

FROM EXPERIENCE: Products Talking to People— Conversation Closes the Gap between Products and Consumers

Aaron Oppenheimer

Product design is a complex undertaking, requiring manufacturers to find a balance among business goals, regulatory guidelines, the requirements of the distributor and retailer, and consumer demand. Along the way, it is sometimes easy to forget that, in the end, an actual person will attempt to operate the product.

Manufacturers are skilled at collecting and analyzing data about their customers and at using this information to formulate a “brand message”: those things a company wants to communicate to consumers, the expectations that should be set in the consumer’s mind, and what feelings and associations consumers should take away from an interaction with the company and its products. But while this information may be useful in understanding demographic and economic aspects of consumers, it often does not help the manufacturer understand how the product and user should interact—how the product and user should “talk” to each other. Something is lost in translation as a product goes from being a conceptual instantiation of a brand to being an actual physical object that a consumer must communicate with and must manipulate. The result is an experience that reflects poorly on the product and the manufacturer.

This article presents a framework for developing the interaction between user and product, using the metaphor of conversation between people. Human beings have developed a set of conversational rules and norms over the last few thousand years, and by leveraging these standards, manufacturers can create product interactions that provide a better experience to the consumer.

Manufacturing companies spend vast sums of money to develop their “brand message”—those things a company wants to communicate to consumers, the expectations that should be set in the consumer’s mind, and what feelings and associations consumers should take away from an interaction with the company and its products. Companies seek to present a unified face to consumers, with consistent messages communicated through advertising, corporate policy, and, of course, their products. The brand defines—and in large part is

defined by—a company’s product line. Consider companies producing digital cameras: one may develop products for use by professional photographers, while another aims its products at consumers with no prior experience. Still another may produce products for both but may use a special moniker (say, “pro line”) for professional products. In each of these cases, the product features, advertising messages, and customer service support are tuned carefully to communicate a specific brand message—“this is the company and product for you”—to different types of users.

Marketers long have thought about the message being sent to consumers through print and television advertising, but a consumer’s view of a brand is shaped not only through these communications but

also through experiences with the product itself. A positive interaction that delivers fully on the promised value of a product can reinforce a brand message like nothing else.

So, then, with all of this careful crafting going on, why are so many products difficult to use? If a product is meant for the average consumer, why do so many average consumers have difficulty with their products—especially high-tech, computerized products, which would seem to have more flexibility in their design than mechanical products? Certainly, no company purposefully is creating products that are frustrating or confusing; “frustrating and confusing” is not part of anyone’s brand message. Nevertheless, one of the more enduring symbols of modern technology is the video cassette recorder (VCR) continually flashing “12:00” because its owner cannot figure out how to set the time.

The problem often is to be found in the space between concept and execution. The idea may be “Let’s create a digital product that is so easy to use that anyone can operate it,” but this may be in conflict with the realities of product development: the screen must be small to save cost; obscure functions must be added to make the product appear “competitive” to retail buyers trained to expect long lists of features; the user interface must function the same way as legacy products to save development time and cost. Balancing all of these issues (and innumerable others) is the hard work of product behavior design, and unless trade-offs are made carefully, the product will suffer.

Product behavior designers use a variety of frameworks to organize the requirements and information that go into the design process. These tools assist the design team in understanding the priority of various elements of a product and its interface, and the design of the framework itself influences the final “look and feel” of that interface.

This article presents a framework that often is useful when designing product user interfaces. Just as

people relate to other people, consumers form relationships with their products, and designers can leverage concepts in people-to-people associations when designing people-to-product associations.

The Conversation

Human beings talk to each other endlessly. We exchange pleasantries on the street, hold long meetings, debate, argue, convince, explain, and spend a large portion of our days conversing with each other. Children learn how to communicate effectively using the social rules that we all have developed over the course of thousands of years. By the time most people are two years old, expectations have been set and norms understood: kids know about greeting, conversing with, and saying goodbye to other people.

