Chapter 6

HUMAN FACTORS AND E-COMMERCE

Dennis F. Galletta

Abstract: Many Web sites have been built hastily and without regard for the user. Extensive research on pre-Web user interfaces could have prevented such poor designs. It appears that both Web developers and Web users are less experienced than developers and users of other software applications. One of the most important guidelines provided by the literature is that of testing, which should be mandatory for any site to enable revision and retesting. There are several specific areas of understanding in usability that are also described, along with implications for Web designers. These areas include Fitts' Law, site organization, difficulties of user search, ill effects of page loading delay, trade-offs between site depth and breadth, options for what is seen by the user in inputs and outputs, and problems with users' mental models. Tools such as storyboards, checklists, and evaluation models are also described to prevent design errors. Once designers become aware of these guidelines, usability problems, and tools, Web site design quality might begin to catch up to that of systems sold in the mass market.

Keywords: e-commerce, Fitts' Law, human–computer interface, human-computer interaction, HCI, human factors, usability, Web site design

Preview Version: Published in *E-Commerce and the Digital Ecomony,* 2006, Volume 4 in the AMIS (*Advances in Management Information Systems*) Series: ME Sharpe Publishers, pp. 91-111

Note: pagination might not be accurate compared to the actual book chapter

THE PROBLEM

The scenario is perhaps all too familiar but, nevertheless, surprising each time: In an attempt to gather some comprehensive information, compare prices, and purchase a product, a user becomes frustrated with various sites and gives up in frustration. One online store was very sluggish, using an elaborate video introduction and unnecessarily large photos on each page. The second site was filled with garish graphics, contained little useful information, and triggered a pop-up window that said "Call Sales at (555) 555–1234." The third used unfamiliar terminology to categorize the information on the site and lacked a search function.

It is somewhat surprising and somewhat discouraging that most people reading this scenario will probably have experienced at least two of those three real situations. Although many might have trouble understanding why sites like these are created and why they still exist, some might chalk it up to unrealistic expectations of a shopper's ability, patience, and motivation to keep trying. However, there are more fundamental reasons that can explain the widespread nature of these problems.

Why does this occur? One likely reason is that Web site designers have been using a technique Negroponte (1988) calls "designer introspection," where the designer builds the system according to his or her own taste. To combat this tendency, McCracken and Wolfe (2004) recommend that a designer tape a large sign to the wall above the workstation that says "You Are Not Your User" (p. 37). This paper describes Web usability problems and offers methods of avoiding them based on decades of research in human–computer interaction (HCI).

Web Usability Problems

Brinck et al. (2002) describe common usability problems on the Web:

- Human perception problems, caused by allowing the physical storage of information to dictate the design or by allowing artistic style to dominate usability.
- Navigation disorientation, caused by inadequate logical architecture, insufficient indicators of the current location, or a mismatch between the site's language and organization and the user's needs.
- Human memory limitations, when too much is required of the user, over a long time frame, or when something must be memorized among other items that are very similar. A useful example: needing to remember an account number on a second screen after the first screen has disappeared.
- Database integration, where information on the screen becomes out of synch with information in the database (such as when prices change or items go out of stock), or when status information is not shown (so users might accidentally order multiple items).

Why do these problems exist? Perhaps Web designers and users are quite inexperienced.

Designers Are Inexperienced

Decades of research and experimentation on user interface design have been adopted by two design communities to varying degrees of success: those designing general use, mass market packages and, perhaps to a lesser degree, those designing high-visibility customized applications in organizations.

About a decade ago, some of the largest organizations provided software with puzzling messages, pre-installation steps, and mysterious questions. Documents were stored in folders tucked away with the applications and users often had trouble finding where they "went." Some applications were very creative, redefining the user's experience to the extent that they were placed on a bookshelf for future experimentation that never occurred. Although the user's path was obviously unpleasant then, going back even another ten years paints a sadistic picture of software designers.

In contrast, today's software mass market is close to maturity in designing applications. We have finally reached the point where the installation of a new high-market share software package (or even an operating system upgrade) provokes few puzzling questions, and, in most cases, can be accomplished in a nearly unattended short session. Today's situation might even be called uninterestingly reliable. Programs and documents are assigned by default to standardized locations, there are consistent menu and icon schemes, and most dialog boxes are standardized (for example, in most modern applications, opening and saving files will invoke the same dialog boxes). The standardization is at such an extreme that a user will not have much trouble moving from applications on one operating system to another. That is quite an accomplishment!

Custom applications used in organizations by smaller groups of users have improved as well. With a smaller set of potential users, lack of incentives to win market share, and dramatically smaller budgets, there are fewer opportunities to achieve the usability demonstrated by the mass market. While custom applications will probably always lag behind the mass market, they have also come a long way over the years.

What has changed over the years to improve usability so much? Obviously, today's mass market and corporate application designers have benefited from decades of experience in constructing complex applications. This experience has brought with it:

1. graphic user interfaces assembled with visual programming languages and oriented around objects and events,
2. interface standards recommended (or even imposed) for the operating system by the vendors,
3. high-level tools that allow creation of standard system components (such as pull-down lists, dialog boxes, and file management) with little programming,
4. published research describing usability standards and practices,
5. tools that allow users to critique quick prototypes of systems,
6. more involvement of users in defining requirements, and
7. planning for incremental improvements and other iterative processes.

It is somewhat curious that many Web designers do not seem to avail themselves of these helpful resources. One likely explanation is that Web designers come from a community that is less experienced with software design than those who design mass market and custom software in organizations.

At least in smaller firms, it might unfortunately be quite descriptive of practice to use the common joke that a firm does not have to pay thousands of dollars for building a Web site if the CEO has a computer-savvy teenager at home. For larger firms, it is common to relegate the design completely to outside consultants who self-proclaim what is often self-taught skill. In some firms, design is done in-house by computer-literate members of the marketing department. While such individuals, even the teenager, are likely to be quite creative, few of them are likely to have usability skills or even be aware of the products of experience listed above.

Unfortunately, the difficulties are not only on the developer side. There are also some difficulties caused by the inexperience of a large number of users.

Users Are Inexperienced

Decades of reductions in costs, new applications, and the Web browser have led to a larger computer market than ever before. Long gone are the days in which only computer professionals or selected personnel used computers. The most recent statistics reveal that 81.4 percent of all workers have access to a personal computer at work when omitting agriculture and manufacturing from the industries examined, and that 71 percent of all computer use at work is Internet related (Hipple and Kosanovich, 2003). According to the most recent study by Nielsen/Netratings (2003), about 168.1 million people in the United States (about 60 percent of the population) have Internet access. It is quite obvious that people from a variety of backgrounds are online.

With a small group of fervent experts, the quality of the interface is important. With a large, heterogeneous population of potential users, the quality of the interface is crucial. For example, an expert user might avoid using double or single quotes in text boxes, knowing that it can cause some systems to report an error message. A server's error message that warns of illegal content in the order form could cause a novice user to become very confused or frightened. They might wonder who broke the law. Another novice might try to choose multiple radio buttons at the same time and become frustrated.

