to play the tune of the phone number that you want to call. The public performance that dialing demands, makes you look somewhat ridiculous, so you have to be a pretty confident that it is appropriate to make a call in that time and place. The prototypes were built using the electronic hardware in the photograph and simple code on programmable chips.*


NOKIA
[My Locker]
Options
Exit
Games
Ringtones
Pictures
Take Me Home
Help
Add to My mMode
Share This Page
NOKIA
[mMode Home]
Options
Exit
My mMode
Spotlight
mMode Guide
Games & Tones
Mail & Messaging
Search
Where I've Been
NOKIA
[Spotlight]
Options
Exit
Cole Porter
Chris Gallery
Latest & Greatest
Special Offers
People's Choice
Pam's Picks
Mike's Picks
NOKIA
[My mMode]
Options
Exit
ESPN Leaderboard
Wired Magazine
Media Line
New For You
New to My mMode
Manage My mMode
My Account

# 3. Designing Services

The third kind of interaction design described by Duane Bray is the design of services, where the interactivity occurs between a company and the broader relationship with the customer. The design of services has usually in the past been thought of in terms of training the employees who work for the service provider, but as technology enabled services have become more prevalent, interaction design is increasingly needed to create successful solutions. The need to articulate processes for the design of services is a recently emerging challenge, so chapter 6 of this book is devoted to the subject. Duane sums it up this way:

> The service category to me is an interesting one, because there may be outcomes which you would look at and say that they are very specifically something that an interaction designer would do, but often there is something there which is about designing the relationship or the interaction that a company has with its customers. That is much less tangible.

■ AT&T mMode service

*Photo*
Courtesy of IDEO

As an example, Duane was working on the design of a new service for AT&T Wireless, called "mMode." The company was facing the challenge created by the late arrival of cellular phone services in the USA. Europe and Japan had been evolving mobile phone services over a long enough period for people to get used to the new types of functionality on offer, but in the USA the situation was suddenly changing. Cell phone coverage had been incomplete enough to drive competition on the basis of the ability to make a call, so that people were thinking about their mobile phone as only useful for voice conversations. Suddenly they were offered a phone with a camera in it, so that they could send pictures to people, or they could trade stocks on the phone. It was necessary to reposition the phone as a much more multipurpose lifestyle tool. The team developed a design philosophy for the new service that could be applied to the whole offering, whether it was text messaging, data-services, or photography. Duane describes the prototyping techniques:

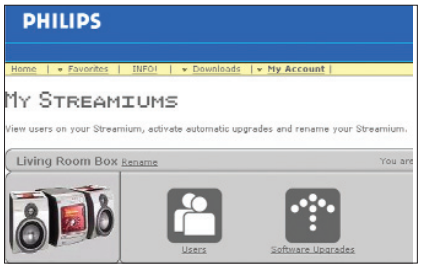
With live prototyping, we have the ability to embed prototypes in the real world in the way that they would be as finished products. We can also connect them to actual data, so that they can make use of real-world content. For the AT&T Wireless project, we prototyped a new way to have a user interaction running within the cell phone's browser. We were able to create an initial prototype of the experience and put it up on a server that we could point our mobile phones to, to prototype and test as if it were the running service.

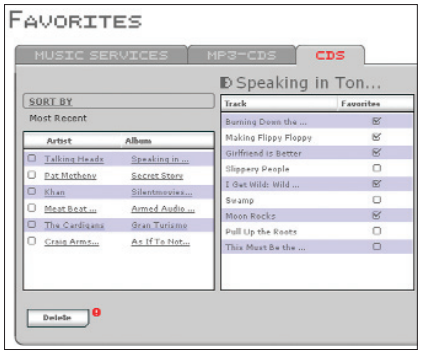
Because our prototyping tools are becoming more sophisticated, we were using XHTML and using the same tools as the in-house developers, so there was no effective difference between the prototype and the finished product, either in appearance on the phone, or in the reality of the code; it was just a matter of who actually owned and created that code. For the prototype it was us, and for the final product it was the software developers at AT&T Wireless.

When you are designing a complete service, your contexts and constraints are intrinsically complex. They are formed by the combination of all the individual interactions, whether technology-based or human-to-human, which make up the various touch-points within the service. This makes the use of live prototyping even more valuable, as the results of testing the prototypes are much more likely to be realistic.