In addition to interacting with each other, we interact with products: alarm clocks, electric razors, cereal boxes, refrigerators, and toasters—all before breakfast! The more complex of these devices have user interfaces that might include buttons to press, dials to turn, displays to read, or beeps to hear. Often, despite a manufacturer’s best efforts, these interfaces are complicated and confuse the consumer. Hard to learn, to understand, and to navigate, these user interfaces block a consumer’s access the functionality and benefits promised by the product and the company that manufactured it.

Confusion over a product’s behavior is often the result of mismatched expectations—the user believes a product operates in a certain way, but the product does not work that way. Chief among the causes of this mismatch is the message sent by a product about how it expects to be operated. All products send messages about what and how they do their thing, explicitly or implicitly, and they often either are misinterpreted by the user or are just the wrong message to send. An example that we all have encountered is the door with a handle that seems to say “Pull me” but that actually requires one to push. Avoiding such miscommunication is key to creating interfaces that successfully delivers the functionality promised by the device.

The problem of products sending bad messages about their own behavior can be solved by designing messages that users can understand, delivered in ways users are accustomed to receiving them. Designers can leverage the rules and social norms that people have spent years learning (or, indeed, have evolved to appreciate) and can apply them to product interactions.

BIOGRAPHICAL SKETCH

Aaron Oppenheimer leads product behavior development at Design Continuum in Boston, Massachusetts. He has worked with a wide variety of clients, including Hewlett Packard, Sprint PCS, Polaroid, and General Motors, and writes and speaks on interaction design topics in various forums. Mr. Oppenheimer’s experience includes design for medical monitoring equipment, vehicle dashboards, knowledge management applications, and music composition software. He holds a degree in computer engineering from Carnegie Mellon.

The idea is to consider the design of an interface to be the design of a conversation between the consumer and the device and to tune the behavior of the product to make appropriate messages an explicit part of that conversation. This makes the process of organizing interactions between a person and a product a matter of observing conversational rules. Just like people can be “polite” or “rude,” products can also appear “polite” or “rude” depending on whether they work with or against patterns of conversation. Polite products hold polite conversations.

Conversational Patterns

Two people engaged in a conversation follow predictable, standard patterns. A greeting is followed by a response. Questions are answered. The response to “goodbye” is “goodbye.” One who fails to follow these patterns is a bad conversationalist.

At its simplest level, the conversation between a consumer and a product resembles the conversation between an employer and a candidate for a temporary job, as seen in Figure 1. This is not an arbitrary metaphor. Consider the process of stapling two pieces of paper together: one looks around for an appropriate tool, evaluates the tool for its suitability, applies it, and evaluates the result, as in Figure 2.

First, the consumer needs to understand what the product does, which he will do by examining it. The product must “respond” with an explanation, via either an explicit message or an implicit one based on its similarity to products with a similar function. In other words, a stapler either must boast a large label reading “stapler,” or it must look like other staplers. The consumer tells the object to perform some task by manipulating its interface in some way, and it responds with status to let the user know that the job is underway or complete, which also may be explicit or implicit—if the paper clearly is stapled together, the stapler need not announce “Stapled!” through an on-board loudspeaker.