The only effective remedy to reduce problems with inexperienced users is training. Such a remedy cannot cover all possible Web sites, and might even be unrealistic for a moderate number of sites, because people do not generally wish to spend as much time as would be needed to shift from "interacting with the technology . . . (to) . . . interacting with information, tasks, and other people *via* the technology" (Furnas, 2002, p. 53). The desire for technology that is transparent places a large burden on the developer. The burden is to become aware of methods for improving the transparency, or usability, of systems, so that they are no longer a task, but facilitators of our tasks.

WHAT WE KNOW ABOUT IMPROVING THE USABILITY OF SYSTEMS

Unfortunately, three statements might make Web designers who are reading this throw up their hands in frustration:

- There is no ideal design for a system.
- There are no universal guidelines for every aspect of an interface.
- Sites must differ in meaningful ways or the online world will bore the user.

Fortunately, many design principles and approaches that have been adopted in the software industry can provide some help.

Design Principles

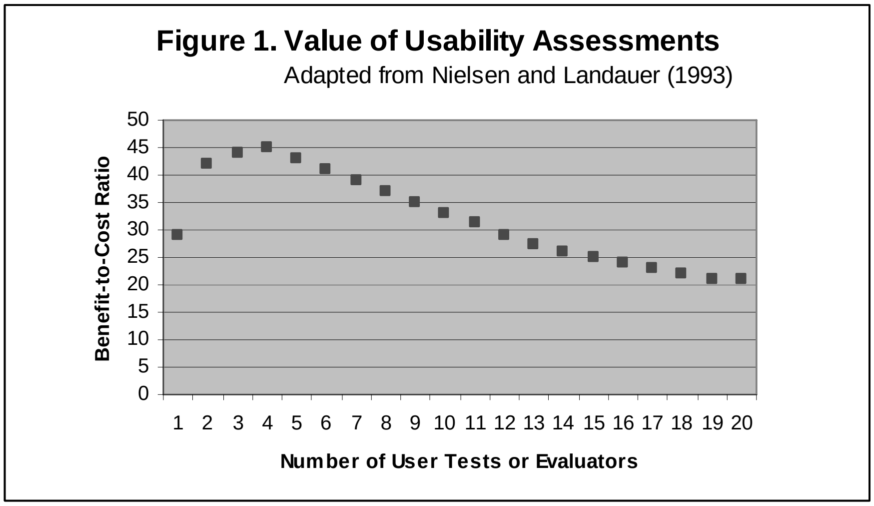
Quite some time ago, Gould and Lewis (1985) studied and wrote about three simple principles of design. A subsequent study provided a more detailed demonstration of the principles in practice (Gould et al., 1987).

The principles aim at building both usefulness and ease of use, and are as follows:

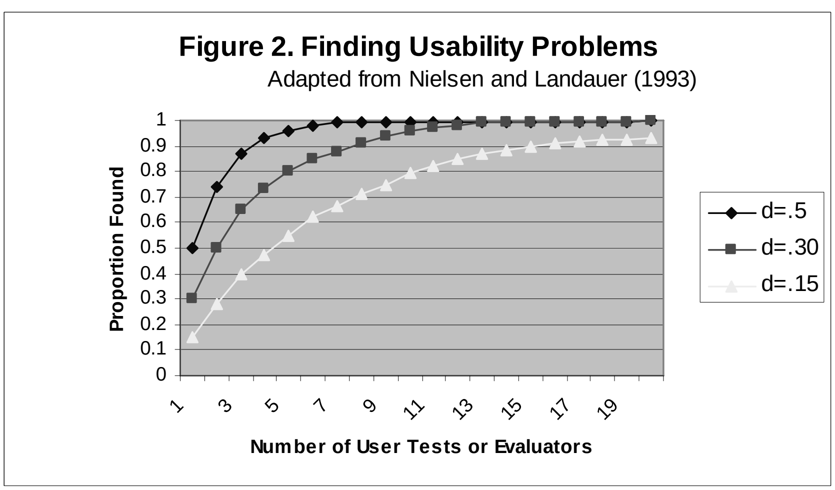
- Early focus on users and tasks (to understand them very well).
- Empirical measurement (observe actual users doing actual tasks with a candidate design or simulation).
- Iterative design (redesign as necessary and try again).

According to Gould and Lewis, using these principles is all that is necessary to achieve ease of use and usefulness. The assumption is that flaws in an early design will become obvious and will eventually be eliminated. Gould et al. (1987) demonstrated the process in a very successful design of an early voicemail system for the 1984 Olympics that had to operate in multiple languages and be used by novices who could not undergo training. The story reveals a surprising number of changes to the system as they went along.

Testing and iteration certainly helps. Based on an analysis of thirteen software development cases, Nielsen and Landauer (1993) provided some estimates of the benefit-to-cost ratio for additional iterations of usability evaluation. In Figure 6.1, the benefit-to-cost ratio beyond one iteration jumps dramatically until it reaches five iterations. Landauer (1995) provides the observation that the "benefit-to-cost ratios are rather large . . . (and he knows of) no other software engineering techniques with anything approaching these payoffs" (p. 304). Landauer estimates that such testing is worth 500 times its cost for large systems and five times its cost for small systems.



Subsequent testing is clearly considered to be valuable, but the value decreases slightly with each test. Clearly this is dependent on the particular set of users and the system under scrutiny. Nielsen and Landauer provide the corollary of Figure 6.1 in Figure 6.2, based on the same thirteen cases. The figure shows the proportion of interface problems found in each subsequent test. In this view, the curves are subdivided into those systems with high, medium, and low detectability of problems.



Naturally, the better the initial system, the fewer iterations that would be needed, so researchers did not stop with these three principles; they continued to study and invent other useful design guidelines.

The Smith and Mosier Guidelines

At the other extreme in terms of the amount of detail offered is the classic 478-page set of guidelines from Smith and Mosier (1986). Based on a comprehensive review of the literature of that time, and not anchored to any particular technology, most of the guidelines are still quite relevant today and deserve careful attention by designers. The 948 guidelines are arranged into the following categories:

- Data Entry (for example, “Provide prompting for the required formats and acceptable values for data entries,” p. 23)
- Data Display (for example, “The wording of displayed data and labels should incorporate familiar terms and the task-oriented jargon of the users, and avoid the unfamiliar jargon of designers and programmers. When in doubt, pretest the meaning of words . . . to ensure there is no ambiguity,” p. 101)
- Sequence Control (for example, “If a command entry may have disruptive consequences, require the user to review and confirm a displayed interpretation of the command before it is executed,” p. 262)
- User Guidance (for example, “When the computer detects an entry error, display an error message to the user stating what is wrong and what can be done about it, (such as) ‘enter two letters then three digits’ (rather than) ‘invalid input’,” p. 316)
- Data Transmission (for example, “Allow users to defer the transmission of prepared messages, to be released by later action,” p. 358)
- Data Protection (for example, “When a password must be entered by a user, ensure that password entry can be private; password entries should *not* be displayed,” p. 385)

The guidelines do overlap to some extent, perhaps for the sake of completeness; the example chosen for the “sequence control” category above also appears in “data protection.” Also, each category provides a very useful narrative introduction that includes design objectives. Nearly all include the following objectives:

- Consistency (of entry, display, actions, procedures, or transmission);
- Efficiency, minimizing user actions;
- Minimal memory load on the user;
- Compatibility with other tasks;
- Flexibility for greater user control.