Duane also cites the example of his helping the team work on a service called Streamium from Philips, which allows people to access digital content for entertainment, including music and video:

> We prototyped a version of it in Flash that was remote control–driven, for TV or Media center use. Just testing the user interface of browsing categories was not very effective. One of the challenges of this user interface was that there is such a wealth of content out there, including so many digital radio stations, so many different playlists, and all these different archives. What you want to be able to experience in the prototype is not the subset of that world that you might be able to think about or code in. You actually need the reality of the scalability of the actual content that is out there. A live prototype is needed to go out onto the Internet, access the content, and update it dynamically. From a user experience





- Philips Streamium—personalization
- Philips Streamium—favorites
- Philips Streamium—splash screen

perspective the prototype was the same as the final product, although Philips did not develop the final version in Flash.

There is often a disconnect between the design process and the real world. We like to have a small and comfortable set of parameters to work within. For example, if we are designing a new operating system, we don't actually prototype the situation when you have ten thousand files to work with. Lack of scalability often hurts our design solutions. With the Philips example we allowed scalability to intrude into the design process by accessing the real-world material.

## Designers and developers are converging

Live prototyping is not the only trend that is pushing software developers and designers together. Another is called "Xtreme programming" (XP). This approach to software development adopts some of the principles of human-centered design. A "client" is appointed as an end-user advocate to work within the team, and the developers use both collaboration and iteration techniques that are similar to the design process. Duane enjoys the convergence:

When we've worked with clients where Xtreme programming exists, the developers are pre-wired toward understanding and embracing the process that we're using; the process of going out into the world and understanding end users, and turning those into insights and opportunities. They try things out repeatedly, knowing that they are not perfect the first time and learning from the imperfections as they evolve. They collaborate, realizing that the value of the team is greater than the value of the single individual contributor. That process is sometimes very alien to traditional software developers, who work in isolation on a piece of the whole.

One of the big challenges in software development is the notion that code is precious; once you've got something in code, you can't possibly throw it away if it's inappropriate, because too much value has been put into it. That used to be one of our philosophical disconnects with software developers.

In Xtreme programming, people code in pairs for a particular iteration and rotate the pairings over the process of development, so you are cross-pollinating the entire team. This gives you collaboration and cross-pollination happening at the same time. You build

constantly! One statistic that I read was that an Xtreme programming team of forty people might have eight to ten builds of their product in a day. The idea is that you always make something that you can use, even if it's only the tiniest piece of a feature. You always make something that can be validated. Validation is against the "client," who represents the end user, or it could also be sent out for end user validation.

It is a way to say, "We believe that this product has all of these features, and needs to do all of these things." You are testing the reality by iterating on as many of those components as you can, and seeing how they work. Then you can either eliminate the feature, change its importance, or modify it based on what you find out. This set up makes the train of thought of the programming team mirror what is happening in human-centered design, so designers don't have to be evangelists for user validation, or iteration, or collaboration; it's there!

### Space matters

The design of the work environment itself affects the way that people can collaborate and cross-pollinate ideas. In order to develop the kind of collaborative pattern required for the Xtreme programming model, and also for interdisciplinary design teams, one has to provide open and flexible workspaces, with furniture that is easy to move, and portable equipment, such as laptops that can communicate wirelessly. Not many of the major software development companies are providing environments like this. One of the reasons that Microsoft has such an enormous campus of buildings is that every developer has his or her own office. If you look through the halls, you see endless rows of offices, where each person can close the door and feel free to work on their individual piece of the puzzle.[27] There's nothing about the design of the space that helps people come together. Duane describes the contrast between this and the offices of Xtreme programmers.

I visited a client where they're trying to use some of those principles. There you see shared offices where between two and four people are working together. They delineate the individual space where their computer is, but they also have public space in each office where they come together. The same client also has very public open spaces,

where people can come together to collaborate, share, work around a computer or whatever. When you look at what the people are doing there, you see something that looks exactly like the offices of an interdisciplinary design team. They are using the same principles! I think that if schools of thought like that become more and more adopted, we are going to see a world where design and development can work hand in hand much more easily than they have in the past.

We know that we can do that successfully at IDEO; it is in our culture. It will be great to find ways of doing it in collaboration with our clients, to create that sense of telepathy.