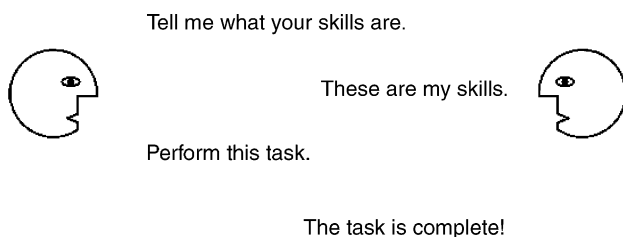


Figure 1. The Conversation between Two People Follows Predictable Rules

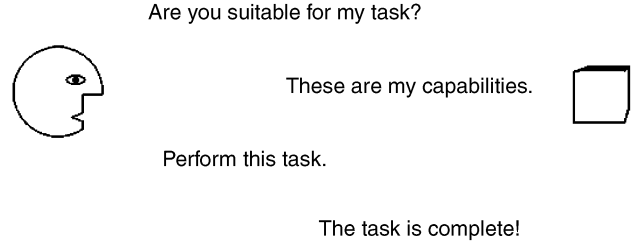


Figure 2. The Conversation between Device and Consumer Is Equally Predictable

This four-step process may seem simplistic, but it is important to recognize the difference between the mechanics of an interaction and the content of that interaction. Whether using a stapler, a calculator, or an air traffic control station, the steps a consumer goes through to explore and to command an interface can be broken down in this same manner.

For another example, think about someone operating an unfamiliar microwave oven. On approaching the device, a user locates the controls, paying special attention to controls for timing, for turning the oven on, and for opening the door. This is the “What do you do?” phase of the conversation, where the user is trying to figure out if the device will really do what she needs. Once the controls have been located, the user attempts to understand how they work—for instance, the order in which they must be pressed. This is the “Here is what I do” phase, where the device must explain the manner in which it is controlled to the user. This is a common point of communication breakdown in the conversation—the user has located the controls but cannot figure out what to do with them. The user moves into the next phase, “Do this,” commanding the oven to heat up the food, and the oven eventually lets the user know that the mission is complete, with some “Here is what I did” feedback—typically a sound or flashing light.

Product interactions usually can be broken down into these four stages, which we call here *announce*, *explain*, *act*, and *notify*. Using this framework as a tool in the design process can help manufacturers and designers get a better handle not only on the structure of a product’s user interface but also on all of the ways the product communicates with consumers.

For Example

In 2000, Sunbeam and Design Continuum began a project with the goal of revitalizing Sunbeam’s



Figure 3. The Oster In2itive Blender, designed by Sunbeam and Design Continuum (It features a “results-oriented” interface; instead of a row of buttons that simply spin the blades at different speeds, the buttons on the In2itive are labeled with textures of food and drink. The blender runs a different program of blade spinning, pausing, and timing to deliver a consistent result every time.)

blender product line. Blenders are an old product with high penetration in the marketplace, so the joint team faced several issues such as getting consumers interested in using blenders for more than just drinks, creating a blender more responsive to the desires of a new generation of consumers less involved in their kitchens than generations past, and how differentiating a product in a very crowded market. The result was the Oster In2itive Blender (Figure 3), launched in 2002, with new versions entering the market today. This blender changes the paradigm of blenders. Instead of a simple tool, the In2itive acts as an assistant; buttons on the face of the device describe specific textures instead of blade speeds. When pressed, each button initiates a different program of blade speeds, timings, and pauses that deliver a consistent end product every time. The interface for this product had to communicate this new use model without disorienting or alienating consumers, and so I use it here as an example to illustrate the conversational interface concepts as we go along.

Announce: What Are You?

The first stage of a conversation between people—even before the talking starts—is a getting-to-know-you period. When one meets a stranger, there are a multitude of first impressions collected, compared, and correlated. What does he look like? Who does he remind you of? Does he have an accent? Do you

have friends in common? These things, and 100 others, happen in the first few minutes of a relationship and can set the tone for the rest of the interaction. It is trite but true: one never gets a second chance to make a first impression. Even with an old friend, a conversation starts with a reacquaintance and reassessment: “how have you been?”