A designer would do well to at very least keep those five objectives in mind when developing systems.

Shneiderman’s “Eight Golden Rules”

Shneiderman (1998) proposed “Eight Golden Rules” for designing interfaces, which he says were “derived from experience,” and “should be validated and refined” (p. 74):

1. Strive for consistency (in procedures and terminology).
2. Enable frequent users to use shortcuts.
3. Offer informative feedback (for every user action).
4. Design dialogs to yield closure (allowing people to know their progress and when they are finished).
5. Offer error prevention and simple error handling.
6. Permit easy reversal of actions.
7. Support internal locus of control (allowing users to feel they control the computer, not the other way around).
8. Reduce short-term memory load.

The eight rules offer valuable yet general guidance to the designer. However, they are intended to serve all systems in general. Guidelines that are specific to the Web are discussed below.

McCracken and Wolfe's Heuristics

Taking into account the special properties of the Web, McCracken and Wolfe (2004) specify several guidelines for designing usable Web sites. Each of the following is represented by a chapter in their book:

- Know the user. Learn who the users are, analyze their tasks, determine their surroundings, and determine how to measure quality.
- Organize content in a meaningful way. This will be discussed later in this chapter in more detail, but in general, if the content has a natural structure familiar to the users, then use that structure. If not, perhaps another structure (alphabetical, geographical, or chronological) is more useful.
- Organize the screen in a meaningful way. Proximity, alignment, and consistency suggest a relationship among items; contrast suggests that items are different.
- Build a prototype of key screens, test the prototype on real users, and redesign based on user feedback.
- Use colors based on their physical and perceptual aspects to make sure that text contrasts with its background and is in harmony with it.
- Become familiar with basic typography, to avoid fonts that are difficult to read. Use a basic font, use 10- or 12-point type, avoid extended use of bold, italics, or uppercase text, use 2 points of leading between lines, use cascading style sheets to facilitate changes, use left alignment, use dark text against a light background unless you are willing to enlarge the font size, and never use underlining for emphasis.
- Use multimedia sparingly, and only when it is the only way to convey important information.
- Design the site with due consideration to users with sight or motion impairments. Such impairments are discussed in the next section.
- Learn about global differences in language, text (such as date formatting), colors, icons, and images to extend the site's reach and avoid conveying the wrong ideas.
- Learn to build trust by suppression of personal information, judicious personalization, limiting of personal information collected, certification, and references.

There are more sets of principles and guidelines, but those offered above are likely to provide a good head start. It is perhaps becoming evident that there is no shortage of system design guidance at nearly any level of abstraction. Some of that guidance now addresses Web sites specifically. For example, a new book by Koyani, Bailey, and Nall (2003) has identified 187 guidelines focused on Web design. The book presents and rates each on relative importance and the strength of the underlying research evidence.

It would probably be useful for a Web designer to become familiar with several sets of guidelines for creating the first iteration of a site. Then, using Gould and Lewis's (1985) iterative approach, a final design will involve less iteration.

One additional set of guidelines is quite important yet underutilized. These are guidelines to improve the accessibility of sites to people with impairments, a topic we now turn to.

Web Site Accessibility Guidelines

Millions of people have impairments. The World Health Organization (WHO, 1997) estimates that 180 million people worldwide have visual impairments and 121 million people have hearing impairments. Paciello (2000) reviews several strategies to assist those with mobility impairments.

IBM (2005) recommends producing accessible Web sites because:

- Everyone should be able to access a Web site.
- An e-commerce firm will enjoy a larger customer base.
- Worldwide regulations are emerging.
- Benefits can help everyone. Making a site accessible actually seems to make it more understandable and usable for everyone, even those without physical challenges.

The Web Accessibility Initiative (WAI) has created fourteen Web Content Accessibility Guidelines (W3C, 1999), mainly to ensure that documents "transform gracefully" to the medium that is needed (for example, to voice for vision-impaired individuals) and to make content understandable and navigable:

1. Provide equivalent alternatives to auditory and visual content.
2. Don't rely on color alone.
3. Use markup and style sheets, and do so properly.
4. Clarify natural-language usage.
5. Create tables that transform gracefully.
6. Ensure that pages featuring new technologies transform gracefully.
7. Ensure user control of time-sensitive content changes.
8. Ensure direct accessibility of embedded user interfaces.
9. Design for device independence.
10. Use interim solutions.
11. Use W3C (World Wide Web Consortium) technologies and guidelines.
12. Provide content and orientation information.
13. Provide clear navigation mechanisms.
14. Ensure that documents are clear and simple.

Finally, a very popular software tool called "Bobby" (Watchfire.com) allows for automatic analysis of the conformance of any given Web page to detailed accessibility guidelines.

Synthesis: How to Design

It seems very clear that design for usability requires testing and iteration. The entire development process might benefit from being restructured and made to revolve around usability. Brinck et al. (2002) recommend a "Pervasive Usability Process," where evaluation takes place at every stage. The stages of design include (1) requirements analysis, (2) conceptual design, (3) mock-ups and prototypes, (4) production, and (5) launch. At each stage, the results of evaluation can divert the team back to an earlier stage for additional work. For example, if a user reviews a mock-up that emerged from misunderstandings in the requirements analysis phase, the project can revisit that phase to make appropriate adjustments.

In any event, it is important to move beyond our current meager state of the art in Web development. Lest the reader believe that designers are being criticized with little mercy, it is important to look more deeply into the environment faced by a firm attempting to advance as either an offensive or defensive move.

E-commerce touches many specialized areas in a firm. For example, in a firm such as Wal-Mart, recent moves to plan for future implementation of RFID (radio frequency identification) for inventory control can have implications on at least four independent functions: marketing, strategy, accounting, and supply chain management.

Providing adequate representation from all functions is not a trivial process. Communication barriers can be found at nearly every turn, and conflicts over resources do not help. It is important that an authority provide more strategic, objective assessment of relative importance over needs to make sure reasonable understanding of a broad variety of users is developed. For obvious reasons, identifying the users with greater accuracy would be an important early step in kicking off a testing program.

Maxwell (2002) describes a maturity model for human–computer interaction (HCI), where level 1 occurs with achievement of basic usability; level 2 is achieved when devices and software support a person's role, collaboration, and organizational needs; and level 3 provides individualized and holistic interaction. Level 3 describes an idealized future environment, where people take advantage of embedded and pervasive devices, electronic agents, and virtual presence. Let us agree that with many sites we are still chasing the pot under the level 1 rainbow in e-commerce. However, there are several cases where organizational needs have been served well in spite of poor usability, and we are prototyping on a grander scale, approaching usability after discarding earlier, immature products.

