BROWSERS, PLATFORMS, AND MONITORS, OH MY!!
MAINTAINING COMPATIBILITY ON THE YELLOW BRICK ROAD OF WEB DESIGN

by
Harry Ahlas

A Master's paper submitted to the faculty
of the School of Information and Library Science
of the University of North Carolina at Chapel Hill
in partial fulfillment of the requirements
for the degree of Master of Science in
Information Science.

Chapel Hill, North Carolina

April, 2001

Harry Ahlas. Browsers, Platforms, and Monitors, Oh My!! Maintaining Compatibility on the Yellow Brick Road of Web Design. A Master's paper for the M.S. in I.S. degree. April, 2001. 63 pages. Advisor: Gary Marchionini

This study describes data collected regarding design decisions Web developers make in order to make their interfaces and content compatible across user environments. Qualitative data was gathered through interviews with developers across the United States. Quantitative data was collected through an examination of several Web sites' source code.

Web content does not render the same in every environment. A site may have an attractive interface when viewed through one browser while being completely incomprehensible in another browser. Web developers face the complex task of deciding what types of environments to design for. This research explores the current trends and standards that developers implement to achieve compatibility.

Headings:

World Wide Web — Accessibility

Information systems — Design

Internet

Electronic Commerce — Standards

Table of Contents

Problem Statement and Introduction

What strategies and reasoning do Web designers implement in order to compensate for the countless combinations of computer platforms, browsers, monitors, and plugins that the world utilizes?

One of my favorite Web sites is espn.com, where I can check up-to-the-minute scores and updates on my favorite teams' games. Over this past summer, my work provided me with an older Macintosh computer that had an early version of Netscape Navigator as its browser. I would periodically check espn.com for scores and often I found that several of the site's Hypertext Markup Language (HTML) tables had broken so that the pages' appearances were distorted. Much of the text was cut off. These miscues were related to my computer's improper handling of layers. I looked at the source code and found that espn.com had attempted to make its site readable on all platforms and browsers—particularly Navigator and Microsoft Internet Explorer. Despite the site's use of JavaScript to alleviate the problems surrounding the different platforms and browsers, the pages still looked funny.

This problem of differences in platforms, browsers, versions, and monitors is a difficult barrier to hurdle. I interviewed for a job at a prominent Web design company in the Bay Area who kept telling me how important browser/platform compatibility is to a good Web site. Later that day, I

checked out some of the sites the company had designed and noticed that an HTML table on one of their sites had broken on my personal computer at home (and my personal computer was much newer than the Macintosh at work!).

It is almost impossible for a Web site to look exactly the same on all graphics-enabled browsers. Browsers are not the only elements of the client's machine that interpret Web pages' appearances. The client's operating system as well as his monitor size and resolution also play a significant role in the final appearance of a Web page. Furthermore, not all users have the same plugins, namely the Flash player, which enables the use of fancy motion graphics on Web pages; users without this plugin cannot view Flash content. My research delved into this problem of interoperability and analyzed how different Web sites address the issue. What are the various methods that Web designers employ? What factors motivate their decisions? There is no correct "by-the-book" method of designing complex and cross-compatible Web sites; this research probed into the various methods and evaluated patterns that designers follow.

Relevant Literature

In reading about browser and platform compatibility, I have found a few significant problems. First, books related to my topic are often outdated. Even the newest books seem to be behind the times. Because books relevant to Web compatibility quickly become dated, much of the literature I cite comes from an alternate source—Web resources. This leads to a couple other problems. Web pages usually do not have dates and their content can be inaccurate. Additionally, Web authors often do not cite their sources well nor do they thoroughly discuss their methods.

For these readings, I sought out a variety of views with different types of books and articles. The readings provided me with a solid foundation for understanding the problems that Web designers have with browser and platform compatibility. Several of the authors I read seemed to ignore my question entirely, which made the reading even more interesting. Knowing that respected authors in the large field of Internet studies ignore compatibility reinforced my desire to research the subject. What follows is a review of various readings relevant to my research.

One of the first topics Holzschlag (1998) discusses is cross-browser and cross-platform design issues. While her writing on these considerations is not entirely comprehensive, she makes the extra effort to stress their important role in good Web design. Because her discussion of the different platforms and browsers is concise and not extremely detailed, it serves as a good general introduction to the potential problems that can result from the

neglection of browser and platform compatibility. Her strength comes in clearly listing the general potential problems with compatibility. For instance, she lists the major platforms—PC, Macintosh, and UNIX—as well as some of the minor (but still important) platforms—VMS, Sun/SGI, and Linux (note that in 1998 Linux was considered *minor*). Holzschlag also touches on the differences between using HTML text versus graphical text (i.e. saving text as GIFs), and how they play into browser and platform compatibility.

By far the most important part of her work for my purposes is a very thorough "Cross-Browser Tag and Attribute Support Table" which pits the various up-to-date HTML tags on one axis against the three major browsers. Here is an example of one line of the table:

HTML Tag:	Internet Explorer Versions:	Netscape Navigator:	Lynx:	HTML versions:	Author:
<APPLET>	*3/4	*2/4	*	*3+/4	W3C

(145)

This format is easy to read and a very helpful reference for someone looking to see what HTML tags work with what browsers.