# Process

As a designer, I am much better at synthesis than analysis, so I am short on good theories about designing interactions, but I can tell you in practice how I do it. To finish my story, here is a description of the process that I have evolved for designing interactions, working with my colleagues as a community of practitioners. I have done my best to explain why interaction design exists, and how interaction designers contribute to the development process. I have stated my belief that if you are going to create good designs, you first have to understand people—what they need, want and enjoy, as well as how they think and behave. I talked about my conviction that the practice of designing interactions is enabled by prototyping, that we arrive at good designs by prototyping early and often, by trying out ideas as quickly and frequently as possible, and by taking them to the users for responses and evaluations.

■ A community of design practitioners from IDEO looking through stereo glasses

*Photo*
Joe Watson/IDEO

Is this focus on people and prototypes enough? Can we rely on just those two simple strategies to create excellent designs? I'm afraid not, as the constraints will come from the full context of the design problem, not just the people. Even so, a focus on the people is important enough to be the best place to start and is particularly valuable when we are designing something new; if we are designing a new version of something that already exists, we have to be sure that we have a full understanding of the "state of the art"—the constraints that defined the previous result.

## Designing Something New

The keywords of "people and prototypes" are needed most when innovation is the only possibility. The essential first step will be to start from understanding the latent needs and desires of the people who will use the design. You are not just designing for the average person either; you will need to understand the viewpoints of the full range of people who will interact with the outcome, from the slowest to the fastest, the most naive to the most expert, and the least experienced to the most fluent. They will probably be surprisingly different from you, so it will only be by understanding them that you can avoid the trap of designing for yourself.

The keywords of "people and prototypes" are needed most when innovation is the only possibility.

The context of the design problem is not just the people. You will need to understand as much as possible about everything that will affect the solution: what it is for, how it will work, how much it will cost, and so on. Each of the team members will need to listen and learn from all of the other experts to find out as much as they can about the context in a short first burst of discovery, but without worrying about not understanding everything.

You, that is, the collective "team you," will need to stop researching and let your tacit understanding of the problem help you come up with design ideas, creative leaps, and first solutions. Prototype as quickly and roughly as possible, just enough to communicate each concept to one another. Then evaluate the designs. They are most likely really bad solutions, so try again. If the ideas seem a little better next time, you can make the prototypes a bit more descriptive. Perhaps the evaluation this time will be with end users as well as your peers. That will probably be a shock: "Why don't they think more like me?" Try again! Prototype early and often, making each iterative step a little more realistic but minimizing the time and effort invested each time, relying instead on the learning that feeds your subconscious each time you try. At some point you will know that you have arrived at a good design, both from your shared intuitive judgment and from the way the people who are evaluating your attempts react. When their response changes from critique to involvement in the result, you can start to hope that you are on the right track.

# Designing a New Version

If you are designing a new version of something that already exists, "state of the art" is the most useful starting point. The chance to set a precedent with something completely new is rare. In most cases you are designing a new version of something that is already there, so you can research what has been done before, learn the lessons from previous attempts, discover guiding principles, and extract knowledge from the precedents. There is design wisdom out there, but it takes time to shake out.

Thinking about the people that will use the design is just as important, and the process of working through iterative versions of your solution with prototypes and evaluation will still be the best and fastest way to get to a good design. The difference is that you need to spend more time and effort understanding what has already been done in the first place, so that you are building on the state of the art rather than trying to reinvent solutions that others have developed before. The research phase that comes before the first design solution or creative leap must be thorough; your team must catch up with everything that has already been done.

If you are designing a new version of something that already exists, "state of the art" is the most useful starting point.

Look at the competition, try the previous designs, research the literature, understand the design principles, compare and criticize alternative versions, get to the point that your shared mind is so full of the existing designs that you can drop them back into your subconscious, and know that whatever you come up with will automatically build on the past. Once you have reached that state, you can move back into the iterative design process of thinking about other people, coming up with an idea, building a prototype, and trying it out.

I hope that this book is in itself a good summary of the state of the art for designing interactions and that the collection of interviews will help you discover relevant constraints for your design contexts. When I look back at the material, I am conscious that it takes time to establish a state of the art. The chapter about designing services has very little in the way of notes referring to

- Bruce Tognazzini
- Jakob Nielsen
- Don Norman

well-established principles and knowledge of the subject, because at time of writing, technology-enabled services are only just emerging as an opportunity area for interaction design.