Beyond general guidelines, we know a great deal about individual interactions with software and devices; several well-known studies have been pivotal in shaping our understanding of HCI. It would be very useful for a Web designer to be aware of these theories, rules, and findings.

HCI THEORIES, RULES, AND FINDINGS

There are several perspectives in the literature that will help build usability during the design, development, or testing phases in a Web project. We will describe each one briefly to provide the interested designer or researcher with a head start.

The areas will be subdivided into those that allow understanding of human reactions to design and those that help in the design process.

Understanding Human Reactions to Design

Humans often have what seem to be unpredictable reactions to systems and their design features. After producing a "masterpiece," a designer might be extremely disappointed to find that users are not happy with the usability of the site.

A large number of studies of user acceptance have established the need for usability. In the Technology Acceptance Model (TAM), Davis et al. (1989) show that ease of use and usefulness are the two critical antecedents of behavioral intentions. Dozens of studies have provided results that are consistent with that model. Interestingly, the most recent incarnation of TAM, called UTAUT (Unified Theory of Acceptance and Use of Technology [Venkatesh et al., 2003]) modified the causal links by predicting and demonstrating that effort and performance expectancies, along with social influence and facilitating conditions, have important impacts on behavioral intentions and usage. In any event, with either model, usability is clearly an important antecedent to sustained and committed system usage.

Fitts' Law

Decades ago, Fitts (1955) discovered, in a noncomputer context, that difficulty of pointing to a target increased as the distance from the target increased and as the size of the target decreased. That is, a large target that is close is much easier to reach than a small target that is far away. The difficulty is actually based on information theory, and can be modeled in the following equation. The index of difficulty is measured in bits, but can be converted to time, as illustrated by MacKenzie (1992).

$$IndexofDifficulty = \log_2 \left(\frac{2 * Dis\tan ce}{Width} \right)$$

Jones (2003) studied Fitts' Law as one variable in an Intranet context, by providing different icon sizes (as well as colors and shapes) for an Intranet (purchasing) site. As predicted, icon size was a strong predictor of icon pointing performance, and in turn, performance was a significant predictor of attitudes about the Web site. Icon size also interacted with color and shape in predicting performance. That is, if using different colors and/or shapes of icons, the contribution of icon size is diminished.

Implications of Fitts' Law for Web design include:

- Icons should be larger, rather than smaller.
- If screen space is at a premium, the most frequently used icons should be larger than the least frequently used icons.
- Provide differentiation of icons through colors and shapes and provide icons that are easier to reach through larger size.

Navigation Behavior

Web sites are similar to menu-based systems. People hunting for a particular page will scan keywords in the hyperlinks and choose the one that sounds most promising along the "trail" to finding the desired page. Because this hunting behavior is similar to that of an animal searching for some food, it is often referred to as "information scent" (Paciello, 2000). Some trails provide additional clues and stronger scent, and some trails do not, requiring the user to turn around and backtrack to find another more promising turn.

The ability to find an item depends on the categorization scheme and the user's familiarity with that scheme. The goal of the search is also important. A firm might employ different product lines, each with a particular category that is unfamiliar to the user or unrelated to the goal of the search. For example, Sony, the most popular brand of camcorders, subdivides camcorders into several categories, including Digital 8, DV, and Mini-DV. If a user is searching for a TRV-350 thanks to the recommendation of a good friend, but has no idea which of the formats is represented by that model, he or she will need to search every category until the TRV-350 appears. If the goal is merely to find the most inexpensive item, this categorization scheme will again be completely useless.

From research on menu design, we know that familiar categorization schemes can be powerful forces in leading a user to the goal extremely quickly (Norman and Chin, 1988; Robertson et al., 1981). If users are unlikely to be familiar with the scheme, but know the target of what they need, it might be more effective to arrange the content in alphabetical order. For example, in searching an encyclopedia site for an opossum, many people will not know to look for mammals, then marsupials. It might be more efficient for most people to either search for the term, or, lacking a search facility, use an alphabetic set of links.

A promising option is to provide multiple paths. In the opossum case, providing both schemes will allow the zoologist to quickly move to the correct page by using the natural categorization scheme, and will also allow the grade school student to use the alphabet and browse the resultant groupings.

A creative designer might arrange items into unusual but highly effective categories. For example, for children, there might be a set of categories such as “wet creatures,” “dry creatures,” and “flying creatures.” Further subdivision in each category can include “smooth creatures,” “furry creatures,” and “feathered creatures.” A child might be so delighted with such a scheme that he or she might browse eagerly through dozens of listings.

To sum up:

- The site designer should not assume that the user understands the firm’s official categorization scheme, and should use more familiar terms if possible.
- The designer should provide search facilities if at all possible.
- The designer should provide multiple paths to important content.
- The designer may find that creative paths can be particularly effective.

Searching

However useful, search engines are not a panacea. Research has shown that novice users are confused with search engines (Marchionini, 1995). Queries are often difficult to formulate, and results often include too many hits (false positives) or too few hits (false negatives). Brinck et al. (2002) and Dumais (1988) each paint a gloomy picture:

> “Most people are relatively poor at searching. They use terms that are too broad or too narrow. They overconstrain the search. They don’t consider synonyms. They don’t know how to filter out documents that are irrelevant” (Brinck et al., 2002, p. 169). “It is clear that even the best formal query languages pose problems for users” (Dumais, 1988, p. 675).

Users’ mental models are often incorrect when they search. They often have little experience with Boolean expressions and make errors when they use AND and NOT operators (Jansen et al., 2000). They provide more information to the search engine, thinking that will result in more hits, but the result is usually the opposite of what they expect and the search becomes surprisingly more constrained by the extra terms.

In a study by Jansen et al (2000), a surprising number of actual search engine queries failed. Fully 50 percent made errors when using the AND expression and 28 percent made errors when using OR. Interestingly, only 19 percent made errors with NOT and AND together, but a negligible number of subjects were brave enough to even try it.

To try to help solve these problems, Brinck et al. (2002) recommend several search engine capabilities, such as:

- Fuzzy search, where related items are also found;
- Spell-correction, where the engine is tolerant of less than perfect spelling;
- Alternate spellings, where the engine finds words spelled in equally acceptable ways;
- Synonyms, resulting in hits for words that mean the same as the one entered by the user;
- Stem words, where hits will be found with or without a suffix or prefix (for example, for singular and plural forms);
- “Stop words” that are not indexed, such as “and” and “to.”

