10

Cultures of Prototyping

Michael Schrage

The great ethnographies of prototyping have yet to be written. However, it is demonstrably clear that fundamental differences in corporate prototyping cultures lead to qualitatively and quantitatively different products. Understanding those fundamental differences is essential for any organization that wants to transform its new-product development.

Software developers have devoted a great deal of attention to the design of software-prototyping tools, with the hope that the use of these tools can greatly increase the software designer’s fluidity of iterative design. The use of rapid prototyping to accelerate creativity can also be observed in other design disciplines and is embedded in what Michael Schrage calls a company's *culture of prototyping.*

Schrage approaches the question of design as an observer and journalist—a starting point that is different from those of the practitioners and scholars of design in the previous chapters. As a syndicated columnist writing on computing technology and business for the *Los Angeles Times*, and as a research associate for the Sloan School of Management at MIT, he has studied what succeeds in practice in the commercial world of high technology.

In this chapter, Schrage describes the role and conduct of prototyping in industries such as automobiles and consumer electronics, drawing on material from interviews with design-firm managers. His analysis illustrates how the principles discussed by David Kelley and Donald Schön in Chapters 8 and 9 are put into practice in industry. The reflective conversation that Schön describes takes place in a rapid cycle of building prototypes, testing them, scrutinizing them, and redesigning them. Schrage's portrayal of the culture of the prototype-driven organization, which fosters creativity and innovative design, is based on part on IDEO (Profile 8), which has been built around a prototyping philosophy.

Although the examples in this chapter are drawn from products other than software, the analysis applies directly to software development. Prototypes provide Schön's *backtalk* to the designers, and also can serve as an essential medium for information, interaction, integration, and collaboration.

—Terry Winograd

§ § §

Whenever organizations innovate, culture matters. The Toyota culture that builds a Lexus projects a design sensibility different from that of the Cadillac produced by General Motors. The technocultural idealism carving Apple's Macintosh (see Don Norman's account in Chapter 12) is inherently different from the organizational imperatives revealed by Microsoft's Windows development. The values that organizations hold shape the value that they create.

The culture of an organization has a strong influence on the quality of the innovations that the organization can produce. Each enterprise reflects a community of people with characteristic work and thought patterns that show up in the results of their activities. If we truly want to understand and influence how corporate cultures create valuable new products, we need to understand more fully the role that culture plays in creating new prototypes.

The *prototyping culture*—the media, methods, and styles that companies use to manage their multiple models of reality—offers a wealth of critical insights and opportunities into how organizations design and build value. In fact, David Kelley (see Chapter 8) asserts that he can tell almost anything worth knowing about a company's new-product development by simply sampling a few prototypes. "I could tell you absolutely everything, from the care of the models to the quality of the thinking of the designers."

A prototyping culture, like all cultures, is a mixture of the explicit organizational structures and the tacit understanding and practices of the participants. Just as companies have formal organizational charts and informal interpersonal networks, most companies also have formal prototyping processes and informal prototyping activities. In some corporations, formal prototyping processes rule; in others, the informal prototyping culture—like the informal network—is the context in which work actually gets done. At Apple Computer, for example, there is a strong formal prototyping culture. But Apple's invisible colleges of technical and marketing experts ensure that the informal prototyping culture enjoys considerable influence over the ultimate design. Within some innovation environments, prototypes effectively become the media franca of the organization: the essential medium for information, interaction, integration, and collaboration.

In this chapter, we first characterize the different aspects that make up a prototyping culture: prototypes and specifications, prototyping media, and the prototyping cycle. Then, we look at what it means to change to a more prototype-driven culture, and how this change can affect the values and the success of a company.

Although the bulk of the observations here are drawn from the world of industrial design, they distinctly resonate with the software-development experience. Software developers have prototyping cultures every bit as strong as those of industrial designers. Much has been written about the difference in corporate cultures among companies such as Apple, IBM, and Microsoft; the differences show up in these companies' approaches to prototyping as well.

Prototypes and Specifications

Virtually all innovative designs emerge in the interplay between two dueling representations: the wish list of specifications that describe and define the new ideas, and the prototypes that attempt to embody them. Prototypes too often confirm that what we wish for is unrealistic or ill conceived. Conversely, prototypes can reveal that the designer's wishes were not sufficiently imaginative. Specifications and prototypes can be mutually reinforcing, or they can prove to be implacable enemies.

The tension between specifications and prototypes is not unlike the historic tensions between theory and experiment in physics. Theory describes what is supposed to happen, and experiment tells what happens. The culture of physics has always been a push and pull between theorists and experimentalists. At times, theory dictates the experimental agenda; at others, experimental discoveries drive the theoreticians. Just as managing the dialog between theory and experiment is essential to the advancement of physics, managing the dialog between specifications and prototypes is essential to the advancement of design innovation.

