The Complete Reference

Web
Desig

Chapter 2

User-Centered Design

As discussed in Chapter 1, Web sites are often developed from one particular philosophical reference point. Sometimes this point of reference is content centered; other times, it is technology-centered. Even more frequently, it is graphics-centered. However, the real emphasis when building sites should always be the user. Keeping users in mind and always trying to meet their needs should be the key focus of user-centered design.

Understanding users needs isn't easy. While users may share common capabilities such as memory or reaction time, each user is still a distinct individual. Sites should be built for common user capabilities, rather than for the extreme novice or power user. Sites should be accessible to all and be able to account for the differences exhibited by individuals. Building a usable Web site is challenging, since what is usable to one person may be problematic for another. The likelihood of building a user-centered site is greatly improved through user interviews, testing, or even iterative design. Always be wary, though, of falling into the "user trap." While a site should always be built for users, the desires of the site's creators must also be met, even though these may be somewhat at odds with the desires of the site's users. The fine balance of power between user and designer is not always easily achieved.

Usability

Everyone has a vague idea of what it means for something to be *usable*. People will talk at length about how Web sites are supposedly user friendly, intuitive to use, or simply "usable." What, exactly, does it mean for something to be usable? First, consider the concept of *utility* in connection with two e-commerce sites that sell books and offer the same basic features. Both allow the user to search or browse for books, read information on books, purchase books, and track their orders. If both sites have basically the same features, they have the same utility—meaning they can do the same thing. Given that the sites have a few basic functions, you may find it easier to perform the same task on one site than the other. In this case, we can say that one site is more usable (has greater utility) than the other. Unfortunately, it is difficult to agree on what is usable. Plenty of people have attempted to characterize what usability is. Consider the following definition adopted from an ISO standard definition of *usability*:

Definition: Usability is the extent to which a site can be used by a specified group of users to achieve specified goals with effectiveness, efficiency, and satisfaction in a specified context of use.

Consider each piece of the definition. First, note that we should limit the group of users when talking about usability. Recall that usability will vary greatly depending on the user.

Next, usability should be related to a task. You should not consider a site to be usable in some general sense. Instead, discuss usability within the context of performing some task, such as finding a telephone number for contact, purchasing a product, and so on. Usability is then judged by the effectiveness, efficiency, and satisfaction the user experiences trying to achieve these goals.

Effectiveness describes whether or not users are able to actually achieve their goals. If users are unable to, or only partially able to, complete a task they set out to perform at a site, the site really isn't usable. Next, usability is related to efficiency. If users make a great number of mistakes or have to do things in a roundabout way when they visit a site, the site isn't terribly usable. Last, the user must be satisfied with the performance of the task.

Many other definitions of usability exist. Some usability professionals suggest that usability can be concretely defined. Maybe it could be computed as some combination of the completion time for a typical visit and the number of errors made during the visit. From the user's point of view, that might not mean much. Users might just be concerned with how satisfied they were after performing a task. Many usability experts, such as Jakob Nielsen (http://www.useit.com), tend to have similar definitions more in line with the ISO one. For example, Nielsen suggests that the following five ideas determine the usability of a site:

- Learnability
- Rememberability
- Efficiency of use दक्षता
- Reliability in use विश्वसनीयता
- User satisfaction

Jakob nielson

By this definition, a site is usable if it is easy to learn, easy to remember how to use, efficient to use (doesn't require a lot of work on the part of the user), reliable in that it works correctly and helps users perform tasks correctly, and results in the user being generally satisfied using the site. This still seems fuzzy in some ways, and conflicts arise easily in the usability area. For example, a site that is easily learnable by a novice user may be laborious to use for a power user. Because people are different and come with different levels of capabilities and Web knowledge, not everyone is going to agree on what is supposedly usable. A site that is easy to one user may be hard for another.

Rule: There is no absolute description of what constitutes a usable site.

Even without considering user differences, we may find that usability varies according to how a single user interacts with a site. Usability also often depends on the medium of consumption—textual content viewed on the screen may be more usable in a large size, but when it is consumed on paper, it might be better smaller. If you have tried to read large amounts of small-size content online, you know it can be difficult. People tend to find that it is much easier to read it on paper. Some experts have suggested that people read much slower onscreen and tend to scan more than read content online. In this case, the medium of consumption—screen or print—has affected the usability of the content. In the case of the Web, the medium, which includes networks, browsers, screen sizes, and technologies like HTML, often contributes in a large way to usability problems. Throughout this book, the mantra of "know thy medium" should be repeated over and over.

Rule: Usability depends on the medium of consumption.

What is considered usable often varies between sites. An entertainment site would have different usability constraints than a commercial one. Further, the user's familiarity with a site—as well as how often the user accesses the site and for what purpose—will affect the site's perceived user friendliness. Consider how people may feel about the usability of a site that they have never been to before and are only marginally interested in, as opposed to one that they frequently visit or must use. They may be much more forgiving of errors in the site they need to use or have come to use than in the one they are just casually interested in. In short, a "throwaway" single-time-visit site has different usability constraints than a site a user relies on day to day.

Rule: Usability depends on the type of site as well as the user's familiarity with it.

Tnis idea might seem a tad unusual, but it shouldn't. People often come to believe inefficient ways of doing things are perfectly acceptable. Be careful about getting too scientific when talking about usability (measuring page clicks, mouse travel, errors rates, and the like). How users "feel" about the experience when they come away—their satisfaction with the site or the task performed—is really the most important thing. For some people, how they feel may not always be logical or even totally related to what happened during the site visit. Consider how many people gain satisfaction from performing difficult tasks; they may feel that way about some sites as well. Also, people let organizations that they are familiar with outside the Web get away with things at their sites that a new company can't, simply because they trust the name brand of the older firm. On the other hand, don't assume that the occasionally illogical user can be used as an excuse to produce a site that is hard to use. A site that requires the user to learn a new way of doing things, is inflexible, results in errors, or just doesn't work will generally result in poor user satisfaction. Improve usability and users will be happier.

Rule: Usability and user satisfaction are directly related.

To understand how to make something usable, you must understand users. The next few sections will discuss usability in light of user capabilities and tendencies. The conclusion of the chapter will revisit these subjects and present a few rules of thumb that can be applied during Web site design to improve a site's usability.

Who Are Web Users?

Site designers often make the common mistake of oversimplifying or completely ignoring the capabilities and desires of users. In some cases, concerns about designing the site with a particular browser or bandwidth in mind replace any serious thought of the user. Don't design your site for Netscape—design for people who happen to use the Netscape browser. Always remember the following very important Web design rule:

Rule: Browsers don't use sites, people do.

Fortunately, most designers don't go the extreme of completely forgetting the user, but often they do oversimplify who the site's users are. Far too often, sites are built for some elusive stereotypical Web user—the modem user accessing via AOL, perhaps. This user is just a nameless person surfing the Internet to be enticed into visiting the site and performing whatever task the designer desires. The reality is that users are not automatons with the same capabilities and desires, but individuals with a wide range of physical capabilities, needs, wants, expectations, and goals. Real Web users have bad days or can't always figure things out sometimes, just like the rest of us.

Suggestion: There are no generic people. Always try to envision a real person visiting your site.

While it may not be possible to create a perfect stereotypical user to design Web sites for, there are some general things that can be said about users. The first thing is to think about how today's typical user interacts with a Web site. Until alternative browsing environments such as cell phones or PDAs become much more commonplace, a user of your site is almost certainly sitting at a desk or table with a computer.)Users sit at most a few feet from a monitor and generally use a keyboard and a mouse to interact with a Web site shown on the monitor. Primarily, they are using their eyes to access the information on the screen, though sound may also come into play. The stimulus from the site is filtered, and choice items may be consumed or, more accurately, committed to short- or long-term memory. The information they consume then may cause them to react by, for instance, clicking a link or entering data into a form. This simplified view of a user interacting with a site is shown in Figure 2-1.

ote *Adding in the constraint of mobility, the user's environment can really change usability. If you consider a user who is walking while browsing on a PDA or cell phone, you can see how the environment can affect usability.*