User searching involves five phases (Shneiderman and Plaisant, 2005, p. 566):

- Formulation of the search expression,
- Initiating Action (submitting the expression or triggering the search),
- Review of results,
- Refinement of the expression if necessary.
- Use of the expression by various means, and for future reuse and modification

Finally, Wildemuth (2003) provides a review of studies that focus on the design of search interfaces and shows how practice has already been influenced by those findings. Particularly interesting is the adoption by many engines of the implicit AND when multiple terms are entered. In her studies, Wildemuth showed that the most frequently used strategies were the simplest, using AND. In contrast, OR was very seldom used.

Implications of the work on search engines are as follows:

- Do not assume that users are highly skilled at searching.
- When an operator between terms is absent, assume AND.
- It would be more helpful to provide additional search features, so that users are not restricted to the use of one particular word that is spelled one way.
- Support all four phases of the entire search process, especially allowing a user to refine the search.

Response Time

Perhaps the most troublesome aspect of Web navigation is the delay imposed when moving from page to page. Several recent studies have determined that delay is one of the worst threats to e-commerce (McKinney and Zahedi, 2002; Rose et al., 2001; Torkzadeh and Dhillon, 2002; Turban and Gehrke, 2000), seriously interfering with a site’s usability (Dellaert and Kahn, 1999; Straub et al., 2002). Delays have many causes (Nah, 2002), and it is not likely that they will disappear any time soon. The weakest link dictates the delay, whether it be the user’s connection speed, congestion at the server, or server misconfiguration.

Several decades ago, Miller (1968) suggested that a response time longer than two seconds makes the interchange between user and computer less conversational. Galletta et al. (2004) reviewed the literature and found several different recommendations for maximum delay, ranging from Miller's two seconds to twelve seconds and greater. Zona's (2001) claim is the most intriguing of all: Web sites that match or exceed a delay of eight seconds are so often abandoned by users that the cost to the economy is estimated to be about $21 billion.

Improving page load speed from eight seconds and higher to between two and five seconds has doubled the traffic of some sites (Wonnacott, 2000). Further, an analysis of the most popular sites led Nielsen (1999) to conclude that the enhanced usability of a fast site leads to increased usage.

An empirical study by Galletta et al. (2004) found evidence that user attitudes, behavioral intentions, and performance all exhibited a pattern similar to that found in Figure 6.3. Large declines in the outcome (behavioral intentions as shown in Figure 6.3) occurred immediately for subjects who experienced even a two-second delay, and outcomes of subjects in treatments with longer and longer delays continued to show lower values, but at a declining rate. The study indicated that delay is not tolerated very well, except in the case of a site with familiar organization, where there is some evidence that users are somewhat more tolerant of delay.

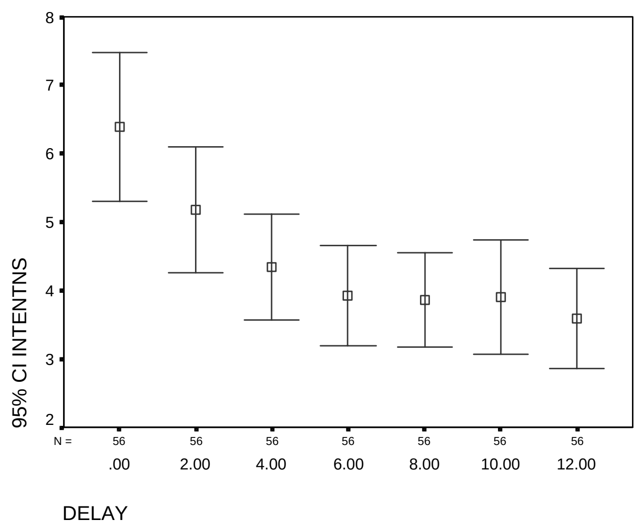


Figure 3. An Example of Declining User Reactions with Longer Delays

Lessons for the designer are as follows:

- A user's patience wears thin very quickly, so make the page load as quickly as possible.
- A user will have more patience when the site uses familiar categorization schemes in following hyperlinks to the goal.

Depth and Breadth of a Site

A designer can choose to display few links per page with several levels, or a large number of links per page, with fewer levels. The most well-known studies on site depth have been performed by Miller (1981), Snowberry et al. (1983), Kiger (1984), and Jacko and Salvendy (1996). These and other studies show us that a constant number of menu nodes can be decomposed into several different structures. For example, sixty-four items can be placed into two items per menu for six levels, four items per menu for three levels, or eight items per menu for two levels.

In general, the research indicates overwhelming support for broad menus with many items. However, Larson and Czerwinski (1998) found, in applying depth versus breadth trade-offs to the Web, that limits to breadth are desirable. Designers were advised to be aware of the complexity of a list of hyperlinks and consider both layout and semantics on a page to minimize the need for scanning the list of potential links. As mentioned before, a "scent" can be offered that makes the trail more obvious, as if a person could "sniff" where to go next, like a hunting dog.

A deep site is usually less desirable because it forces the user to click through the structure and suffer many page loads and their accompanying, inevitable delays. Galletta et al. (2005) found an interaction between site depth and delay. When a site is deep, the ill effects are not as serious when delay is short. When a site is slow, making it broad (less deep) helps.

In summary, designers should therefore:

- Design broad, rather than deep, sites;
- Organize the links into meaningful, easily scanned sets;
- Deepen a site only when delay is short.

Visualization of Information

Understanding how users visualize information is multifaceted and crucial. Some of the research has addressed how information should be entered on forms that are organized into sections. Some studies address the layout of reports on the screen. Other research examines how to allow a person to browse such a large number of items that not all links to those items can fit on the screen at once. Finally, research has provided guidelines for displaying results of data analysis.

Forms are often problematic. Poor forms make use of inappropriate controls, provide inadequate feedback, or do not work at all. Good forms are short, identify required fields, provide an explanation of the purpose of each form and field, and provide a clear understanding of how to submit the information from the form (Brinck et al., 2002).

Reports can be provided in text and/or graphical format. Research has shown consistently that the choice of tabular or graphical output should focus on matching the representation to the task, making for "cognitive fit" (Vessey and Galletta, 1991). The primary thrust is that there should be consonance among three factors: the user's cognitive skills, the task, and the representation of the task (as presented to the user). If there is a match, the proper mental representation will be formed, and proper solution of the problem will be possible.

Cognitive factors and aesthetics combine in Preece's (1993) guide for usability. Preece recommends:

- Only providing information that is necessary.
- Grouping information by color coding, locations and borders, or boldfacing.
- Highlighting important information so users do not have to hunt for it or draw incorrect conclusions.
- Standardizing screen displays so that users do not become confused.
- Presenting text in readable format, especially by using upper- and lowercase, using left-justification, and making sure that the space between lines is equal to or slightly greater than the characters themselves.
- Using graphics that match the user's task.
- Providing icons that work in a consistent manner.
- Using realistic colors to represent real objects and symbolic colors to represent common interpretations (for example, red would mean stop), or, lacking meaning, to differentiate objects (without overloading the user). (pp. 70-77)

Tullis (1998a, b) provides heuristics for evaluating the design of a textual screen, as well as a program that calculates an overall evaluation that correlates highly with users' subjective evaluations. The software evaluates five measures when provided a text file containing the screen layout as input:

- Ratio of uppercase characters to total characters
- Overall density (number of characters filling the screen as a percent of total spaces)
- Local density (average number of filled spaces in a five-degree visual angle around each character, weighted by distance from the character)
- Grouping (number of groupings that are made evident by calculating distances among the characters)
- Layout complexity (whether or not information appears in neat columns and/or rows)

Another view is primarily concerned with aesthetics. Many Web designers have technical backgrounds but have little training in visual design and typography.