When meeting a product, the same process takes place: manufacturers strive to get across the “concept” of a product—what it is, what it is for. The shape, color, and markings on a product can communicate its purpose, and many manufacturers are good at manipulating a consumer’s perceptions to convey brand messages. However, manufacturers are less adept at communicating how a product works and what it does, and the getting-to-know-you concept is just as important here. Consider renting an unfamiliar car. As with meeting an unfamiliar person, the first task is to familiarize oneself with the new friend: in this case, finding and learning about the controls on the dashboard. Clearly, its use is obvious—that is why the car was rented. But where are the gauges? Where are the headlight and wiper controls? A well-designed car gives some assistance by grouping the controls together in one or two places. Automotive designers use various techniques to highlight controls and gauges clearly through variations in form, color, and materials. This is the *announce* phase of the conversation, where the product is revealing itself to the user. Products need to make it clear to users where the controls and feedback are located and should strive to deemphasize those controls the user probably does not need—a reset switch, for instance, used only by a repairperson.

One powerful tool for forming helpful first impressions is to leverage the fact that consumers come to any product interaction armed with a collection of preconceived notions about what a product is and does based on prior encounters with similar products. We all know what a car is, and a product can communicate a lot of information about itself by relating itself to these preconceptions. If it looks like a car, people will treat it like a car and should expect it to behave like one.

For example, most people are used to the big analog gauge behind the steering wheel—it is the speedometer. The analog dial gives lets the user know how to control a car he has never seen before, because it is just like the ones she has used before. On the other hand, if a product does not provide what it is promising, the user can be confused. Although digital



Figure 4. The Dashboard of the Mini Cooper (It places the speedometer in an unusual place: the center of the dashboard. This makes a definite design statement—"I am not just another car." However, it could require extra effort on the driver's part to find and to use the gauge, and in at least one country, the manufacturer has had to move it to the usual spot to conform to safety regulations. Photo courtesy of MINI USA.)

speedometers have been around for years, manufacturers do not use them in most cars because they do not say, "I am a speedometer" to the car-buying population. And a consumer knows, indubitably, to look behind the steering wheel for the speedometer—a car that puts it in a nonstandard location, as in Figure 4, forces the consumer to spend extra energy to locate and to use it.

Referencing other products is one way to communicate the "concept" of a product. Successfully communicating the concept gives the user a sense of comfort, an "Okay, I know where to start" feeling. Although designers often think of "first impressions" only around the way a product looks, the concept certainly applies to how the user thinks the product will behave. For example, TiVo borrows from video recorders of the last generation, using terms like "rewinding live television" and the VCR-like "digital video recorder" (DVR). A TiVo is not a VCR—it uses completely different technology, which could have allowed its designers to invent an entirely new mode of interaction. They did not, though; keeping certain aspects of previous products helps ease consumers into the new technology.

Although the Oster In2itive blender uses an entirely new model for interacting with a blender, it was imperative to use "cues" common to standard blenders in order to reassure consumers that this product, the Oster In2itive blender, is indeed a blender. The buttons and lights on the interface are typical of products in the category; it is not until closer examination that the consumer discovers the difference (discussed in the next section). The key is for consumers recognize that the product is a blender; this recognition sets certain expectations about what the product does and what it

is for, and these expectations form the basis for the overall use of the device.

Evaluating the success of the *announce* phase can be done at early stages in the design process. Even showing a simple sketch of a product to a representative consumer or two can throw light on the issues. Ask these questions:

- What is this picture of?
- Can you identify the controls?
- How do you think you turn it on?

The answers to simple questions like these get to the heart of the basic understanding formed by a user. The fewer details on the sketch, the more visceral the feedback you will get. Remember, at this point in the process, you are after first impressions and an understanding of how those impressions align with the messages the product is supposed to be communicating. Once appropriate initial impressions are formed, the product will be more effective at communicating the details of its operation later on.

Explain: Here Is What I Do

Getting back to our conversation with a new friend, consider the first few questions you might ask. "What do you do?" "Where are you from?" "Why are you here?" These are the questions one asks in the process of forming attachments to an acquaintance and are a typical part of a conversation. Similarly, interaction with a product can be seen as leading from an initial "What are you?" stage to a second "What do you do?" stage.