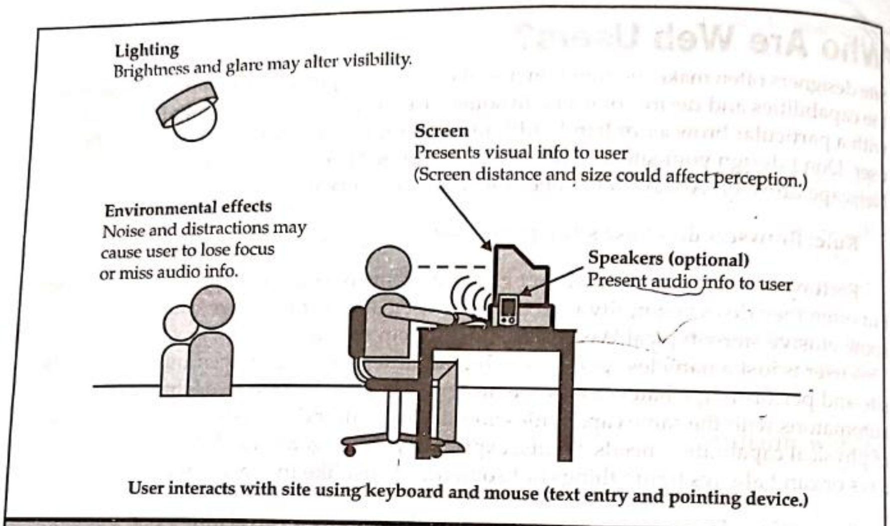
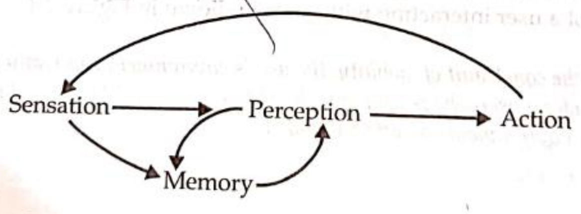

Lighting
Brightness and glare may alter visibility.
Screen
Presents visual info to user
(Screen distance and size could affect perception.)
Environmental effects
Noise and distractions may cause user to lose focus or miss audio info.
Speakers (optional)
Present audio info to user
User interacts with site using keyboard and mouse (text entry and pointing device.)

Figure 2-1. *Typical environment of user interacting with a site*

You can describe how people tend to react to the world around them, including Web sites, in the following way. First, they encounter some sensation that is stored in memory. Then they try to understand the sensation, which is filtered both consciously and unconsciously. Information from past experiences may be called into action, influencing how they perceive things and possibly helping them decide what to do. From this perception, they may perform an action—or possibly take no action—that will later result in more sensations to be interpreted. This simplified action/reaction/action loop is shown here:



Do not think that people can be simplified to a formula where a stimulus is provided that results in an action. People are more complicated than that. People are capable of learning things, and information they encounter is committed to memory that can be used to modify what they do. Further, people aren't perfect. Problems may occur, such as not remembering things properly. Different people perceive stimuli differently. Not everyone sees color quite the same way, for instance. Despite its simplification, the model does force designers to consider how people interact with the world—which includes their Web site. Common user characteristics such as sensation and memory need to be well considered, at least in a general sense, when building sites.

Common User Characteristics

There are no generic people, but people tend to have similar physical characteristics. Most people tend see about the same, are capable of remembering things, and react to stimuli in about the same way. However, remember that people are individuals. There will be some users who will be able to see much better than others. There will be people who can memorize hundreds of links and be able to quickly filter them, and others who will be overwhelmed when presented with more than two choices. There will also be a few users who react much faster or much slower to information than the average user. However, as with all aspects of Web design, we should aim first for the common user and make sure to account for differences. Let's first consider common user characteristics such as vision, memory, and stimulus reaction.

sion

The first aspect to consider about users is how they receive information from a Web site. The primary way most users consume data from Web sites is visually. They look at a screen and consume information in the form of text, color, graphics, or animation. The user's ability to see is obviously very important. Consider, for example, users with poor eyesight. Unless the text is very large and the contrast between foreground and background elements very distinct, they may not be able to effectively interact with the content of the site. Unfortunately, many sites seem to assume that users have nearly superhuman vision, as text is sized very small, or a minor degree of contrast is used between foreground and background elements. A simple example of some of contrast and sizing problems can be found at http://www.webdesignref.com/visionissues.htm.

In order to avoid troublesome color combinations, designers should be aware of how color is perceived by the human eye. Three factors affect how color is perceived:

- **Hue** the degree to which a color is similar to the basic colors—red, green, and blue—or some combination of these colors.
- **Saturation** the degree to which a color differs from achromatic (white, gray, or black).
- **Lightness** the degree to which a color appears lighter or darker than another under the same viewing conditions.

Users with vision that is somewhat color deficient are often unable to differentiate etween colors of similar hue when those colors are of the same lightness and saturation. or example, someone with the most common color deficiency—red-green color blindness—has trouble distinguishing between red and green when the red and green are close in saturation and lightness. Such color vision issues can be troublesome when you consider the difficulty in distinguishing between red and green traffic lights. Does the color-deficient user really know when to stop or go? In the real world, probably so, since the red light is always the top light— but on the Web, things aren't always so cut and dried. If links are similar in hue, lightness, and saturation, it might be difficult for someone to determine which links have been clicked and which have not.

Web page designers can avoid vision issues for users if they follow a few simple rules. First, make sure not to use text or graphic combinations that have a similar hue. Instead of using light blue on dark blue, use blue on yellow or white instead.

Suggestion: Avoid using text, graphics, and backgrounds of similar hue.

It is possible to get in trouble when using colored text on backgrounds with similar saturation. For example, instead of using a grayish blue text on a rose color background, where both colors are close to achromatic gray, use white text on a rose background, or vice versa.

Suggestion: Avoid combining text, graphics, and backgrounds of similar saturation.

The most obvious problem is when contrast is not great enough. Designers need to consider that dark text on a dark background or bright text on a bright background just may not be readable on all monitors or by people with color or vision deficiency. Instead of using a light blue text on a pale yellow background, use blue text on a white background. Or, black text on a white background is always a safe bet. Yellow and black contrast very well, and, therefore, they are used on road signs that are very important to read. However, before changing your Web site to this color combination, consider that design shouldn't be thrown completely out the window just because of usability concerns.

Rule: Keep contrast high. Avoid using text, graphics, and background of similar lightness.

A very important use of color in a Web page is link color. In general, you should really avoid modifying link colors in any way. However, if you do modify link colors, make sure to avoid using link state colors of similar hue, similar saturation, or similar lightness to the background or to one another. For example, avoid links that change from red to pink. For some reason, designers seem to favor such types of combinations. Instead, consider using links that change from dark blue to pink, similar to the normal link state. Be careful with the background color, as it may interfere with link readability. Because of this, white is a good background color. However, if a sacrifice has to be made with color contrast, make the visited state color the one with the contrast problem, since these are links the user would generally be less interested in.

Links, as well as normal text, often have problems with backgrounds. In particular, avoid patterned backgrounds with multiple hues, saturations, or levels of lightness.

Backgrounds like speckles or texture patterns tend to make poor backgrounds; instead, choose a subtle pattern or simple color.

Suggestion: Avoid using busy background tiles.

To make pages more readable and to deal with users who might have some color or vision deficiency, Web designers should make sure colors that are meant to distinguish items are significantly different in two areas (for example, hue and lightness). By following this rule, if the user is color deficient in one area (for example, red-green hue), he or she can still distinguish the item by another attribute, such as its lightness or saturation.

Rule: Make sure colors that are meant to distinguish items like links are significantly different in two ways, such as hue and lightness.

For more in-depth discussions of vision, color, and imagery on the Web, refer to Chapters 12, 13, and 14.

Memory

Memory is critical to a user being able to utilize a site. If users are unable to remember anything about a site as they browse it, they will become hopelessly lost, since they will not be able to recall if they have been someplace before. However, any user's memory is far from perfect, and users don't consciously spend time trying to memorize things. Users tend to always follow a simple rule: try to do minimal work for maximal gain. Simple human nature suggests that a user is not going to spend a great deal of time to figure something out unless there is a potentially good payoff.

Rule: Users try to maximize gain and minimize work.

Of course, what is considered a good payoff will vary from person to person. Some people like to solve complex puzzles just for personal satisfaction. For them, the payoff is an intense feeling of accomplishment from solving a problem. However, let's assume that users are generally not going to exhibit such behavior; rather, they will only work hard if they know they need to or if there is a really good payoff that will result. If you want a blunt or somewhat negative way to remember this idea, just assume users are lazy! More general rules of thumb about how users tend to act will be presented later in the chapter. The previous rule is simply presented to tie in with a few ideas about memory.

Now, assuming users will not like or even avoid Web sites that require them to work too hard, forcing them to memorize things is not a good idea. To illustrate this idea in practice, consider the interface of an automated telephone banking system. When you call the bank, you are prompted for your account number and then read a list of items and corresponding keys to press—"Press 1 for balance, press 2 for transfer, press 3 for payments... ." If you encounter such a system and are unfamiliar with all the choices, using the system can be difficult. You may find that you will try to remember a choice presented in your mind until all the choices have been presented. If too many choices are presented, you might not be able to recall the range of choices or you might even forget which item you chose and have to listen to the choices again. Now, if the same information were presented on a small text menu, it would be much easier to find the item. You would just look over the list and pick the appropriate one. The voice example requires you to recall the choices, which is very difficult. In general, it is easier for users to recognize choices than to recall them. Because users who make mistakes will then tend to favor easier-to-use systems, we should always try to rely on recognition over recall.