McCracken and Wolfe (2004, p. 83) propose four principles of visual organization of a Web page:

- Proximity (A designer should know that when people see a small distance between items, they infer that there is a relationship.)
- Alignment (Place related items along an imaginary line.)
- Consistency (Make related items look similar.)
- Contrast (Make unrelated items look different.)

McCracken and Wolfe also point out that the printed page should not be a definitive guide for design of a Web page, and review characteristics of font size, shape, alignment, color, and other characteristics that apply to Web design. In general:

- Legibility is aided by several characteristics of the typeface, such as large size, variation in case, and high contrast with the background color.
- Many aspects are not in the control of the designer, such as the monitor's resolution, the window's size, text size, condition of the monitor, and available fonts.

It is often difficult to squeeze a large amount of information onto a small screen. Yet people want to have a large number of choices immediately at hand. There are many innovative strategies that have been proposed. Borges et al. (1997) provide a very specific list of recommendations for Web page layouts, for main ("home") pages as well as subordinate pages. With a series of usability tests, they evaluate and adjust their recommendations. Shneiderman (1998) reviews ways in which information that is not at the center of focus can be made into an abstract representation. He describes techniques such as color coding, location coding, a movable image of a magnifying glass with magnified text inside and a large amount of tiny text outside, using pinpoints on the screen, using markings in a 3-D representation of a navigable room, or shrinking into progressively smaller and smaller fonts until they collapse into lines and dots (fisheye display).

These disparate studies and visualization elements lead to the following implications:

- Provide short forms with clear explanations, labeling, and feedback.
- Provide output that matches the user's skills and the task..

- Several heuristics can enable a user to understand what is on a screen, such as simplification, grouping, and consistency.
- Case, location, density, and complexity can provide strong cues about grouping.
- Consistency and contrast are tools to indicate whether items are related or not.
- Many aspects of legibility are under the designer's control, but many others are not.
- Tools such as color coding, 3-D displays, and fisheye displays can help users wade through a large amount of information.

Mental Models

Occasionally, users have surprisingly inaccurate mental models. Such models have been studied in several fields, especially physics (Mayer, 1988). Inaccuracies in understanding physics become dangerously obvious when drivers point their steering wheel in the wrong direction when slipping on ice, just as inaccuracies in understanding terminology result in searches that are irrelevant or incorrect.

Mayer (1988) categorizes knowledge into that which is syntactic (knowledge of elements and how they are combined), semantic (knowledge of meaning), schematic (knowledge of approaches to problem solving as a single "chunk"), and strategic (knowledge of how to devise and monitor solution plans).

Galletta (1985) performed an experiment to build first-time e-mail users' schematic knowledge by providing informal sketches of a hotel messaging system; such a sketch was intended to invoke their "schemata" about such systems and then infer the need for and role of corresponding functions of the e-mail system. The group members who were presented with the sketches and a five-minute explanation were able to recall more elements, capabilities, and procedures of the e-mail system than the group not presented with the schematic aid.

The lack of proper mental models can provide sometimes humorous situations, as seen in common experiences of multiple computer companies. Many computer company help lines have received photocopies of floppy disks in the mail (Carlton, 1994) and calls asking "Where is the 'any' key?" Some resources have compiled such stories. *Computerworld* magazine has a "Shark Tank" column on the final page that reveals many misconceptions of technology. Rinkworks (2005) has a "Computer Stupidities" site that organizes the stories by topic. To prevent the expansion of these resources, it is rather clear that some heuristics should be followed:

- Do not make any assumptions about the Web shopper's knowledge or experience.
- Make instructions very clear and explicit.
- Build schematic knowledge whenever possible by employing useful analogies.

Designing the Site

Once the designer has built a reasonable level of understanding of trade-offs and user knowledge, the system is designed, built, and tested. There are several tools available to the designer.

Storyboards and Checklists

Movie directors have found that, before creating the actual product, it is very important to sketch a list of scenes in the story's sequence. Likewise, Web developers have found that the Web pages and their structure can be sketched informally, providing easier visualization of the Web site to be built. The resultant "storyboards" can be tacked up or placed on overheads and discussed by a design group. Software is also available to allow storyboard creation and manipulation. However they are developed, it is easier to make adjustments of the design on the storyboards before building the site, rather than make adjustments of the design after building the site.

Checklists are useful in complex tasks such as auditing and aviation to make sure no steps have been omitted inadvertently. Design goals and tasks can be listed, and, for each development task, a checklist can be completed to determine if any important steps have been skipped.

An illustration of the design and usage of storyboards and sets of sample checklists are available in Brinck et al. (2002).

GOMS and the Keystroke Model

No model has been more influential than the GOMS model developed by Card, Moran, and Newell (1983). GOMS stands for Goals, Operators, Methods, and Selection Rules, which are used to drive design and evaluation of a system. Rather than a single technique, GOMS represents a family of techniques that are positioned at various levels of granularity. However, most researchers have used the "Keystroke Model," which allows a designer to estimate the specific subtasks (such as typing, inserting text, and removing text) needed to accomplish a task (such as producing a document or executing a transaction) by an expert user. These subtasks can be further decomposed into micro-level acts, such as individual key presses, repositioning of the hand to the mouse or back to the keyboard, or simply having to think. From those acts, the designer can use standard estimates of time required for those acts, and then can estimate how long it would take to perform the given task.

Estimates of time required by expert users for common acts, from Card et al. (1993), can be found in Table 6.1.

Table 1. Average Time Requirements for Selected Actions (from Card et al., 1980)

Symbol	Act	Time Required (sec)
K	Keystroke	.2 (average typist)
P	Pointing to a target with a mouse	1.1
M	Mental preparation for a physical act	1.4
H	"Homing" the hands on the keyboard	.4

Although the original Card et al. book describes in detail the entire family of models, many other researchers have focused on both methods and applications of GOMS. Kieras and John have studied a variety of methodological issues (see John, 1996a, b; Kieras, 1988) and proposed a notation. Gray et al. (1992, 1993) applied the Keystroke Model to a proposed system for use by telephone operators at NYNEX. The results underscore the importance of testing a design. The new system was designed following several principles of good design, making use of a fancy graphical interface, animations, and special keys for repetitive command combinations. However, the new system would have cost $2.4 million extra if deployed. After decomposing some benchmark tasks to the bottom-level acts needed, it was found that the new system had needlessly eliminated acts that occur during slack time, and paradoxically moved one keystroke from a slack period to the critical path. Using project management software to perform what they call CPM-GOMS, the researchers predicted the results of three months of testing, and the new system was not adopted.