Simply put, some innovation cultures are specification driven; others are prototype driven. Small, entrepreneurial companies built around a brilliant product concept tend to be prototype driven. Companies that need to coordinate large volumes of information and to manage a large installed base of users—companies such as IBM, AT&T, and Aetna Life & Casualty—tend to be specification driven. Specification-driven cultures also draw heavily from market-research data before they move concepts into the prototyping cycle. In prototyping cultures, prototypes are often used to elicit market feedback before final production.

When the dialog is poorly managed or breaks down, the results can be horrendous. An organization may spend thousands of hours developing detailed specifications, only to have the first prototype invalidate most of the work. This kind of setback has been particularly evident in areas such as software development, medical instrumentation, and airplane-cockpit design.

Similarly, industrial designers can craft absolutely breathtaking prototypes that prove impossible for cost-effective manufacture. The same can happen with software—software designers create a rapid prototype that demonstrates functionality that is impossible to implement with acceptable resource efficiency on the available delivery platforms.

Based on his firm's client experiences, David Kelley argues that organizations intending to be innovative need to move from *specification-driven prototypes* to *prototype-driven specifications.* In any event, it is clear that organizations prizing prototypes over specifications have fundamentally different design perceptions and processes. There are counterpressures, however. Many organizations believe that manageability means predictability. The idea that you can play your way through prototyping to a new product is anathema to managers educated to believe that predictability and control are essential in product development.

The Prototyping Media

Not all prototypes are the same, either in how they are built, or in the role they play in the design process. The medium of prototyping can have a strong influence on the whole design enterprise. Looking at the use of prototypes in the automobile industry, observers such as Michael Barry of GVO (a highly regarded design firm) have speculated that Detroit's competitiveness problems in the 1970s and 1980s can be traced in part to the prototyping media used by American automobile companies. The intricate, elaborate, and expensive clay models sculpted by America's Big Three body designers (Figure 10.1) did not readily lend themselves to easy modification or rapid iteration. The work required to craft them made them more like untouchable works of art than malleable platforms for creative interaction.

>>>>>>INSERT FIGURE 10.1 ABOUT HERE<<<<<<

Figure 10.1 Prototypes in Clay The automobile industry has traditionally based the design of new models on mockups done in clay that give a highly polished impression of how the finished product will look, but do not invite further changes. (*Source:* Courtesy of the Arts Center College of Design.)

Based on a survey of the role of computer-aided design (CAD) tools in Japan, Daniel Whitney of the Draper Laboratories at MIT observed that, until recently, American car companies attempted to use the clay models as input for their CAD systems—a laborious and imprecise process. By contrast, Toyota did precisely the opposite: It insisted that its stylists design the car body with CAD tools from the beginning. The clay model became the output of the CAD system, based on the computer representations, such as those shown in Figure 10.2.

>>>>>>INSERT FIGURE 10.2 ABOUT HERE<<<<<<

Figure 10.2 Virtual Prototypes With modern tools for computer-aided design (CAD), much of the design of an automobile can be completed on-line, and can be visualized through high-quality graphics. The ease of visualizing changes greatly speeds the prototyping cycle and enhances the potential for fluidity in design. (*Source:* Courtesy of the Arts Center College of Design.)

When a clay model is the design *input*, it becomes a bottleneck to effective use of the CAD tools. Capturing in digital form the aesthetic essence of a clay model is difficult. When a clay prototype is the *output* of a computer model, modification and enhancement become comparatively quick and easy. Toyota can go from a new idea to a one-quarter–scale clay model in just 40 days.

The properties of these prototyping media undeniably shape the prototyping process. Although it would be foolish to blame the decline of General Motors on clay, it would be equally foolish to ignore the role that prototyping media play in determining the speed and quality at which automobiles are built.

The Questions that a Prototype Can Answer

Prototypes are designed to answer questions. The quantity and kind of questions that generate prototypes are at the heart of prototyping culture. Different questions may require different kinds of prototyping media—such as foamware or stereolithography. In software, different questions may lead to the use of different representations or even different languages. Note that sophisticated questions do not necessarily need sophisticated prototypes to answer them. Conversely, seemingly simple questions may defy even the most creative efforts to prototype. The questions that organizations choose not to ask are just as important as the ones that they do ask. This point is particularly relevant in software development, where each bit of functionality demanded may require a prototype.

One benefit that prototyping can offer is to generate new vocabulary for describing product features. When IDEO prototyped a toothpaste tube for a consumer-products company (see Figure 8.1), one of the questions that the prototype identified was what level of *suckback* the tube provided. *Suckback* refers to the way the tube pulls back the toothpaste into the tube after the user stops squeezing it. Prototyping enabled both designer and client to create a vocabulary and to calibrate the desired characteristic.