Rule: Recognition is easier than recall.

There are plenty of examples of how recognition is easier than recall. Students generally consider a multiple-choice test to be easier than a fill-in test. You must study, of course, for each (assuming the tests are created correctly), but the amount of memorization required is much higher for the fill-in test. The multiple-choice test doesn't require the depth of memory because you will see the answer and recognize it (hopefully) with only a minimal amount of "recall effort."

It turns out that many of the rules and suggestions presented in this book ultimately are related to this idea of recognition being easier than recall. For example, consider the idea of modifying link color. If we turn off link coloring so that links never look visited, we are forcing users to recall whether they have selected a certain link before. If the links do change color, users simply have to recognize the different color to know they have been there before.

Rule: Do not make visited links the same style or color as unvisited ones.

Another important aspect of memory to consider is that it isn't perfect. Users are not going to memorize things easily and often will have only partial memory or a flawed memory of something. Just as in real life, repetition will lead to improved memory. For example, frequent users or power users may actually rely on memorization of the location of objects on the screen, but most users will have only vague memories of how link choices or pages are organized. However, when people are memorizing things, it is known that image memory is one of our most robust forms of memory. It is far easier to retrieve pictures or even words or ideas that evoke pictures than it is to retrieve abstract ideas without visual cues from memory. It is often far easier to remember a person's face than it is to remember the person's name. Given that users will generally find it easier to remember visuals, it would be wise to make pages that should be remembered visually different from the rest. For example, in site navigation, a home page serves as a safe zone for a user. Using a distinct image or a different color is important in making the home page memorable. However, do not assume the user to have perfect memory—don't make the home page only subtly different from the other pages or expect the user to notice or memorize text items.

Suggestion: Make pages that should be remembered visually different from the rest.

Another aspect of memory that is important to the usability of Web pages is the amount of information a person can recall from short-term memory. Let's return to the automated phone banking system example. When users hear the choices, they have to memorize them. If too many choices are presented, they might forget an item. This is an example of short-term working memory. In a sense, we need a little scratch space in our brain to remember something for a few moments. This memory does not hold a great number of items and is highly volatile. Cognitive scientists have long been interested in short-term memory and have conducted many experiments where participants are presented random objects or words and asked to quickly look at them or to make choices from them to test short-term memory. What is found is that participants are able to recall a range of seven items, plus or minus two, from short-term memory. What this means is that when given five to nine items, the user will be able to recall all the items for a short period of time and have them equally present in mind for choosing among them.

The implication of users being able to remember quickly 7 (±2) items on Web design may or may not be profound. If you present a user with a set of links, shouldn't you limit the choices to from five to nine? It would seem you should—if you want the user to choose from the choices "fairly." For example, if you present a list of dozens of what may appear to be randomly ordered links to a user, you will find that the user will have a tough time picking from them. You may notice that users will tend to favor extremes. In practice, the author has seen this happen on Web sites. For example, a large music site faced a problem in that bands listed in the site having names beginning with A or Z had a much higher download rate than anything else. What was happening is that users had little knowledge of the bands, so they would scan the lists and—unless something jumped out at them—they tended to choose the first or last items in the list to see what happened. They really couldn't remember all the names of the bands that were interesting as they went along—there were just too many of them. If you want users to easily choose from a list of things that are equally important, you should limit your set of choices to between five and nine items.

Suggestion: Limit groups of similar choices such as links to between five and nine items.

However, do not go overboard with the five to nine items idea. Some usability experts, in fact, believe this rule has no place on the Web. This seems unwarranted, given the support for the rule both from long-term human capabilities studies and from GUI practices, which tend not to put 100+ choices on a single screen. However, there is some merit to the idea of not putting too much stock in the 7 (±2) rule. Consider that some designers might be tempted to use this rule to suggest that pages should have only five to nine links on them. However, this could be rather limiting if you have a lot of content. Users can focus on items progressively. Consider, for example, being presented five to nine distinctly different clusters of links on a page. Maybe the clusters are labeled and colored so the user chooses a cluster after looking at each. Once in the cluster, there are five to nine links. In this sense, there might be as many as 81 links on a screen, and the user will still be able to use them easily. When looking at well-designed pages with numerous links, you hopefully will see fewer than 100 links and notice that the clustering used an organization method, such as alphabetical, to avoid memorization.

Memory rules of thumb can also be applied to clicks. It appears that users are able to remember about three pages presented sequentially. Anything more than that and there tend to be gaps in memory. For example, as users click through dozens of pages, they will probably remember a variety of pages but not all sequentially. The memorable pages may be visually different enough to trigger recall. Usually, such distinguishable pages are termed *landmarks*—the most obvious landmark page in a site being, of course, the home page. However, if you want users to remember a path, they tend to remember only about three page views sequentially—and maybe fewer if the pages look nearly identical. Therefore, you should not expect a user to memorize a sequence or path longer than three items without repeated use. The number of markers showing location and path in today's sites and the user's continual reliance on the Back button and browser's history mechanism demonstrate how tenuous sequential memory tends to be. Because of these memory constraints, we tend to see many sites trying to reach content within three clicks or complete transactions in as few screens as possible.

Suggestion: Aim for memorization of only three items or pages sequentially.

This is by no means a complete discussion of memory, but it does serve to remind Web designers that, in order to make a site easy to use, we need to limit the amount of memorization going on. The less effort the users expend trying to recall what sequence of buttons they pressed or what choices they may have seen, the better.

Response and Reaction Times

If you have watched people browse around Web sites, it is obvious that some people are faster than others. Some users appear to cut quickly through page content and make choices rapidly, and are frustrated with even the slightest download delay. Others struggle to keep up and seem to have the patience of Job when it comes to waiting for pages to load. However, over time you'll come to find that people's patience for Web page loading will go away, particularly as they become more frequent users. Consider, for example, how long it takes for users to become annoyed at an automated teller machine that has not returned their money to them. The entire transaction may only take a few seconds, but customers are quickly annoyed. But when automated tellers first came out, a wait of even 30 seconds to a minute seemed tolerable compared to waiting in a long bank line.

Tip: Users tend to be more patient with something they are unfamiliar with or that is a novelty.

We see this novelty/patience dynamic on the Web all the time. Sites that could be considered single-visit sites, like movie promotion sites or designer portfolios, get away with huge download times. These sites could be termed single-visit or "throwaway" sites, since the user is unlikely to return. Splash pages, excessive animations, and long downloads are less annoying to a user who hasn't seen them before, but patience wears thin on return visitation. Consider that even when a splash page has a "skip intro" button, a return visitor will still be frustrated with having to even make such a choice. The very fast loading design of successful, heavily frequented sites, such as portals or e-commerce sites, shows that patience wears thin. The needs and desires of the first-time visitor, who in some sense could be considered a novice user of the site, are different than the frequent or expert user of a site. However, users do not have infinite patience, and they are getting more and more impatient as they get used to what facilities the Web, or a particular site, provides. In general, we find the following rule to hold true:

Rule: The amount of time a user will wait is proportional to the payoff.

The better the payoff, the longer the user will wait. Users who get something for free or who are stealing some desirable piece of software or music seem to be willing to wait an eternity. Consider users who illegally download software, songs, and movies from the Internet with a modem. They'll literally spend hours searching for and downloading songs when they could have gone out, worked at a near-minimum-wage job, and earned enough to purchase an entire CD in a similar period of time. Of course, this imbalance will certainly change with the increase in bandwidth—much to the annoyance of the music industry. But it remains true that if you are going to expect a user to wait for a page to load, there better be something useful there.

The amount of time users will wait will vary based on the individual user, his or her personality, and the potential benefit of waiting. However, there are some things we can say about response and reaction times for users in general. Some usability experts (for example, Jakob Nielsen, www.useit.com) relate that studies about response times report similar results. Common response times and user reactions are summarized in Table 2-1.

Time Elapsed	Probable User Reaction
0.1 second	When something operates this fast or faster, it appears instantaneous or nearly instantaneous to the user. Unfortunately, due to bandwidth and technology constraints, few Web pages will exhibit this level of responsiveness in the near future.

Table 2-1. *Response Time and User Reactions*

Time Elapsed	Probable User Reaction
1.0 second	When something reacts in around a second, there is no major potential for interrupt. The user is relatively engaged and not easily distracted from what is happening on the screen.
10 seconds	This is suggested to be the limit for keeping the user's attention focused on the page. Some feedback showing that progress is being made is required, though browser feedback such as a progress bar may be adequate. However, do not be surprised if the user becomes bored and decides to move on to something else.
> 10 seconds	With a delay this long, the user may actually go about other business, look at sites in other windows, talk on the phone, and so on. If you want users to continue to pay attention, you will have to give them constant feedback about progress made and some sense of when the page will be finished (as when downloading software, the browser lets users know how much time is left to complete the download).