After establishing what a product is and does, the second step for the user is to understand what functions the various controls enable. For example, consider the turn-signal control pictured in Figure 5. The turn signal stalk in this car is in the usual spot, to the left of the steering wheel. It works as expected—move the control up to signal a right turn and down to signal a left. But this stalk also has controls for several other functions: windshield wipers on/off, windshield wiper speed, windshield spray, and headlight high-beams on/off. These functions are explained through a combination of text labels ("push," "delay," and so forth) and a variety of icons representing the windshield wipers and headlights. When a driver encounters this stalk during the *announce* phase, it will communicate, "This is a turn signal, plus some other stuff." The point of this next phase is to define the "other stuff" for the driver.

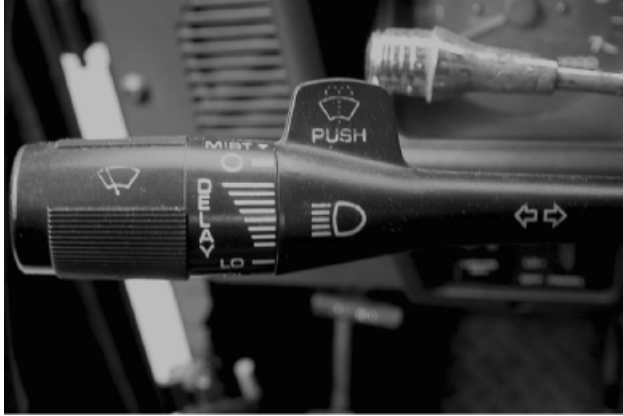


Figure 5. Control for Turn Signal (and Other Things) (Combining functions reduces the number of controls, but may take longer to learn to use.)

This is the *explain* stage of the conversation. The consumer knows what he is looking at and now is trying to figure out the details of how to make it do its thing. This stage of course is key to the consumer's understanding of what the product actually does. Just as with a human conversation, consumers probe around, getting a feel for what all of the "topics of conversation" are. It is important for a product to support this exploration either by clearly explaining itself or by allowing the user to try things out without experiencing negative consequences. Our turn signal example uses icons to label various functions; if those icons do not get the message across, there is not much downside simply to prodding the various functions to see what they do. This may not be true of, for instance, an ignition switch, which should not be turned except actually to turn the car on or off. This notion of *exploration* is important in the design of user-product interaction in order to give the user a feeling that nothing can go too badly wrong if the user just pokes around a bit. The "undo" function of many software applications provides not only the ability to recover from mistakes but also to experiment with unfamiliar functions—if the result is unpleasant, the user always can undo.

The *explain* step is the foundation for all future interactions with a product. If the device does not perform as expected, the consumer will come back to this step, demanding to know what went wrong, asking something like, "Well, if that was not it, which button is the one I want?" Unfortunately, this stage is often the one given shortest shrift by manufacturers. Often, "universal" icons bear the burden of explaining abstract concepts. Clear labels in the local language al-

most always work better, but the cost of and stocking products labeled in multiple languages for use in several countries can be seen as offsetting these ease-of-use gains. When this trade-off is made without real data—say, a basic usability test showing that the icons are effective—the user may miss out on the functionality promised by the device and its manufacturer.

The *announce* and *explain* phases are not separate. Based on our first impressions, we employ a set of questions "tuned" for these perceptions. For example, if I believe, based on the *announce* phase of a conversation with a new friend, that she went to the same high school as I did (maybe she's wearing a "Brighton High School" sweatshirt), I might ask her some questions about growing up in Rochester, New York. I would ask a different set of questions if she seems to be from New York City or from Paris. Similarly, a consumer who believes that a device is a lamp will look for typical lamp controls—a button near the bulb or a switch on the base of the lamp. If the product says "typical lamp" but uses a nonstandard control, like a handclap-activated switch, the consumer will be stymied just trying to turn it on.