The Prototyping Cycle

The role of time in prototyping practice offers one of the clearest markers to distinguish among cultural priorities. At a moment when management wisdom insists that speed to market is the key ingredient for competitiveness, the time dimension of prototyping assumes greater significance. An organization that advocates product-development speed will have to attend to the speed at which it designs, builds, and tests prototypes. *Rapid prototyping* has become a buzz phrase of the nineties.

For example, Sony and 3M take great pride in the speed at which they can produce a functional prototype. Companies such as Microsoft and LSI Logic (a custom silicon-chip designer) also have cultures where the *mean time to prototype* is exceptionally short. Other companies—notably IBM, AT&T, General Motors, and Johnson & Johnson—tend to study a concept for weeks before even beginning to turn it into a prototype. The culture of the latter companies treats the prototype as an end product of thought, rather than as a partner with the functional specification in development.

Historically, companies go through several prototyping iterations before moving a product into production. Sometimes, several weeks elapse between prototype iterations; at other points in the development cycle, the delay can be months. As a general rule, companies with short mean times to prototype tend to generate more prototypes and to go through more prototyping cycles than do those with slower ones.

Many companies have evolved a methodology based on a fixed number of prototyping cycles. This number is as much a function of tradition and culture as of economic or competitive necessity. Industrial designers with whom I have spoken report that few companies go through as many as five prototyping iterations over 1 year. As a rule, the more prototypes and prototyping cycles per unit time, the more technically polished the final product (Figure 10.3)

>>>>>>INSERT FIGURE 10.3 ABOUT HERE<<<<<<

Figure 10.3 Multiple Prototypes The Microsoft mouse went through many cycles of prototyping before the designers settled on a final version. (*Source:* Courtesy of IDEO.)

The Players in the Prototyping Culture

The key elements of a prototyping culture are who gets to be a part of it and why. Nothing says more about an organization's culture than that organization's networks of power and communities of influence.

Who owns the prototype? Who manages it? Who gets to see it—and when? Who determines which constituencies have a say in the next prototyping cycle? Is there an internal model shop responsible for prototyping on demand? These are the questions that most starkly reveal the corporate prototyping culture.

At one highly regarded Silicon Valley company with a strong engineering culture, people are happy to show peers their bench prototypes, but avoid showing the prototypes to executives. Their edict is "Never show fools unfinished work." Good ideas may be rejected by ill-informed executives based on what is perceived as inadequate execution of the prototype. Top management may find it difficult to see beyond prototype roughness to the ultimate product. As a result, many engineers conceal provocative prototypes from senior managers until the models have been polished appropriately. In addition, a danger can arise from showing a prototype to an executive and finding that there is an order to ship the prototype even though it is not up to industrial strength. Here again, an ill-informed manager can make a wrong decision.

Demonstrations to senior management at times assume the character of theater performances, rather than of interactive dialogs among members of the team. In some cases, showing a prototype assumes all the logistical trappings and investment of a Broadway musical. The prototype becomes a medium for persuasion, rather than a vehicle to evoke discussion. It is used to prove a point, rather than to create a platform for a design dialog.

As a result of managers not having access to a dialog around emerging prototypes, top-manager insights in some organizations tend to come later in the design cycle than they should, thereby losing value. The later the time, the more likely that the top managers are being asked to approve—rather than to review or assist—new-product creation. Any organization that wants to gain a deeper understanding of its prototyping culture would do well to time how long it takes between the initial creation of a prototype and the moment that the prototype is first shown to senior management.

Ownership of the Prototype

If it is to succeed in its purpose, a prototype cannot be seen as the property of the engineers, of the developers, or of the marketer. It has to be community property. Traditionally, prototypes have been a weapon in interdepartmental power struggles. At high-tech companies, for example, engineers and technicians traditionally own the prototype, and bicker with marketing and manufacturing over suggested modifications. At packaged-goods companies, the brand manager typically owns the prototype. In other cultural contexts, prototypes are little more than sales tools and technical stalking horses for the politically adept. The politics of prototypes play a large part in shaping their potential value.

These politics extend beyond company walls. Organizations must ask themselves, "When do customers and suppliers get to see the prototypes?" Indeed, do customers and suppliers participate in the prototyping process at all? At some companies, showing outsiders the prototypes is a widespread practice. At others, only senior management can authorize such displays. Both extremes have their problems. On the one hand, pre-release familiarity may take away the excitement from release (and ideas may even be stolen); on the other hand, developer isolation can lead to developer arrogance and ignorance.

Collaborative prototyping with customers and suppliers can yield competitive benefits. Nike's successful creation of a new product design language (see Chapter 4) stemmed from intense collaboration between athlete and designer. IBM attributes much of the success of its AS/400 minicomputer to participation by key customers. Microsoft's collaboration with applications-software companies was an integral part of the widespread acceptance of Windows. More generally in the software industry, no successful piece of personal-computer software is launched before potential customers interact with alpha- and beta-version prototypes.