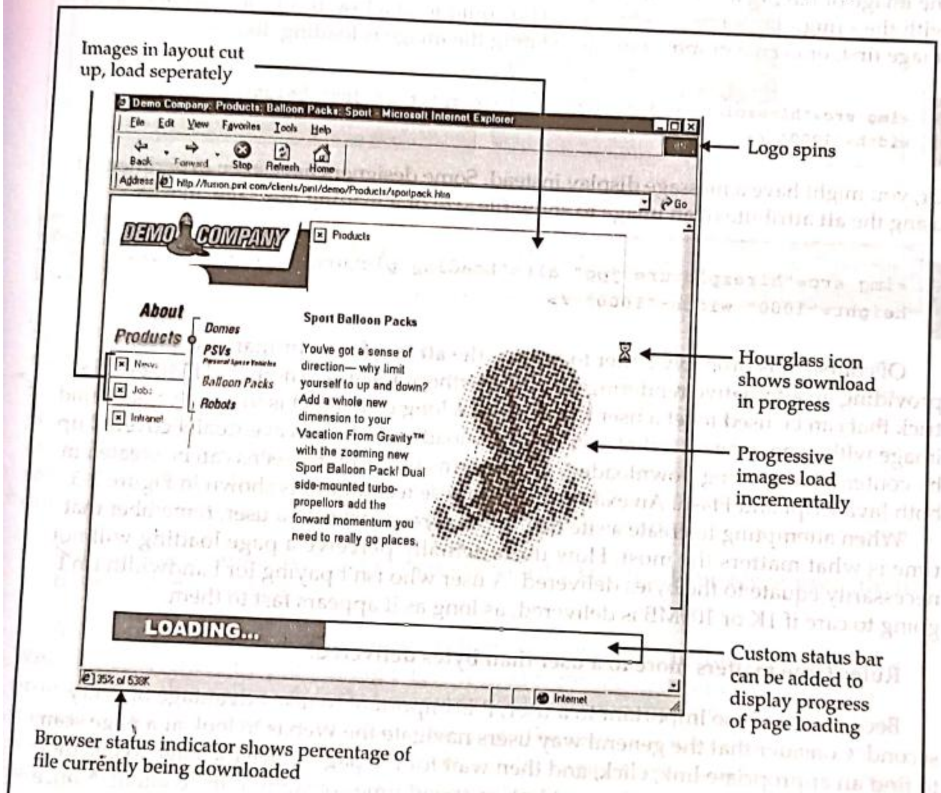
Table 2-1. *Response Time and User Reactions* (continued)

When it comes to the Web, there is generally little chance of going too fast for the user. Most of the time, it takes more than a few seconds even on a broadband connection to download something. However, be careful once something like a Flash file is downloaded. If the user has a faster processor than you, the program may end up running much faster on their system than expected, so much so that the user might not be able to keep up. On occasion, you may notice how animations used in some Web pages appear to travel at a rate only a superhuman could read.

Tip: Be careful with overly fast response times of downloaded objects.

In most cases, a Web site will probably not outpace the user; in fact, it may be much too slow for the user's liking. Because users may get impatient, you need to make sure that they are given some indication of the progress being made. The browser itself actually gives a great deal of feedback about the progress being made. When loading a page, a browser will generally convert a cursor to a wait indicator (such as an hour glass), spin or pulsate a logo (generally in the upper-right corner of the browser), provide a progress meter towards the bottom of the screen, and display messages about objects being loaded in the status bar at the bottom of the screen. The Web designer will design pages to provide even more feedback. For example, the designer may build pages so text loads first or pieces of the page are loaded one at a time. Often, designers will cut images up into multiple pieces, so the user will see a little bit loaded at a time. Also, designers often use images that load in a progressive fashion from an unclear one to a sharp one so that the user is able to get a general sense of a complete fuzzy picture early on and watch its loading progress, if necessary. Figure 2-2 illustrates all these progress indicators in action.

For page loads that only take 10--20 seconds, the feedback given by the browser and incremental loading of a page should be enough to let the user know something is going on. However, when loading takes longer, you should give the user more information. For example, many sites that use binary technologies like Flash use a special loading page complete with a status bar showing progress. Such progress meters can also be created using technologies like JavaScript. However, don't bother with a progress bar


Demo Company: Products: Balloon Packs: Sport - Microsoft Internet Explorer
File Edit View Favorites Tools Help
Back Forward Stop Refresh Home
Address http://fusion.pint.com/clients/pint/demo/Products/sportpack.htm
Go
DEMO COMPANY
Products
About
Products
News
Jobs
Intranet
Domes
PSVs
Personal Space Vehicles
Balloon Packs
Robots
LOADING...
35% of 538K
Internet

Sport Balloon Packs

You've got a sense of direction— why limit yourself to up and down? Add a whole new dimension to your Vacation From Gravity™ with the zooming new Sport Balloon Pack! Dual side-mounted turbo-propellors add the forward momentum you need to really go places.

Images in layout cut up, load seperately

Logo spins

Hourglass icon shows sownload in progress

Progressive images load incrementally

Custom status bar can be added to display progress of page loading

Browser status indicator shows percentage of file currently being downloaded

Figure 2-2. *Browser and site feedback shows progress of download*

or other forms of feedback unless load times are around 30 seconds or more. (With the proliferation of broadband Internet access, this time will certainly diminish; even now, many broadband users are likely to get impatient around 20 seconds or less, and in due time even 10 seconds may seem like a long wait.)

Rule: When response times such as page loads take more than 30 seconds, try to provide your own feedback to the user, such as a load-time progress bar.

If you are building a static site, there are some simple tricks to let the user know about a longer wait for an object. For a very large image download, besides interlacing the image or having it show up progressively sharper, it is also possible to use a trick with the **<img>** tag's **lowsrc** attribute. You could load a low-resolution version of an image first, or even a graphic message stating the image is loading, like so:

```
<img src="hirezpicture.jpg" lowsrc="lowrezpicture.jpg" height="1000"
width="1000" />
```

Or, you might have a message display instead. Some designers have even experimented using the **alt** attribute of an image to show file size or a loading message, like so:

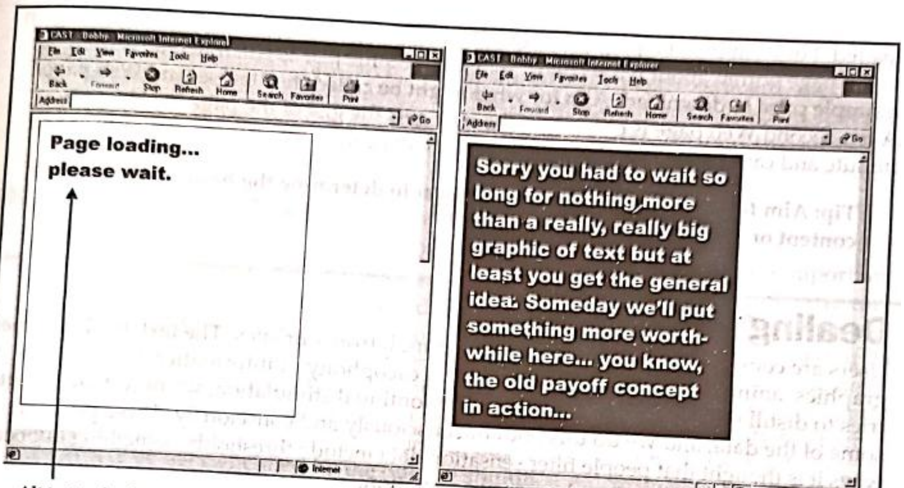
```
<img src="hirezpicture.jpg" alt="Loading picture of Mars (800K)"
height="1000" width="1000" />
```

Of course, it is probably better to reserve the **alt** text for its primary purpose—providing an alternative rendering for users without images. Another HTML or CSS trick that can be used to let a user know about a long download is to use a background image with a message on it that says a page is loading, which is eventually covered up by content that is being downloaded. Other forms of loading screens can be created in both JavaScript and Flash. An example using these techniques is shown in Figure 2-3.

When attempting to create a site that appears responsive to a user, remember that time is what matters the most. How users actually perceive a page loading will not necessarily equate to the bytes delivered. A user who isn't paying for bandwidth isn't going to care if 1K or 100MB is delivered, as long as it appears fast to them.

Rule: Time matters more to a user than bytes delivered.

Because time is so important to a user, it is important to take advantage of every second. Consider that the general way users navigate the Web is to look at a page scan to find an appropriate link, click, and then wait for the page. Once the page loads, they then look at the page to find the next link or spend time consuming the content. Notice the time is split between user "think time" and download time. The reality is that for most users, the think time for navigation pages is pretty small compared to the wait


CAST : Bobby - Microsoft Internet Explorer
File Edit View Favorites Tools Help
Back Forward Stop Refresh Home Search Favorites Print
Address
Go
Internet

Page loading... please wait.