The controls for the Oster In2itive blender are organized to communicate its functionality. Unlike a standard blender, with speed controls lined up from left to right, this product must communicate its use in creating textures, rather than just running the motor. Food textures are clustered on one side; drink textures on the other; organizing the controls this way conveys the idea that this product is more about the end product—the food being produced—than about the process. This may be a bit of a shock to the user, since we have set expectations in the *announce* phase that may be broken here. However, though the controls are unfamiliar, they are undeniably controls, and the product is undeniably a blender. The effect is similar to that of finding out that someone has a foreign accent: a few expectations change, but the overall relationship does not.

As with the *announce* phase, the effectiveness of a product's performance of the *explain* phase is simple to evaluate. When a user has a sense of what a product is, simply ask a few more questions:

- What are all the things this product does?
- How do you control it?
- What would you do to perform the main function?

The answers to these questions tell the designer how well the device has explained itself to the user. Note

that it is not necessary for the consumer to have a clear “like this”-style answer; for many products, it is acceptable for the user to have a process in mind for exploring the product. In fact, for more complex products, like a high-end washing machine or a piece of software, it is not reasonable to expect a user to understand all of the features within the first few minutes of acquaintance with a product. It is enough to have an attachment, a feeling of “I know pretty much what this thing does, and I am comfortable relating to it.”

Act: Do This for Me

Once one is comfortable with speaking to a new acquaintance, the next step is to get some work done. The impressions and explanations set up in the *announce* and *explain* stages set the ground rules for a working session; the ways in which it is possible to communicate have been established, and the parties are ready to have a real conversation. The contents of a conversation between people are typically an exchange of information, and the same holds true for conversation between people and products. At its core, a product’s user interface is all about information exchange. Pressing a button gives the product information about what the user wants done; a flashing light gives the user information about what happened. These two exchanges happen again and again and make up the final two phases of our conversation.

During the *act* phase, the conversation has progressed to the point where the consumer is ready to tell the device to do something. She has achieved a level of familiarity with the product that gives her the confidence to flip a switch or to move a latch, with the understanding of what should happen as a result (what actually does happen, she does not find out until later).

Assuming that the consumer has gotten good information from the *announce* and *explain* phases, he should feel comfortable giving a command—but how will he know she has been heard? Again, viewing user-product interaction as a conversation gives clues for enabling proper behavior. Imagine having a conversation with someone who fails to acknowledge what you said. You may wonder, “Is he ignoring me? Perhaps he did not hear me? Should I say it again? Did I do something wrong?” This is not a very comfortable position in which to be and is the reason a product must acknowledge commands from the user: to show that it is listening.



Figure 6. “Walk” Buttons Do Not Acknowledge Actions Like Elevator Buttons Do

Consider the difference between an elevator call button and a street corner “walk” button, as in Figure 6. The elevator button lights up when pressed, telling the user (and anyone standing nearby) that the system heard the command and that the elevator will arrive presently. The “walk” button does not do anything to acknowledge an action, leaving pedestrians to wonder if it is working at all.

The style and level of feedback required depends (as does everything) on the context in which it will be used. If a request will be answered quickly, the “I am listening” message even may be implicit. For example, if I ask someone for change for a dollar, and he immediately reaches into his pocket and pulls out some coins, I know that he heard me; the feedback is implicit in his actions. On the other hand, if he needs to think for a few minutes about whether or not he has the coinage, it would be helpful for him to say, “Let me think about that for a minute”—otherwise, I will likely assume he has not heard me and will ask again, louder.

As in the human-human world, feedback is also important in the product world if the requested action is going to take a while or otherwise not be obvious. E-commerce sites on the web are a great source of examples. Many sites have a prominent “Click once!” label next to the “Pay Now” button, as in Figure 7, because clicking the button twice may cause the customer’s credit card to be charged twice. However, because of the delay between the time a user clicks and the time the system responds, she may become concerned that the site did not hear her and may click again. Warning the user to click only once seems to be a simple solution, but for the user, knowing whether



Figure 7. Click Once, or You May Be Charged for Every Click

she actually has “clicked” or not is the very problem. Some sites change the “pay” button to a “wait . . .” message when the button is clicked, which alleviates the problem to some extent. It is better to design a system to respond to a click with feedback as quickly as possible, even if that means deferring time-consuming tasks (like verifying the credit card numbers) until a later stage in the process.