The GOMS model is perhaps the most mature of all of the models available in the literature, and scores of empirical and conceptual studies still cite it. An e-commerce perspective certainly has concern for issues not included in the model, such as site visitation by intermittent or one-time users, the attitudes of the users, intentions to return to a site, or actual purchases. However, useful implications for the Web designer can still be drawn, such as:

- Usage can be modeled, minimizing expenditures of user time.
- Time required is a useful surrogate for effort and clumsiness of a design.
- Following principles alone might not be enough.

CONCLUSIONS

Given the extensive set of findings for enhancing usability of pre-Web software applications, poor Web designs can be prevented. Many pre-Web recommendations are also quite useful for designing Web applications.

This article has asserted that both developers and users are less experienced in Web applications than developers and users of the organizational applications of yesterday. The lack of user experience dictates that organizations make usability more, not less of a priority today, unfortunately at a time when designers seem to be less experienced as well. Furthermore, the task of Web design has an impact on several functions within an organization, so there is need for wide input and collaboration across those functions.

One of the most important guidelines provided by the literature is that of testing. Users and their tasks should be identified and target users should test the system for additional adjustments, if necessary. In some organizations, making testing a routine practice would require a dramatic change in organizational culture. Developers also would need to be prepared to make changes quickly and then test again, repeating the process as needed until adequate levels of usability have been achieved.

To provide quicker arrival at an adequate design, several areas of understanding and several tools are available. It is helpful to build awareness of Fitts' Law, impacts of terminology used in organizing a site, searching limitations, ill effects of page loading delay, trade-offs between site depth and breadth, options for what is seen by the user in inputs and outputs, and problems with users' mental models. The use of storyboards, checklists, and evaluation models can be instrumental in preventing design errors, such as overloading a site with graphics, failing to provide useful information, and confusing the user.

Stated another way, there is no longer an excuse for a site that confuses a user, requires more steps than is necessary, or fails to provide explicit prompts. If we merely become aware of interface design research that has been conducted over the last few decades, we will undoubtedly embark on the road to better Web design. Application of the findings of that research will help us arrive.

REFERENCES

Borges, J.A.; Morales, I.; and Rodriguez, N.J. Page design guidelines developed through usability testing. In C. Forsythe, E. Grose, and J. Ratner (eds.), *Human Factors and Web Development..* Mahwah, NJ: Lawrence Erlbaum Associates, 1997, pp. 137–152.

Brinck, T.; Gergle, D.; and Wood., S.D. *Designing Web Sites that Work: Usability for the Web.* San Francisco: Morgan Kaufmann Publishers, 2002.

Card, S.K.; Moran, T.P.; and Newell, A. *The Psychology of Human-Computer Interaction*. Hillsdale, NJ: Lawrence Erlbaum Publishers, 1983.

Carlton, J. Computers: befuddled PC users flood help lines, and no question seems to be too basic. *Wall Street Journal,* (March 1, 1994), B1.

Davis, F.D.; Bagozzi, R.P.; and Warshaw, P.R. User acceptance of computer technology: a comparison of two theoretical models. *Management Science,* 38, 8, (1989), 982–1003.

Dellaert, B.G.C., and Kahn, B.E. How tolerable is delay?: consumers' evaluations of Internet Web sites after waiting. *Journal of Interactive Marketing,* 13, 1 (1999), 41–55.

Dumais, S. Textual information retrieval.In M. Helander (ed.), *Handbook of Human-Computer Interaction..* Amsterdam: North-Holland, 1988, 673–700.

Fitts, P.M. The information capacity of the human motor system in controlling amplitude of movement. *Journal of Experimental Psychology,* 47 (1955), 381–391.

Furnas, G.W. Design in the MoRAS. In J.M. Carroll (ed.), *Human-Computer Interaction in the New Millennium.* New York: ACM Press, 2002, 53–73.

Galletta, D.F. *A Learning Model of Information Systems: The Effects of Orienting Materials, Ability, Expectations, and Experience on Performance, Usage, and Attitudes.* Unpublished PhD Thesis, University of Minnesota, 1985.

Galletta, D.; Henry, R.; McCoy, S.; and Polak, P. Sensitivity to web delays: a contingency analysis of users' tolerance for slow web sites, *Journal of AIS.* Volume 5, Issue 1, Article 1, 2004.

———. When the wait isn't so bad: the interacting effects of Web site speed, familiarity, and breadth. Working Paper, University of Pittsburgh, Pittsburgh, PA, 2005.

Gould, J.D.; Boies, S.J.; Levy, S.; Richards, J.T.; and Schoonard, J. The 1984 Olympic message system: a test of behavioral principles of system design. *Communications of the ACM,* 30, 9 (1987), 758–769.

Gould, J.D., and Lewis, C. Designing for usability: key principles and what designers think. *Communications of the ACM,* 28, 3 (1985), 300–311.

Gray, W.D.; John, B.E.; and Atwood, M.E. The precis of Project Ernestine or an overview of a validation of GOMS. In *Proceedings of the SIGCHI Conference on Human Factors in Computing Systems.* Monterey, California: Association for Computing Machinery, 1992, pp. 307–312.

———. Project Ernestine: validating a GOMS analysis for predicting and explaining real-world performance. *Human-Computer Interaction* 8, 3 (1993), 237–309.

Hipple, S., and Kosanovich, K. Computer and Internet use at work in 2001. *Monthly Labor Review,* 126, 2 (2003), 26–35.

IBM. Accessibility at IBM. (2005). International Business Machines, http://www-3.ibm.com/able/access_ibm/reasons.html.

Jacko, J., and Salvendy, G. Hierarchical menu design: breadth, depth, and task complexity. *Perceptual and Motor Skills,* 82 (1996), 1187–1201.

Jansen, B.J.; Spink, A.; and Saracevic, T. Real life, real users, and real needs: a study and analysis of user queries on the web. *Information Processing and Management* 36 (2000), 207–228.

John, B.K., and Kieras, D.E. Using GOMS for user interface design and evaluation: which technique? *ACM Transactions on Computer-Human Interaction,* 3, 4 (1996a), 287–319.

———. The GOMS family of user interface analysis techniques: comparison and contrast. *ACM Transactions on Computer-Human Interaction,* 3, 4 (1996b), 320–351.

Jones, B. *Assessing The Effect of Web-Site Control Differentiation on Single Step Navigation.* Unpublished PhD Thesis, University of Pittsburgh, 2003.

Kieras, D. 1988. Towards a practical GOMS model methodology for user interface design. In M. Helander (ed.), *Handbook of Human-Computer Interaction.* Amsterdam: Elsevier Science Publishers, 1988, pp. 136–157.

Kiger, J. The depth/breadth trade-off in the design of menu-driven user interfaces. *International Journal of Man-Machine Studies,* 20 (1984), 201–213.