As part of an evolution from technology-driven products to customer-centered design, the customer becomes an important member of the prototyping community. The customer must have the opportunity to see and try the prototypes as they evolve. Customer involvement has been the key to the success of companies such as Intuit (see Chapter 13), which solicited extensive customer feedback in developing its home financial product, Quicken (see Profile 13).

Changing a Company's Prototyping Culture

If we take seriously Kelley's claim that organizations intending to be innovative need to move from specification-driven prototypes to prototype-driven specifications, then we need to look at the ways that change can occur, and at the ways that change is already underway.

The great ethnographies of prototyping cultures have yet to be written. However, it is demonstrably clear that fundamental differences in corporate prototyping cultures lead to qualitatively and quantitatively different products. Understanding those fundamental differences is essential for any organization that wants to transform its new-product development. A shift in the prototyping practices is a necessary part of a shift in the culture of the enterprise, and, in the end, a shift in that enterprise's role and values in the market. Companies that want to build better products must learn how to build better prototypes. Change initiatives that do not deal explicitly with the culture of prototyping are initiatives that ignore organizational reality.

I see two strong cultural shifts gradually taking hold in both the industrial-design and software communities:

1. The implicit belief that a structured innovation process drives prototype development is yielding to the belief that emerging sequences of prototypes drive the innovation process.

2. The notion that innovative teams generate innovative prototypes is giving way to a recognition that innovative prototypes are the focus for generating innovative teams.

The innovation process

The move toward a prototype-driven innovation process is most evident in those companies that have instituted repeated quick prototyping. The increased emphasis on speed to market has produced a practice that is radically redefining prototyping culture, called *periodic prototyping*. Instead of producing prototypes when design teams think that doing so is appropriate, many time-sensitive organizations are now institutionalizing the prototyping process around explicit schedules. The result is that developers are now forced to double or even triple the number of prototyping cycles per unit time. Quantitative changes inevitably lead to qualitative changes. At Honda, for example, an automobile might go through 10 or 12 trial builds, with a new prototype being built every 2 weeks. Motorola required a similar periodic prototyping schedule for its popular Bandit pagers.

By comparing the prototype changes per cycle, management now has a rigorous vehicle for measuring progress. Designers who are held to periodic prototyping schedules are likely to become more prototype driven than specification driven.

The innovating team

As companies push toward cross-functional development teams, the role of the prototype can change. When the community of stakeholders in the development of new products and processes broadens, physical objects that help to bridge disciplinary and functional boundaries become more important.

In essence, prototyping becomes not only a medium for interdepartmental integration, but also a medium for organizational redesign. Prototype-driven innovation ends up promoting a radical deconstruction of the existing organizational charts. It becomes increasingly important to avoid the departmental turf wars that cripple prototypes, which in turn demands a fundamental change in the prototyping culture.

Indeed, mapping the flow of prototypes through the enterprise is one of the most significant exercises that an organization can undertake. Who—insider and outsider—gets to see what when? When are modifications made? Who requests them? Which requested modifications are ignored? This map—rather than an organizational chart—can be the best starting point to evaluate core process redesign. Such a map reveals how essential—or how marginal—the formal prototyping process truly is.

Precisely because a prototype is a concrete artifact, it can be meaningfully tracked and measured. An organization can quickly discover the power points and political bottlenecks that govern the value-creation process.

Conclusions

Organizations can seek to revise the dialog between specifications and prototypes; they can pick prototyping media with desirable properties ranging from flexibility to cost; they can reset the sense of time for prototype management; they can create prototyping communities; and they can change their prototyping culture.

There is no one right answer to "How should we prototype?" Prototyping strategies are varied, and a company needs to develop a mix that serves its markets and its products. Prototypes are as much a medium for managing risks as they are a medium for exploring opportunities. They can be treated as an insurance policy or as an option on the future.

Ultimately, one of management's greatest challenges is to integrate its portfolio of prototypes effectively into an integrated product or family of products. That is the arena where politics, economics, and organizational culture are often in sharpest conflict. Effectively answering that challenge will not only display the organization's cultural priorities, but also will define the organization's ability to innovate effectively in an increasingly competitive marketplace.

Suggested Readings

Michael Schrage. *No More Teams.* New York: Doubleday Currency, 1995.

Steven Wheelwright and Kim Clark. *Leading Product Development: The Senior Manager's Guide to Creating and Shaping the Enterprise.* New York: Free Press, 1995.

About the Author

Michael Schrage is a technology columnist for national newspapers and the author of a book on groupware, *No More Teams* (1995). He is a research associate at the Sloan School of Management at MIT, and has been a visiting scholar at the MIT Media Lab.