Acknowledgement of a command during the *act* phase comes in many forms: a sound, a light, even a “feel,” as when a lever or knob clicks into position. If I send the wrong document to a printer, I may press the “cancel job” button on the front of the device. A few more pages get printed, which might make me think the “cancel” button does not work—so the manufacturer may put a light next to the button that blinks when a job is cancelled. Although some paper continues to come out, I know that the printer has “heard” my command to stop, and I know that the cancellation is in progress.

The In2itive blender uses a row of lights that act as a progress indicator to tell the user how the process is going. This mechanism indicates that the user has been heard (as does the sound of the spinning blades) and how much longer the program will run. With a product that uses an unfamiliar interaction style, a progress indicator helps to explain what is going on—in this case, that the blender will run its program and then will reach the desired texture. The progress indicator tells the experienced user how much time a program will take, freeing him up to do something else during the blending process.

Feedback is an important concept because it gives the consumer continued confidence that he is doing the right thing. The efficacy of feedback is tested easily with a few questions. When a user presses a button or otherwise manipulates a device, we want to know

- What did the user intend to do?
- What does the user think he did?
- Did it work?

The point here is to understand what is going on in the user’s mind. If what the user thinks he did is not what he actually did, the designers must understand the way in which the device has misled its operator. What effects the user thinks his actions have on a device are just as important as the actual effects, and the point is to design feedback so that what the user thinks is happening and what actually is happening are the same thing.

Notify: Here Is What I Did

There is nothing more uncomfortable than two conversationalists staring meaningfully at each other, wondering, “Are we done here? Anything else to say?” The final stage of a polite conversation between people is an acknowledgement that, yes, we are done. In the product world, the situation should be the same—the user needs a message from the product that her command was carried out and the results are available or a message that the command was not carried out with some explanation of why. This is the *notify* step in our conversation: a message that the job is done. Sometimes, acknowledgement is obvious—for example, if a driver turns the headlights knob and the headlights come on, the job is done, and she knows it. For many devices, though, notification is not instantaneous and must be done explicitly by the product.

When using a digital thermometer to measure one’s temperature, for example, the consumer does not know necessarily when the device is showing its result, since he cannot see the display. This may cause him to wait until the device has been in his mouth longer than necessary, robbing him of the main advantage of a digital device. Although the digital thermometer is much faster than its mercury-based counterpart, this consumer probably is waiting a mercury-based length of time before looking at his digital product. A simple beep can let the user know that the temperature reading is correct. An additional way to think about *notify* is as the close of the conversation. This has two important aspects in the person-to-person world that carry over to the person-to-product world, namely “bye” and “until we meet again.”

The first of these is a message to the user that she has completed the interaction successfully. This does not mean necessarily that the job got done—maybe

there was an error, or her credit card was rejected, or something else stood in the way. But even in these cases where the content of the interaction is not satisfactory, the interaction itself is completed properly. For example, when a customer calls a clothing retailer with a phone order, and her desired clothing is out of stock, the interaction with the operator still can be pleasant and satisfying. Unfortunately, many products are not designed to end their interactions as gracefully—imagine an operator shouting “out-of-stock error!” and hanging up the phone.

The other aspect is akin to setting a date for a future interaction. Generally, the goal is for a user to continue to use a product over time, and there are several ways to encourage this during the final stages of an interaction. One is to offer to remind the user to return by setting an alarm, by sending a reminder email, or by having a real person place a phone call. These serve to extend an interaction into a longer-term relationship.