Koyani, S.; Bailey, R.W.; and Nall, J.R. *Research-Based Web Design & Usability Guidelines.* Washington, DC: National Cancer Institute Publication No. 03–5524, 2003. (available at http://usability.gov/guidelines).

Landauer, T.K. *The Trouble With Computers.* Cambridge, MA: The MIT Press, 1995.

Larson, K., and Czerwinski, M. Web page design: implications of memory, structure, and scent for information retrieval. In *Proceedings of the SIGCHI Conference on Human Factors in Computing Systems.* Los Angeles, CA: Association for Computing Machinery, 1998, pp.18–23.

MacKenzie, I.S. Movement time prediction in human-computer interfaces. In *Proceedings of Graphics Interface '92.* San Francisco, CA: Association for Computing Machinery, 1992, pp. 140–150.

Marchionini, G. *Information Seeking in Electronic Environments.* Cambridge: Cambridge University Press, 1995.

Maxwell, K. The maturation of HCI: moving beyond usability toward holistic interaction. In J.M. Carroll (ed.), *Human-Computer Interaction in the New Millennium.* New York: ACM Press, 2002, pp. 191–209.

Mayer, R.F. From novice to expert. In M. Helander (ed.), *Handbook of Human-Computer Interaction.* Amsterdam: North-Holland, 1988, pp. 569–580.

McCracken, D.D., and Wolfe, R.J. *User Centered Website Development.* Upper Saddle River, NJ: Pearson Prentice-Hall, 2004.

McKinney, V.,Yoon, K. and Zahedi, F. The measurement of Web-customer satisfaction: an expectation and disconfirmation approach. *Information Systems Research,* 13, 3 (2002), 296–315.

Miller, D.P. The depth/breadth tradeoff in hierarchical computer menus. In *Proceedings of the Human Factors Society 25th Annual Meeting.* Santa Monica, CA: Human Factors and Ergonomics Society, 1981, pp. 296–300.

Miller, R.B. Response time in man-computer conversational transactions. In *Proceedings of AFIPS Fall Joint Computer Conference Proceedings.* Montvale, NJ: AFIPS Press, 1968, pp. 267–277.

Nah, F. Study of Web users' waiting time. In V. Sugumaran (ed.), *Intelligent Support Systems Technology: Knowledge Management.* Hershey, PA: IRM Press, 2002, pp. 145–152.

Negroponte, N. Invited Lecture, J.J. O'Hare. *SIGCHI Conference on Human Factors in Computing Systems.* Washington, DC: Association for Computing Machinery, 1988.

Nielsen, J. 1999. "Who Commits the 'Top Ten Mistakes' of Web Design?" http://www.useit.com/alertbox/990516.html

Nielsen, J., and Landauer, T.K. A mathematical model of the finding of usability problems. In *Proceedings of INTERCHI'93, ACM Conference on Human Factors in Computer Systems.* New York, NY, 1993, pp. 206–213.

Nielsen/Netratings. Global internet population grows an average of four percent year-over-year. Nielsen Net Media, 2003 http://www.nielsen-netratings.com/pr/pr_030220_hk.pdf.

Norman, K.L., and Chin, J.P. The effect of tree structure on search in a hierarchical menu selection system. *Behaviour and Information Technology,* 7 (1988), 51–65.

Paciello, M.G. *Web Accessibility for People with Disabilities.* Lawrence, KS: CMP Books, 2000.

Pirolli, P. Computational models of information scent-following in a very large browsable text collection. In *Proceedings of CHI 97: Conference on Human Factors in Computing Systems.* Atlanta, Georgia: Association for Computing Machinery, 1997, pp. 3–10.

Preece, J., ed. *A Guide to Usability: Human Factors in Computing.* Wokingham, England: Addison-Wesley Publishers, 1993.

Rinkworks. Computer stupidities, available at http://rinkworks.com/stupid/

Robertson, G.; McCracken, D.; and Newell, A. The ZOG approach to man-machine communication. *International Journal of Man-Machine Studies,* 14 (1981), 461–488.

Rose, G.M.; Lees, J.; and Meuter, M. A refined view of download time impacts on e-consumer attitudes and patronage intentions toward e-retailers. *The International Journal on Media Management,* 3, 2 (2001), 105–111.

Shneiderman, B. *Designing the User Interface,* 3 ed. Reading, MA: Addison Wesley Longman, Inc., 1998.

Shniederman, B. and Plaisant, C. Designing the User Interface, 4 ed. Boston, MA: Pearson Addison Wesley, Inc., 2005

Smith, S.L., and Mosier, J.N. *Guidelines for Designing User Interface Software,* EDS-TR86 –278. Bedford, MASS: The MITRE Corporation, 1986. (Now available at http://dfki.de/~jameson/hcida/papers/smith-mosier.pdf)

Snowberry, K.; Parkinson, S.R.; and Sisson, N. Computer display menus. *Ergonomics,* 26, 7 (1983), 699–712.

Straub, D.W.; Hoffman, D.J.; Weber, B.W.; and Steinfield, C. Measuring e-commerce in net-enabled organizations: an introduction to the special issue. *Information Systems Research,* 13, 2 (2002), 115–124.

Torkzadeh, G., and Dhillon, G. Measuring factors that influence the success of Internet commerce. *Information Systems Research,* 13, 2 (2002), 187–204.

Tullis, T.S. Screen design. In M. Helander (ed.), *Handbook of Human-Computer Interaction.* Amsterdam: Elsevier Science Publishers, 1988a, pp. 377–411.

———. A system for evaluating screen formats. In H.R. Hartson and D. Hix (eds.), *Advances in Human-Computer Interaction.* Norwood, NJ: Ablex, 1988b, pp. 214–286.

Turban, E., and Gehrke, D. Determinants of e-commerce Web sites. *Human Systems Management,* 19 (2000), 111–120.

Venkatesh, V.; Morris, M.G.; Davis, G.B.; and Davis, F.D. User acceptance of information technology: toward a unified view. *Management Information Systems Quarterly,* 27, 3 (2003), 425–478.

Vessey, I., and Galletta, D.F. Cognitive fit: an empirical study of information acquisition. *Information Systems Research,* 2, 1 (1991), 63–84.

W3C (World Wide Web Consortium). Web Content Accessibility Guidelines 1.0, 1999. http://www.w3.org/TR/WCAG10/

Watchfire.com. Bobby. Available at bobby.watchfire.com/.

WHO. Health Report 1997 Executive Summary. World Health Organization, 1997. http://www.who.int/whr2001/2001/archives/1997/exsum97e.htm

Wildemuth, B.M. Evidence-based practice in search interface design. In *Proceedings of CHI 2003 Search Usability Workshop*. 2003.

Wonnacott, L. When user experience is at risk, tier 1 providers can help. *Infoworld,* 2000.

Zona Research. The need for speed II. *Zona Market Bulletin.* Redwood City, CA: Zona Research, Issue 5, April 2001 http://www.keynote.com/downloads/Zona_Need_For_Speed.pdf