In contrast to services, the design of the personal computer has had time to mature, and there are plenty examples of well-expressed explicit knowledge on the subject. The first two chapters tell stories of the emergence of the dominant designs for personal computing, through the accounts of a few of the people involved. The state of the art for designing interactions with personal computers has been developed by many more people than these. There has been time since the creative confusion of the original invention to articulate the user interface design principles clearly and simply. Take, for example, the work of Bruce Tognazzini, who has explained "The First Principles of Interaction Design," by listing twenty-two principles on his Web site.[28] Bruce founded the Human Interface Group at Apple, where he wrote and performed for the video "WorldBuilder"—expounding the design principles for Macintosh software. He went on to Sun and is now with the Nielsen Norman Group.[29] He has published two books[30] on interaction design, making him a "must read" for any designer who wants to work on screen behaviors for desktops. This set of design principles and constraints is wonderful for us now, but it took a long time to articulate after the original innovation achieved by Larry Tesler and Bill Atkinson. And, by the way, the other members of the Nielsen Norman Group have done a lot to clarify the constraints of interaction design. Jakob Nielsen is known as the guru of Web page usability, and Don Norman has done wonderful work bringing an awareness of usability issues to the design of objects both physical and digital, from "The Psychology of Everyday Things"[31] to "Emotional Design."[32]

These are just a few of the luminaries who have helped designers, and those people who are affected by the designs, to understand more about the new constraints of designing interactions. Their contribution is essential to developing the new design discipline to a level of maturity that has an accessible state of the art and allows it to be learned, taught, and practiced on a larger scale. The work of these people, and of many other

members of the HCI community, helps us lift our understanding from the subconscious to the conscious level. It gives us a state of the art to draw upon, so that our process begins with a fast start.

## Elements of the Design Process

We can generalize the interaction design process with these ten elements: constraints, synthesis, framing, ideation, envisioning, uncertainty, selection, visualization, prototyping, and evaluation. They will often be used in the same sequence, and repeated iteratively, but the most productive process is usually out of order; it can sometimes seem almost random. Remember that pinball machine analogy.

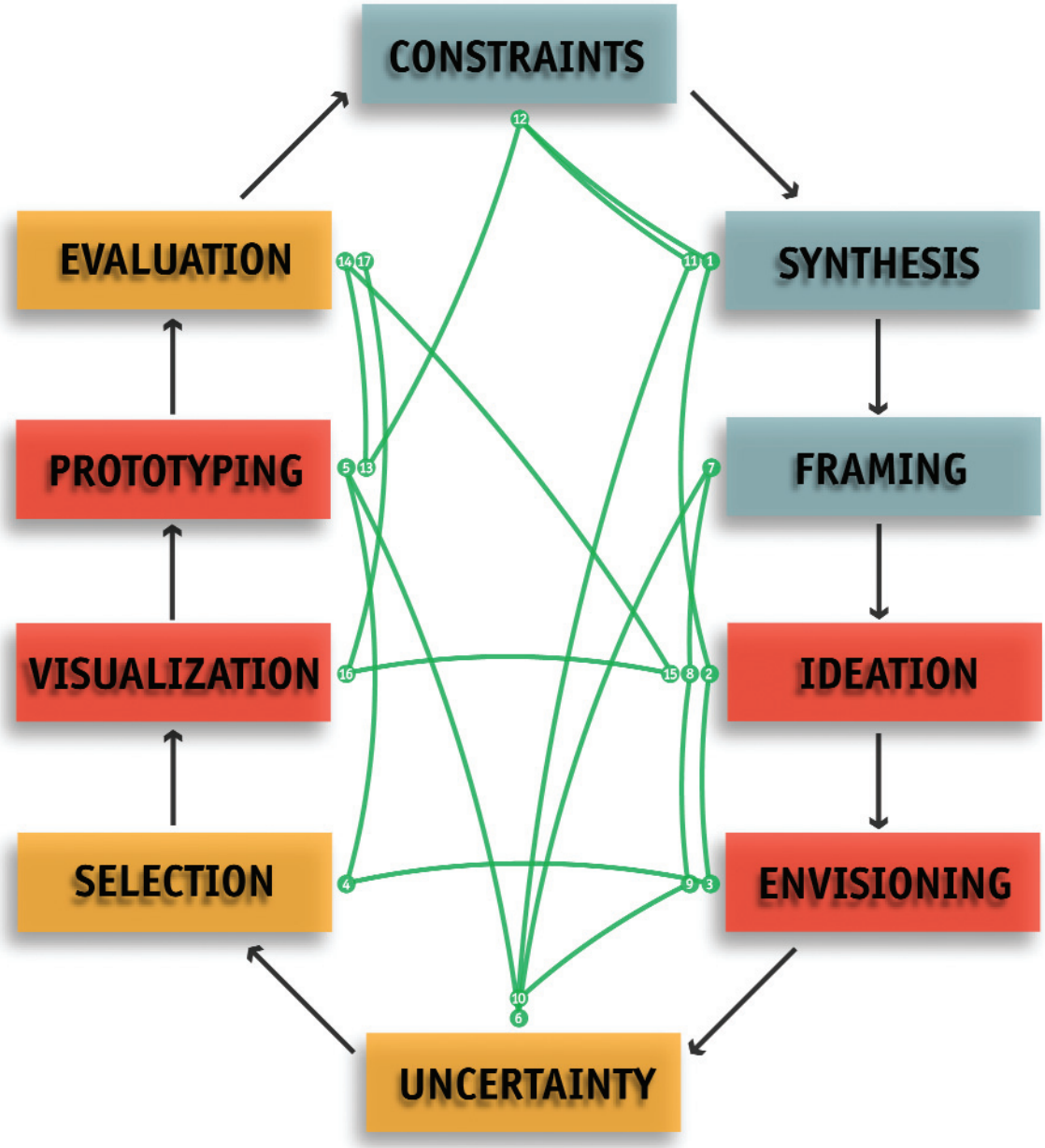
The fastest progress toward a successful design will be made when these elements are used quickly and repeated frequently, but usually not in the same order!