Sorry you had to wait so long for nothing more than a really, really big graphic of text but at least you get the general idea. Someday we'll put something more worth-while here... you know, the old payoff concept in action...

Alt text or background image showing status

Figure 2-3. *Let the user know a long download is in progress*

time. For content pages, however, the user may spend a great deal of time looking at the page. One way to improve responsiveness is to take advantage of the thinking time by downloading information to be used later on. This is often called *preloading* or *precaching*. Assuming you are able to preload most or all of the next thing to be looked at by the user during the think time, the next page load time could be significantly reduced. Somewhat like the magician who has the result of a trick set up in advances, downloading during idle moments can produce a nearly mystical appearance of speed.

Suggestion: Improve Web page response time by taking advantage of user "think time" with preloading.

A variety of browser accelerator tools have been built in an attempt to improve Web responsiveness by preloading pages linked from the current page. The only problem with this approach is that many pages have so many outbound links that it is difficult for the browser to predict the page the user will load next. The best way to improve the odds of caching the correct "next page" is to look at the common paths users take through a site by examining a log file and then putting in code to preload pages along these paths. However, this just improves the odds. The only time you can really guarantee that preloading will improve things is when the user is navigating a linear progression of pages.

The responsiveness of a Web site is a key aspect of a user's feeling of the site's usability. Beyond loading of pages, consider that time is important to a user even after a page has

Rule: Make clickable regions large enough for users to move to them quickly and press them accurately.

General user capabilities are not all that we need to consider when discussing what ideas affect usable Web design. We must also consider the world the user inhabits and the user's general and unique characteristics and experiences.

The User's World

People truly are the centers of their own universe, in the sense that they perceive everything initially from their own point of view. Consider the idea of how a user might perceive the Web site shown in Figure 2-4. The user lives in the real world.

Users are affected by their environment: the physical conditions of their location, the noise around them, the visual quality of the monitor they are using, and so on. From their world, they access your Web site via the medium of the Internet and the Web, which includes things like network connections, servers, browsers, and so on. Once on the Web, they navigate about and visit sites. If they decide to actually interact with a site, they finally begin to consume or react to the content presented.

The presentation and navigation layers in Figure 2-4 could be interchanged considering that a user's ability to navigate Web space is greatly affected by the way it is presented.

Suggestion: Always remember that you need to bring a site into the user's world, not the other way around.

The preceding suggestion is an important one. Designers will naturally believe that they have set the rules for their sites and that users are just visitors. While this may be true, users tend to interpret things from their own perspective. Each user will have his or her own opinions, capabilities, environment, and experiences, all of which will influence how the site is interpreted. A fine balance between what the user thinks and

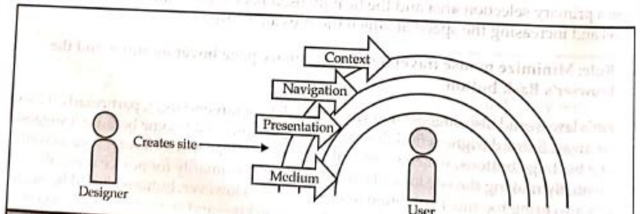


Figure 2-4. *The user's universe*

wants and what the designer thinks and wants has to be struck. This will be discussed in more depth later in this chapter.

User Environments

The user is heavily influenced by what could be called their *environment of consumption*. For example, consider a user in a public place such as an airport using a public Internet kiosk to remotely access their e-mail. The user is standing up—it might be crowded and noisy—waiting to dash off to the plane. Because of this environment, the user may not be tolerant of long waits, excessive menus, or anything that slows down the task at hand. Further, due to the noise, the user may not be able to always hear sound cues. Last, because the user is standing up, the amount of time they might spent during the whole online session will certainly be significantly less than a normal session at the office. When designing for users, always think about where the user is accessing the site from. Table 2-2 details some of the possibilities.

The environment will greatly affect the user's view of what is "usable." For example, color combinations that contrast acceptably indoors might be troublesome outdoors. Designers must take into account the environment of consumption.

Location	Characteristics
Office	Generally computer-based access Single user Relatively quie Should be primarily work or task focused, at least during primary work hours Often high speed
Home office or bedroom	Generally computer-based access Single user Noise level variable, but often quiet Purpose may be work or play Access could be anytime Speed of access varies dramatically from modem to high speed
Home living room	Access may be from set-top box or video game console Distance from device may be farther Use may be less input oriented (reduced typing) Noise level variable May be group-oriented access or single user Access probably more entertainment related Printing may not be an end result

Table 2-2. *Common User Environments Characteristics*

Location	Characteristics
Cybercafe	Probably computer-based access Cost may influence usage Noise level variable Use is probably entertainment or research Speed of access probably high May be group-oriented access or single user Security or privacy may be a concern
Public kiosk	Cost may influence usage Noise level variable User may be standing Use will be less input oriented (reduced typing) Use is probably task oriented, focused on e-mail or access to very important information Access to location-related information may be a high priority Security or privacy may be a concern
Car	Probably noncomputer-based access (PDA or smart phone) Use will be less input oriented (reduced typing) Focus will not be primarily on the access if user is the driver Use is probably task oriented or limited to very important information Access to location-related information may be a high priority Speed and quality of access is probably low
Mass transit or plane	Probably either noncomputer-based access (PDA or smart phone) or a laptop User may be standing or sitting Use could be entertainment or work Access to location or time-sensitive information may be a high priority Speed and quality of access is probably low Security or privacy may be a concern
Outside	Probably noncomputer-based access (PDA or smart phone) Screen glare could be a significant problem Use will be less input oriented (reduced typing) User may be standing or moving Noise level variable Use is probably task oriented or limited to very important information Access to location-related information may be a high priority Speed and quality of access is probably low

Table 2-2. *Common User Environments Characteristics* (continued)

Rule: Account for the characteristics of the probable environment in which the user will access a site if possible.

General Types of Users

There are three levels into which users can be classified to reflect their knowledge of how to use a Web site: novices, intermediates, and experts or power users.

A novice user is one who may have little knowledge of a site or even of how the Web works. A novice user will need extra assistance and may prefer extra clicks with extra feedback to accomplish a simple task. An example of an interface tuned to novices would be a wizard that automates some common task.

At the other end, power users are those users who understand the Web or a site very well. Power users should be considered in two distinct categories: frequent and infrequent visitors to the site. A power user who frequently visits a site to utilize advanced features such as sophisticated searching, may directly form their own URLs, and memorize the position of objects within a page or the site. A power user who is an infrequent visitor to a site may not be familiar with the site's structure but will expect certain facilities, such as search, to be available to navigate a site. Power users will need relatively little handholding and will desire to click less and consume more. Obviously, the distance between a power user and novice user is great. A site geared too much toward one audience or the other will certainly annoy—the power user if the site has been dumbed down, or the novice user if the site is geared mostly toward power users.

The third group of users, the infrequent intermediate user, is actually the largest category of users on the Web. Most users are infrequent intermediate users because they pretty much understand how the Web works, but may not know how to navigate a particular site in a very efficient manner. Infrequent intermediate users do not continually revisit the site; if they do, they will probably eventually become a power user. Because site usage tends to be dominated by intermediate users, you may consider designing the site for the capabilities of these users. However, doing so may lock out novice users and bore or restrict advanced users.

The best approach to building a site for basic user groups is to build a site that provides features that cater to all users. Software applications do this, so there is no reason a Web site cannot. A software application may provides keyboard shortcuts and other features, such as customizable interfaces, for power users while also providing icons and menu systems for intermediate and novice users. Help systems and wizards are other features mostly geared toward the novice user. A Web site could provide features like a clean URL system, advanced search facility, and personalization features for an advanced user. A site with consistent navigation bars that have button labels similar to other sites (About, Products, Careers, and so on) is very friendly to novice and intermediate users, and it can also have dynamically built "bread crumb"–style navigation lines, popular with advanced users. Last, a Web site could provide help systems, maps, and alternative forms of access such as simple text links for the novice.

Suggestion: Aim to create an adaptive Web site that meets the requirements of novices, intermediates, and advanced users.

In the perfect world, there is no reason that a Web site can't be built to meet the needs of all general user groups. However, time and cost constraints may limit the number of features that can be added to some Web sites. In such cases, it is probably best to aim for the largest group of users: the intermediate. This may lock out some novice users unable to figure the site out. There is an argument to be made for aiming at the lowest common denominator in a user. The problem with this is that if you start building only for the complete novice, you can quickly alienate users who know what they are doing.

Suggestion: Design for the intermediate user if an adaptive Web interface is not possible.

Even if an adaptive interface is built, there is bound to be a user who doesn't understand or like the site we have built.

Tip: Remember there will always be users who don't like or get a site, no matter how good it is.