The In2itive blender stops when a blending program is complete, which is a clear indication that the process is finished. But more than that, the labels on the face of the interface serve to point out to the consumer other uses for the product. Someone using the blender to mix a smoothie sees that the product also can make salsa—the interface serves not only to control the product but also to advertise its uses and to invite the consumer back for another session.

As with the other steps, the effectiveness of the *notify* step is tested easily using simple prototype devices (even sketches on paper suffice). When a user directs a product to perform a step, it is important to know what he expects to happen and whether what actually happens meets his expectations:

- In the user’s mind, what should happen next?
- What does the user think actually happened next?
- Is that what the user expected?
- Is the user done with the task?

These are the sorts of questions that continue to illuminate the user’s understanding of the device. As with the *announce* and *explain* steps, the *act* and *notify* steps are linked together and to the prior steps; the expectations held by a user about what he told a product to do and what should happen as a result must match what actually happens, or the user will be confused. Each step in the conversation builds these expectations and so must be in alignment for the experience to be satisfactory.

The Burden of Conversation

As a good conversationalist, I put effort toward being understood. If it becomes clear that my opposite is misinterpreting me, I will explain myself until I am sure we are on the same page.

Products generally do not have the luxury of detecting and correcting a misunderstanding, so they must get their communication right the first time. The burden of carrying the conversation is *always* on the device, not its operator. A communication failure—the user cannot find the controls, does not understand them, pushes the “go” button twice, or does not know when the task is finished—should always be considered a failure on the part of the product. Manufacturers often cite human error as the cause of a failure, but although errors in judgment may play a role (such as when a driver does not look both ways before venturing into an intersection), a failure to understand how to use a device should not be considered the user’s fault.

Of course, this notion—that the user is always right—depends on the definition of the user. One would not expect to have a conversation in French with someone who did not understand French, nor should a manufacturer expect someone not trained in the practice of giving anesthesia to operate anesthesiology equipment successfully. But in the same way it behooves someone traveling to France to learn to at least say, “*Je ne parle pas français*” [“I do not speak French”], product developers must consider the ways in which it makes sense to communicate messages to those not trained to use the product, such as “hands off,” “danger,” and so forth.

A warning sign, as in Figure 8, is a conversation all by itself—a sign must announce its presence, must establish its authority, and must communicate a message, and just like the rest of the functionality of a device, the efficacy of a warning message depends on its ability to be understood. The standard attributes of warning signs—red color, bold type, iconic pictures of bodies undergoing painful experiences—are, at their root, tools for holding a very specific type of conversation.

One last point on the product’s responsibility is that over time, people get used to repeating the same conversations. I might say, “Good morning, how are you?” to a tollbooth attendant every morning, and after a while, it will become automatic. I will not really be listening for a reply—I might not even be aware that I said anything. Products must be mindful of this



Figure 8. Warning Signs Are Miniature Conversations in Themselves (They must work to engage the consumer and to find ways to communicate complex concepts, often to people with little domain knowledge.)

fact of life—someone who uses a product every day will get accustomed to using it. Experience with a product is a double-edged sword: although it makes the user of a product more comfortable, it also can allow her to operate a device with less concentration than is warranted. People become inured to warning signals and signs, and the burden is on the product to ensure continuing communication even when a user is

very experienced. Warnings cannot “cry wolf” but must draw attention when it truly is required.

A Good Conversationalist

When speaking with another person, I find that following the standard rules of conversation allow me to be understood more easily and to provide a better experience for the other person. When designing a product interface, I follow similar rules to ensure communication of the product’s concept and usage. Manufacturers spend large amounts of money to project particular brand images, like “caring,” “clear,” and “helpful” in their advertising and packaging—it is only logical to create products that convey these same feelings by providing good experiences for consumers.

The “product use is a conversation” framework helps organize a product’s features and behavior around simple goals: communication, efficacy, and positive experience. Using this and other tools can give a designer a better look into the user of a product and a better understanding of what a product can be.