### Constraints

Understanding the relevant constraints starts the process. The constraints come from everywhere that matters to the project. The "State of the Art," the needs and wants of the users, their preconceptions, mental models and expectations, brand awareness, functional constraints, technology, environment, financial constraints, business constraints, competitive analysis, conversations with relevant people, briefing discussions, brainstorms, and on and on—everything that could be important. The constraints must be absorbed by the subconscious mind of the designer or designers. If the context is complex, a design team is more likely to succeed than an individual, so the "shared mind" of the team will be entrusted to absorb the information.

### Synthesis

Synthesis occurs as the subconscious, shared mind of the design team (or the designer if the problem is simple) absorbs all of the relevant issues. The ability of the team members to synthesize ideas is an essential skill of design thinking. The ideas may be about design solutions or other elements in the process, such as a framework or the nature of an experiment or prototype. The tacit




*The dark arrows show a general tendency toward a cyclical process, with the color coding of the titles indicating activities of similar types. In real life, as is illustrated by the project shown in the green sequence, the pattern is complex and less orderly than a clockwise cycle.*

understanding of the constraints allows the subconscious background processing of information to be happening all the time. This background synthesis explains why people who work in design teams often come up with significant ideas without knowing where they come from. They say, "I had this idea last night," or "I suddenly realized as we were talking . . .". Because it is subconscious, the element of synthesis is not usually mentioned in explicit descriptions of the creative design process. It is vital to success, however, and needs to be appreciated, planned for and enabled. A successful team will feel relaxed enough to synthesize well, but a stressful atmosphere can get in the way.

## Framing

Framing articulates the synthesis simply enough for ideas to happen. A framework is in itself a form of synthesis, in that it clarifies the issues by applying insights that create the first level of order from the chaos of all the constraints. It is not a design idea but forms a way of thinking about and evaluating possible design ideas. Coming up with the right framework for a particular project is also a design process, involving many of the other elements described here. One project may be best framed by a journey through the experience, another by a four quadrant analysis of people's attitudes, and another by a nested hierarchy of attributes. The diagram opposite is in itself an attempt to frame the design process, by showing both a generalized likelihood of a cyclical iterative pattern and a specific example of a much more chaotic individual reality.

■ An iterative process that tends to be cyclical but in practice is flexible and pragmatic

## Ideation

When the ideas start to arrive, they are not always great ideas, but they seem to the design team (or designer) to have an "Ah ha" quality when they are first thought of. There are multiple levels of design ideas, some of them encompassing the whole context and others about tiny details. If a good framework is in place, it helps to position the pieces, but ideation happens throughout the process, not just between framing and envisioning.