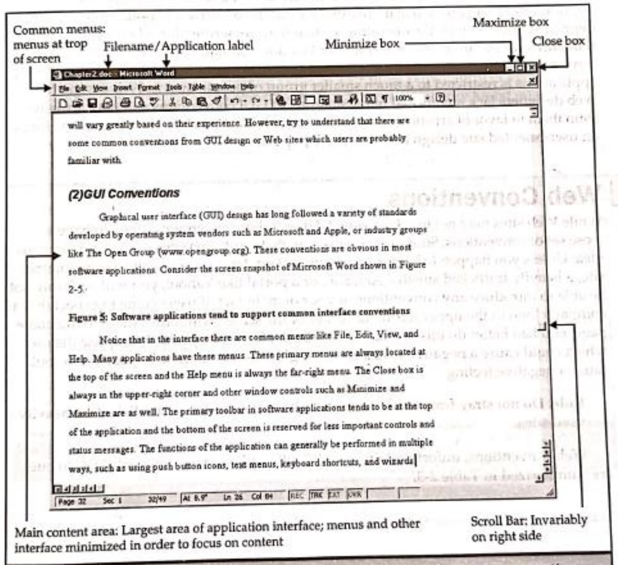
Users are individuals with different tastes and opinions. They will have different experiences, capabilities, personalities, age, gender issues, and cultural issues. Some individuals may have disabilities that prohibit them from using a Web site that most users find easy to use. Users bring what they know from the real world and from other Web sites to your site. They may expect to use symbols from the real world, such as those for navigation. However, they may also bring knowledge of how Web sites work that they gained from visits to many other sites. Knowledge of how traditional software applications work may also be brought into play. Remember, as mentioned early in the chapter, that users bring the site into their world—they don't visit the universe of your Web site. Your site is just a speck in an overall universe of Web sites. In fact, it could be said that most of the time users are not at your site. Some call this the 99 percent rule, since 99 percent of the time, users are probably not using your site. You should, therefore, make sure that your site follows any conventions and meets expectations set up by other sites.

Rule: Users bring past experiences with the world, software, and the Web to your site. Make sure your site meets their expectations.

You need to make sure that your site acts like other sites or software users have used and meets their general expectations. Remember the rule of consistency: if you do things differently from how everybody else does, you can't rely on a user's past knowledge and you force the user to learn something new. Of course, the challenge with real users is that expectations will vary greatly based on their experience. However, try to understand that there are some common conventions from GUI design or Web sites that users are probably familiar with.

GUI Conventions

Graphical user interface (GUI) design has long followed a variety of standards developed by operating system vendors such as Microsoft and Apple, or industry groups like The Open Group (http://www.opengroup.org). These conventions are obvious in most software applications. Consider the screen snapshot of Microsoft Word shown in Figure 2-5.


Common menus: menus at trop of screen
Filename/Application label
Minimize box
Maximize box
Close box
Chapter2.doc - Microsoft Word
File Edit View Insert Format Tools Table Window Help
100%
Page 32 Sec 1 32/49 At 8.9" Ln 26 Col 84 REC TRK EXT OVR
Main content area: Largest area of application interface; menus and other interface minimized in order to focus on content
Scroll Bar: Invariably on right side

will vary greatly based on their experience. However, try to understand that there are some common conventions from GUI design or Web sites which users are probably familiar with.

(2)GUI Conventions

Graphical user interface (GUI) design has long followed a variety of standards developed by operating system vendors such as Microsoft and Apple, or industry groups like The Open Group (www.opengroup.org). These conventions are obvious in most software applications. Consider the screen snapshot of Microsoft Word shown in Figure 2-5.

Figure 5: Software applications tend to support common interface conventions

Notice that in the interface there are common menus like File, Edit, View, and Help. Many applications have these menus. These primary menus are always located at the top of the screen and the Help menu is always the far-right menu. The Close box is always in the upper-right corner and other window controls such as Minimize and Maximize are as well. The primary toolbar in software applications tends to be at the top of the application and the bottom of the screen is reserved for less important controls and status messages. The functions of the application can generally be performed in multiple ways, such as using push button icons, text menus, keyboard shortcuts, and wizards

Figure 2-5. *Software applications tend to support common interface conventions*

Notice that in the interface in Figrue 2-5 there are common menus like File, Edit, View, and Help. Many applications have these menus. These primary menus are always located at the top of the screen, and the Help menu is always the far-right menu. The Close box is always in the upper-right corner, and other window controls such as Minimize and Maximize are there as well. The primary toolbar in software applications tends to be at the top of the application, and the bottom of the screen is reserved for less important controls and status messages. The functions of the application can generally be performed in multiple ways, such as using push button icons, text menus, keyboard shortcuts, and wizards.

GUI conventions are very useful to know, particularly when designing forms and other interactive elements of a site. In later chapters on implementation, we'll discuss the use of GUI widgets and the difference between Web and GUI interfaces. The Web has not been able to develop conventions that are as well understood as those for software applications. There are two main reasons for this. First, software applications are often defined significantly by the operating system they are written for. Microsoft has great influence on how applications written for Windows should work. Apple can dictate conventions for Macintosh software. Second, the ability to author and distribute software applications is restricted to a much smaller group of people than in Web design. Many Web designers lack any formal understanding of GUI conventions and may actually shun them in favor of artistic freedom. This struggle is fortunately changing, as the focus on user-oriented site design becomes more popular.

Web Conventions

While Web sites may not exactly follow GUI usability conventions, they do have a loose set of conventions. Straying from the way that most Web sites work is a dangerous idea. Unless you happen to be running an important day-to-day use site like an internal site, a heavily trafficked site like Amazon, or a portal like Yahoo!, you will probably not be able to introduce any conventions of your own. In fact, if users come to expect that a company logo in the upper left-hand corner of the screen will return them to the home page, you had better do this in your site. If you don't do this, you may surprise the user, which could cause a negative reaction. Forcing the user to learn a new idea also could cause a negative feeling.

Rule: Do not stray from the common interface conventions established by heavily used sites.

Web conventions, unfortunately, are difficult to pinpoint. A few well-known ones are summarized in Table 2-3.

Convention	Description
Upper-left corner logo signals home page return.	Users tend to expect a corporate logo to return them to the home page. Most sites put this in the upper-left corner. An explicit Home button, as well as a ToolTip, is a good idea.
Text links are repeated at the bottom of a page.	Most sites like to repeat text navigation at the bottom of a page, particularly if top or side navigation is a graphical form.
Back-to-top link used on long pages.	While sites will provide text navigation to move to the next page, a back-to-top link or arrow is generally included at the bottom of the page to quickly jump the user up the page.
Special print forms used for heavily printed pages.	Increasingly, sites are providing special printer-friendly versions either in a stripped HTML form or even in an Acrobat form. This is most commonly found on sites that distribute large volumes of content.
Shopping cart in the upper right.	Typically, the shopping cart icon or link is found in the upper-right corner of the screen.
Clickable items are blue and underlined.	Fight it all you want, but most text links are blue and underlined. While many users may be able to understand nonunderlined links or different colors, the best way to signify that something is pressable is making it blue and underlined. Be careful with creating logos or other content that are blue, as users may actually try to click on it.
Secondary navigation elements such as a site map or search are presented separately from sectional navigation.	Because site maps, site indexes, help systems, and search facilities are navigation aids, most sites have tended to put less emphasis on links to them. However, given the rise in popularity of search features, content-rich sites may emphasize search facilities.

Table 2-3. *Some Common Web Conventions*

Figure 2-6 illustrates some of the common Web conventions used in a page within the DemoCompany site.

The problem with Web conventions is that they are moving targets. New conventions may be invented and sweep across the Web like fads. For example, frames and splash pages used to be popular, but they have somewhat fallen out of favor. Conventions are not always well considered and may often have more to do with novelty than usability. However, this shouldn't lead you to invent new conventions or avoid those that are current. The best way to keep up with current conventions is to simply browse the well-trafficked e-commerce and content sites often and look for common features. If users are exposed to features there, such as single-click ordering, it isn't going to be difficult to explain to them how it works on your site. Don't assume that everyone understands common conventions or that all users will be able to use current conventions. Some users will have special needs.

Accessibility

There is no way to account for all the small differences between people. In fact, we only aim to create sites that *most* people like. This may lead us to stereotype groups of users (like casual female surfers under 18, and so on), but this may be an approach we have to make. Yet, this does not mean that you should go out and build a site catering to the largest demographic group of users hitting your site. Try to please as many distinct groups as possible by making your site as accessible as possible. Don't forget that some people may have difficulty if you assume that all users have perfect physical and technical capabilities.

Providing accessibility for people who may have deficiencies involving sight, hearing, or other physical capabilities isn't just a nice idea anymore—it may actually be required for some organizations, particularly government agencies—and many companies could incur serious liability if they do not account for all users. For example, Section 508 of the 1986 Federal Rehabilitation Act requires that the federal government include solutions for employees with disabilities when awarding contract proposals. This would also eventually apply to systems such as intranets, extranets, and most likely public Web sites. Also, the 1992 Americans with Disabilities Act (ADA) states that firms with 15 or more employees provide reasonable accommodation for employees with disabilities. This could apply to intranets or extranet creation!

But making a Web site accessible is something that should be done, not because of some law or to avoid future litigation, but because doing so could result in a much better Web site for everyone. Very often, creating systems that are accessible to all users also creates benefits for all users, regardless of capability. For example, the so-called talking books, initially considered for the blind, fostered books on tape. Also consider that easy ramps to access buildings, and curb cutouts made for wheelchairs, make walking easier for all and tend to reduce the number of people falling flat on their face after crossing the street or severely twisting their ankles as they step off the curb.

The W3C (http://www.w3.org) has long advocated designing sites for maximum accessibility and promotes the Web Accessibility Initiative (http://www.w3.org/wai). The WAI is concerned not only with creating sites that are accessible to people with disabilities but also with making sites that are accessible by anyone who might be operating in a different environment than what a designer considers "normal." Remember that users will not necessarily be using a fast connection and a large monitor like yours—or if you aren't using a fast connection with the latest and greatest, your users just might be! From the W3C guides, you should always consider that users may have different operating constraints:

- They may not be able to see, hear, or move easily, or may not be able to process some types of information easily (or even at all).
- They may have difficulty reading or comprehending text because of language problems.
- They may not be able to use a keyboard or mouse because of access method (such as a cell phone) or physical disability.
- They may have a less-than-ideal access environment, such as a text-only screen, a small screen, a screen without color, or a slow Internet connection.
- They may be accessing the site in a nonstandard environment where they may be affected by environmental factors—accessing the Web in a noisy cybercafe or as they drive a car, for instance.
- They may have an older browser or a nonstandard browser or operating system or use an alternative form of user interface, such as voice access.

To deal with these issues, the W3C has issued a few suggestions to improve the accessibility of a site. These are summarized here:

- **Provide equivalent alternatives to auditory and visual content** In other words, don't rely solely on one form of communication. If you use picture buttons, provide text links. If audio is used, provide a text transcript of the message, and so on.
- **Don't rely on color alone** As discussed earlier in the chapter, not everyone will be able to view colors properly; so if color alone is used to convey information, such as what constitutes a link, people who cannot differentiate between certain colors and users with devices that have noncolor or nonvisual displays will not be able to figure out what is being presented. In general, you need to consider avoiding color combinations with similar hues or not enough contrast—particularly if they are likely to be viewed on monochrome displays or by people with different types of color vision deficits.
- **Use markup and style sheets, and do so properly** Basically, make sure to use HTML for structure and CSS for presentation. Especially avoid using proprietary markup or presentation elements and avoid using technology that may not render the same way in different browsers.

- **Clarify natural language usage** Make sure to define terms and use markup that indicates acroymns, definitions, quotations, and so on. In other words, use more logical markup. Further, make sure to clearly indicate the language being used in the document so that a browser may be able to switch to another language.
- **Create tables that transform gracefully** In short, don't use tables for layout—use them for presenting tabular data such as a spreadsheet. When tables are used, provide a clear caption, column headings, and other indicators of the meaning of cell contents.
- **Ensure that pages featuring new technologies transform gracefully** This is a key idea discussed throughout the book. Basically, make sure that, if you are going to push the limit of design, any new technologies degrade gracefully under older browsers. For example, if you are relying on JavaScript, does the page still work without it on? Or at least evor gracefully?
- **Ensure user control of time-sensitive content changes** Make sure that moving, blinking, scrolling, or autoupdating objects or pages may be paused or stopped by the user. Besides being highly annoying, such distractions may actually make it difficult for users to focus on the site.
- **Ensure direct accessibility of embedded user interfaces** If you use an interface within the page—for example, a Java applet that has its own internal interface—make sure that it, too, is accessible.
- **Design for device independence** Try to build interfaces that can work in multiple devices, including those with different screen sizes, different viewing devices (cell phones as well as computers), and different manipulation devices (keyboard or mouse and keyboard). A particularly important consideration is just making sure that a site doesn't rely solely on the mouse for navigation. Some users may find mouse movement difficult, and power users may actually prefer to use the keyboard for navigation.
- **Use interim solutions** Because not all browsers will support the same technologies or standards completely, make sure to provide alternatives in the short term for noncompliant browsers.
- **Use W3C technologies and guidelines** A somewhat self-evident but occasionally troublesome suggestion. Of course you should always try to follow the W3C guidelines, at least in spirit. However, be careful because many W3C guidelines are no more than proposed ideas, and browsers may lack significant or consistent support for a defined specification.
- **Provide context and orientation information** In some sense, this just means to explain things or provide instructions for complex areas. You should design pages so that the meaning of links is clear through the use of ToolTips or scope notes. Further, forms should be designed that explain what is required. In the most basic way, a site should provide a help system.

will analyze a Web page and see if it meets certain basic accessibility criteria, such as the use of alt text.

Building a Usable Site

One of the keys to usable Web site development is to focus from the beginning on the users of the application. Remember that the user's goal is not to use computers or to use your Web site. The user's goal is to accomplish some task—purchase a product, find a bill payment center, register a complaint, and so on. You should try to make direc contact with users, and you must listen to them. Do not fall into the trap of thinking that you should just simply ask users what they want and then they will design your site for you. Users are not designers, and they make illogical or unrealistic request Because of this, you may be tempted to implement your own idea of a great site instead without regard for user requests. However, the core idea of user-centered design is to always remember we are designing for users and not ourselves. Recall again the followin very important Web design rules:

Rule: You are NOT the user.

Rule: Users are NOT designers.

Although not all user input will be valuable, you should solicit information from your intended audience. You might consider interviewing them or giving a survey. Whatever you do, make sure to let users talk—and listen to them. While this may seer like JAD (Joint Application Design), which will be discussed in Chapter 4, we will not let users control the project; rather, they will be used as a source of ideas and a way tc verify the execution of implemented features. From interviews, you should build a profile of stereotypical types of users. While this may seem to be a bad idea, conside that unless you have a very small audience, it is virtually impossible to build a site th will conform perfectly to all the preferences and task requirements that all possible user might have. Even if it were possible, it would be prohibitively expensive.

From your discussions with users, build a prototype site, or just a set of simpl diagrams on paper of how pages might look, and test it out with users. Make sure yo test your site with users as early as possible in the development cycle so as not to buil a site that users can't figure out.

Suggestion: Perform user testing early and often.

There are many ways to verify usability. Tests might include

- Casual observation of users
- Surveys and interviews
- Focus groups

- Lab testing
- Heuristic evaluations by developers or usability experts

The results of the tests can include more quantitative measurements, such as the number of mistakes made during a task, the amount of mouse travel, the time it takes to perform a task, and so on. Tests will certainly also have to include qualitative measures of what feature the users liked or didn't like. Before you don a white coat and rent lab time in a room with a two-way mirror to observe users, consider that formal testing may be overkill for most sites because of the cost and trouble of performing user tests in a formal fashion. Simple observations might do the trick, and opinions tend to be free from many users, though not always well founded. Collect a few users, or even your friends and neighbors, and sit them down at the site, and have them perform a few tasks. What's interesting is that even an informal test will uncover the major problems with a site. However, informal tests only work if you let them. Designers seem far too proud of their sites and tend to act as co-pilots, showing a user the interesting aspects of a site. Talking too much during a test or guiding the user in any way keeps the user from making his or her own decisions and may actually steer the user away from mistakes.

Suggestion: When performing even an informal usability test, avoid talking too much or guiding the user.

Before running off to round up your friends to ask them what they think, first consider that far too often users will tell you what you want to hear or what they *think* they would do in certain situations. Or they simply may not want to admit their misunderstandings. It is better to observe users' behavior than to rely on statements from them. However, if this is not possible, user input is acceptable, particularly if it is coupled with your own ad hoc usability analysis of a site. For instance, see whether the site follows the basic usability criteria that have been described in this chapter. Table 2-4 presents some guidelines you should use for judging a site.

When evaluating a site, the rules of thumb here cover the basic aspects of usability. However, don't assume just because the site meets most of these basic ideas that it is a good site. There are plenty of other ways for a site to fall down. For example, a site might not contain excellent content, its technology may be unreliable, or its graphics may be hideous to look at. Chapter 5 presents a more in-depth evaluation procedure that accounts for many other aspects of Web design. Remember that usability isn't the only part of a positive Web experience.