Defer judgment
CONTAINER
KIDS
ROLA TUBE BOLLARDS
contact lens

Brainstorming can give a fast start to ideation and is often most useful early on, as the constraints are being shaken out. A typical brainstorm at IDEO has eight to ten participants, with one or two experienced recorders, dubbed scribes, who record the ideas as they flow from the group. Each session lasts about an hour, and 50 to 100 ideas are recorded. The conference rooms have the rules of brainstorming printed along the top of white boards, to remind everyone to defer judgement, encourage wild ideas, build on the ideas of others, stay focused on the topic, and to keep to one conversation at a time.

Ideas can come at any time, often from unexpected directions. The cycle is often interrupted by a great idea, triggered by an other element in the process. If there is consensus among team members that a new idea has value, it is usually worth stepping back from the process and going back to first principles to help decide what to do next. A good idea can cause a process reset.

## Envisioning

Ideas are like dreams until they are visualized into some concrete representation. The representation can be any sort of description of the design, whether visual or behavioral, or a combination. You can use shortcuts when you are communicating to team members or peers, but there must be enough clarity in the representation that you know something of what the design is like. The journey from "head in the clouds" to "feet on the earth" can be sudden and traumatic, as it is the envisioning process that helps you immediately see what the idea is really like. Self-delusion is no longer easy.

■ Brainstorm at IDEO

*Photo*
Roberto Carra

## Uncertainty

Deep uncertainties are likely to follow envisioning, or visualization, or prototyping for that matter, as you analyze the potential of the solution. The design process is good at generating alternatives and making them realistic enough to evaluate in some way. Uncertainty is a necessary factor as a precursor to selection.

The subconscious "shared mind" (or individual mind) is now busy synthesizing unanswered questions about the validity of each of the alternative ideas. Is it simple enough to understand? Is it consistent with what came before? Can it be made to work quickly? There are always plenty of uncertainties that are worth trying out.

### Selection

It is time to choose. A manageable number of alternatives must be chosen to take forward to the next step. When a creative team is working well, there are nearly always too many good ideas, and you have to be firm in choosing the most promising group, without feeling too bad about the need to reject the rest. Lively differences of opinion and discussion are normal during this process, unless a clear leader is entrusted by the organization to take the decisions.

### Visualization

The visualization element is closely related to both envisioning and prototyping. It may be a small step from the representation envisioned earlier, or it may be taken a lot further. The difference is that envisioning implies a glimpse into the nature of an idea, but visualization is more complete as a representation; it should be convincing as a communication of the potential reality of the concept. visualization implies a representation that is perceived by the viewer as realistic but may at the same time be dysfunctional. This is in contrast to a prototype, which always looks to test some aspect of functionality. For screen-based experiences, sketches are often useful. For behaviors, some kind of script will be wanted. I use the word "visualization" broadly, implying more possibilities than the merely visual.

### Prototyping

Prototyping is about testing any aspect of the way a design is expected to work. You can create a prototype that represents an idea that has been selected and visualized. Alternatively, you can

test any uncertainty and come up with the simplest and quickest form of prototype that will allow you to examine it, to decide whether to move forward with that aspect of the idea, or to drop it and try another approach. As the iterative cycle of development progresses, prototypes tend to get more holistic and inclusive. In the early stages you are looking for the roughest possible prototype to help you clarify an uncertainty, but as you approach the final result, the prototype looks more and more like the intended design. The final prototype before release for implementation is likely to include realistic interactions, both for behavior and appearance, which can be tested for evaluation and approval.

### Evaluation

In practice, evaluation is needed many times during the development process. In the early iterations, the choices can be made quickly by the team members themselves, or the captive "clients" who are assigned to the process. As the design matures, more complete prototypes are likely to be relevant, like the experience prototypes or the live prototypes that we have talked about, in which case a more thorough and structured user evaluation will be worthwhile. The results of the evaluation can form a new state of the art for the next attempt to create a good design, so the addition of the results to the package of constraints can trigger a new cycle of design development. When you are getting close to a good design, the evaluation process is more likely to yield minor adjustments. By this time it is too late to go back to first principles, but evaluation still helps the design team avoid the pitfalls of narcissism. A good motto for designing interactions is to evaluate early, often, and as late as possible.