Guideline	Explanation
Be consistent	Consistency is the key to an easy-to-use interface. If something is consistent, the user only has to learn it once. Within your own site, don't change the position of buttons or the way things act.
Don't violate a user's expectations, and make sure to follow Web and GUI conventions	Consistency can go beyond the contents of a site. Users will have expectations about how things work shaped by visits to other sites. Make sure your site is consistent with what they expect. In short, follow any conventions used in GUI or site design that the user is familiar with.
Support the ways people use Web pages	Users use the Web pages in a few basic ways. They load a page, they unload a page, they print the page, they save the page (either by bookmarking the address or saving the file to a local drive), they read the page, or they interact with the page (such as by filling in fcrms or manipulating content objects within the page). As with the previous guideline, make sure users can do all the things they expect to be able to do. If users expect to print or bookmark a page and they can't, they may consider the site unusable.
Use surprise properly and sparingly	Occasionally, being inconsistent is useful. If you want to "wake a user up," it might be OK to dramatically change the way a page looks or acts. Just make sure you don't do this often, since users may never become comfortable with the site and may even become frustrated with the ever-changing interface.
Simplify the site and individual pages as much as possible	Simplicity makes it easy for users to understand a site. Try to pare a site or page down to its bare essentials. Look at statistical logs to determine what pages are not needed from a site. On a page level, remove clutter from layouts and try to reduce visual noise.

Table 2-4. *Common Web Usability Guidelines*

Guideline	Explanation
Rely on recognition, not recall	Memorization is difficult. Don't expect the user to memorize the structure of your site or the position of your buttons. Minimize what the user has to remember by exposing available choices. Even something as simple as hiding a menu when it isn't in use increases the cognitive load on a user, since they have to memorize what items are on what menus.
Do not assume users will read instructions	You may not get a chance to hold a training class for every user who visits your site. Generally, users will read help files only when they are in trouble. Make sure that they don't need to be trained. Avoid introducing features in a site that would require training or documentation for proper use.
Prevent or correct user errors	Don't let users make mistakes that are unnecessary. For example, validate form entries and limit users to doing only what they should. Don't provide a choice that is not easily undone by the user. If errors do occur, let the user know about the error and its possible solution.
Provide feedback	Let users know what's happening. Don't be imprecise with feedback. If there is going to be a delay, let them how long it is going to take. If an error has occurred, provide a clear error message.
Support different interaction styles	Try to provide multiple ways of doing the same thing to deal with different approaches to problems. For example, some users may prefer to use a site map rather than a search engine when looking for something. Don't limit users, but do account for a range of interaction styles, from novice users to power users.
Minimize mouse travel and keystrokes	Typing and moving the mouse around the screen is work for the user, so try to minimize it. This means successive button choices should be nearby. Try to minimize the distance from primary navigation to the Back button, which is certainly the most commonly pressed button in a browser. Navigation should probably be toward the top of the screen.

Table 2-4. *Common Web Usability Guidelines* (continued)

Guideline	Explanation
Consider medium of consumption.	Make sure to understand where the user will consume the content—on screen or on paper. If users print pages to consume them, shouldn't the usability test be performed on the paper document as well?
Consider environment of use.	If known, consider where a user will interact with a page. Where users interact with a page will affect how usable they perceive it. For example, relying on sound in a noisy environment isn't a wise idea.
Focus on speed.	Users dread slow-loading sites. Make sure pages are fast-loading by practicing minimal design. This doesn't mean eliminating graphics, only that a page should be no slower than it needs to be to deliver its message.

Table 2-4. *Common Web Usability Guidelines* (continued)

Usability Above All Else

One problem with usability discussions is that it is easy to use usability concerns as a way to squash any other reasonable value. For example, some people have gone so far as to discuss how banner ads contribute to poor site usability because they are animated or increase the download time. However, consider that without the banner ads the site may not be economically viable. Pleasing graphics also are a common target for usability experts. It is interesting to note how boring most usability gurus' sites actually are. While a site without much graphics may be usable, it might not do much to improve the brand identity of the organization running the site; in fact, without graphics, it may undermine brand identity built through other mediums. In some situations, it may be important to let the user endure a slightly longer download in order to see the corporate logo and new advertising look.

Advanced technology also is a common enemy of good site usability. The truth is that while advanced technology may lock out some users, what is provided may be worth it. If we always designed for the lowest common denominator, we'd still have text-only Web pages. Don't let usability completely stifle innovation. Usability is certainly very important, but there are often other considerations in a Web site's design. Always remember that while we design for users, we are ultimately in control of our site.

Suggestion: Do not use usability concerns as a way to avoid or eliminate visual, technological, or economic aspects of a site.

Who's in Control of the Experience?

While it is true that we must give the people what they want, the masters of sites—meaning those who pay for them—may have desires that are not congruent with the desires of the site's users. Do not become a slave to the user; remember that, in some sense, we are the masters of our own sites. How we want to treat our visitors is going to influence greatly how they feel about visiting our sites. Do you want to be a dictator, forcing the user to download certain plug-ins or resize a window? Conversely, you could be very democratic and let users pick their own path through your site. You may even allow users to modify content on the site or influence other users with indicators of link popularity. Last, you could aim for a middle ground and maybe act as a benevolent dictator, trying to help users along the way and giving them freedom within certain constraints, but always trying to guide them along.

The issue of control during a site visit is somewhat of an unwritten contract between the site user and the developer. There is give and take in the relationship. While one of the main tenets of user-centered design is to put the user in control, users are imperfect like everyone else; if we give them complete control, they may make serious errors. Developers will want to keep users from making mistakes. However, the role of the benevolent dictator of the online experience is difficult. If you control things too much and users notice that they can't resize their window or press certain buttons, they may become angry or frustrated. The key is to provide an illusion of control.

Users should be able to do everything they need to do and nothing more. People need to feel like they are in control, but the control should have limits. Good interfaces exhibit this control. Consider, for example, the famous adventure game *Myst*™. In *Myst*, the user can click objects onscreen and move in a direction simply by clicking in the appropriate direction. The interface is very simple and also very restrictive, though game players rarely notice this. In *Myst*, as in many well-designed video games, the progression is very controlled by the game designer, but the illusion of control is always preserved. A great Web site would follow the cue of a video game by trying to guide someone to a conclusion like purchasing a product, but in a manner that the user doesn't really notice.

The best example of the balance of control in an experience is probably Las Vegas. Casinos create a complete experience of visiting an ancient land, tropical paradise, or foreign country. A gimmick outside the casino like an exploding volcano or pirate battle attracts hordes of visitors. The intent is that some of these visitors will step onto the nearby conveyor belt to be quickly whisked into the casino. Inside the casino, temperature, lighting, and oxygen level are carefully controlled in an attempt to create a pleasant environment. The passage of time becomes difficult to determine because windows are few and tinted, and clocks are nonexistent. Assistance is plentiful from dealers and waitresses who will provide free drinks. If you get hungry, cheap food is nearby at an all-you-can-eat buffet. Want to stay overnight? Rooms are reasonably priced—and if you spend enough, they might even be free. But when you come to your senses as your wallet begins to empty, notice how difficult it is to find the exit! Good

Las Vegas casinos practice the ultimate in experience design, second only (maybe) to Disneyland. The experience is always controlled; the point is to maximize the money the casino takes in. If you step out of line, get irate and loud when you lose, or try to do something to win back control in gambling by card counting, you'll find that you are quickly escorted outside. The experience is fun and you can win, but the control is there and the house always has the edge. It's pure math. If you plan to run a commercial site, learn from Las Vegas.

Suggestion: Practice "Las Vegas" Web design. Provide the user with a pleasant experience, complete with perks and the illusion of unlimited choices, but control the situation strictly at all times.

Summary

Usability is about the aspects of a site that aren't always noticeable but yet seriously influence the ease in which a user is able to accomplish a task using the site. Usable sites should be easy to learn, easy to use, and easy to remember. They should also result in few errors and be satisfying to the user. While some ways to improve usability, such as consistency and simplicity of design, are easy to formulate, sometimes it is difficult to satisfy the needs of every user. One reason is that users have different Web skill levels—novice, intermediate, and advanced (power users)—that will affect site usability. Another is that, while users generally share certain capabilities for accessing a site, such as vision and memory, users are also individuals, with unique characteristics, opinions, and experiences. They will also tend to view your site as a mere island in a big ocean of sites, and it is best to assume that they won't want to learn your special rules.

With so many varieties of users, you probably won't be able to perfectly accommodate every user's unique tastes and requirements. However, if you create an adaptive interface that can be used by the three broad categories of users and make sure to test your site carefully with real users, you stand a good chance of making a site that is usable by most users. Be particularly careful not to lock users out, particularly those who may be disabled or slightly different from your average user.

Finally, a site should always be built to meet the needs of its users within the constraints or the desires of its creators. However, never use the quest for a usable site as a way to avoid difficult problems or as an excuse not to use graphics or technology or introduce new features that a user might want. An overzealous Web professional waving the usability banner can easily stifle innovation. Balance is always the key to great Web design.