There are two limitations to Holzshlag's work. First, her Tag and Attribute Support Table focuses only on three different browsers, leaving out statistical columns for other significant browsers. Second, she fails to state how she attained the data for the table, making it difficult to judge its reliability and validity. I had to personally contact her to find out that she collected the data from the World Wide Web Consortium (W3C).

Niederst's work (1996) is a more simplistic look at Web design issues. While her book is good for looking into simple cross-browser and cross-platform facts, its content is too outdated to apply *directly* to my work (as you will see, however, this does not mean that her book is not worth reading!). Four years is an eternity on the Internet, and I suspect that four years from now my Master's thesis will too be more of an historical artifact than a technical guide. For instance, she favors the Mosaic browser, which is presently very sparsely used, as her main browser example. Moreover, Neiderst's writing only applies to HTML 2.0, which does not even include tags for supplying a page's body background color, an absolutely necessary feature for almost all of today's Web designers.

Still, there are several positive contributions made by this book as well as the other outdated literature on the topic. First and foremost, this book provides a historical perspective on why and how the different browsers evolved the ways they did. There seems to be such a rush with the latest books that these historical perspectives on the evolution of browsers are left out until the Internet's escalation slows to enough of a non-chaotic pace that historians can catch up with this growth. For this reason, I will not discard literature on this subject simply because it may be obsolete. Instead, I can utilize this older information to understand the past perspectives of designers. Newer books rarely discuss issues of the past, perhaps because their readers are only interested in the latest and most up-to-date design techniques. Furthermore, Niederst provides basic HTML information that newer books tend to skip over, such as how different browsers and operating

systems display font sizes in <H1> tags. It is amazing how the older books provide a basic foundation for my research in a way that the newer books have not made available.

So far I have uncovered two types of books for my research: new books with solid, accurate information (Holzshlag 1998); and older books with outdated information, but with historical perspectives that cannot be easily found in newer books. Vitanza (1998) and Sinclair (1999) provide us with another type of book: newer text that has out-of-date, inaccurate information and little or no important historical perspective.

Sinclair's work is a 500-page book that discusses all types of Web typography that only has one paragraph regarding browser compatibility. It states how Webmasters have only partial control over the way their content is presented. This book failed in the same way that Vitanza's book fails; almost complete disregard for my subject.

Vitanza (1998) has little grasp on the importance of operating system compatibility issues and only enough of a grasp of browser compatibility issues to generalize it into two vague sentences. Here are two examples of problems I have with his work: 1) he does not discuss the Internet Explorer browser, despite the fact that this book was published in 1998; 2) he claims that using the HTML code * * is more efficient at creating a text-indent on a Web page than using a GIF image. Furthermore, he denigrates graphic designers who use the GIF method, stating that their HTML skills are inadequate because they do not know how to use * *. In actuality, the GIF method provides much more consistency between browsers: a GIF indent on

Netscape will look much more like a GIF indent on Internet Explorer than an * * will look on the two. Clearly, this author is ignorant of the issues I intend to discuss in my research.

Interoperability has not become a crucial element of Web design until fairly recently. The fact that books such as that of Sinclair and Vitanza exist illustrates the importance of my research. Even Web "experts" have a lot to learn about this growing industry.

Stephanie Redman (1999) covers Web Design from a creative standpoint. She focuses on how to make pages look attractive, and strays from the inevitable technical jargon that designers must use. She discusses Web colors in as much detail as Vitanza talks about writing, only Redman seriously acknowledges the importance of paying attention to browser and platform compatibility. She does not delve far into the compatibility subject matter; instead, she leaves it up to the reader to learn about it from other sources:

> Do you know every difference between every version of every browser with regard to every Java, JavaScript, ActiveX, animated GIF, table layout, frame design, download method, XML, SGML, PC vs. Mac display/load/transfer consideration? Neither do the experts. (16)

She also reminds the reader that even if you do know these differences, they change all the time and are nearly impossible to keep up with. Her concise contribution to my research is short but sweet. Redman illustrates an important point that I must consider in my data collection and analysis—*in choosing a particular compatibility scheme, do designers consider the advent of future technologies?* Perhaps some designers choose a particular

interoperability method because it will likely work with future multimedia technologies that their sites may implement. Is there a danger of choosing a method that will not work with certain technologies that will soon become standard (such as Flash or XML)?

A more detailed work, Jones (1997), is one of the most helpful of all the readings I found. He supplies a full chapter on browser support especially regarding Cascading Style Sheets (CSS). Furthermore, he talks about future issues with CSS. Jones also supplies a significant amount of history about Style Sheets and browser issues, which ties in nicely with Neiderst (1996). He also uses concrete examples to support his work. For instance, he provides an actual list of problems with CSS on Internet Explorer 3.0—the list was actually created by Microsoft.

More importantly, Jones points out some of the techniques that designers use to solve browser (and platform) compatibility problems. He suggests using "dynamic, database-driven content" that can easily be created on the fly for delivery to any browser through "Browser Sensing." Browser Sensing involves detecting what browser type the client is using to create Web pages customized for the client's particular browser. Additionally, Jones presents another option for compatibility: "Hybrid Web Design," which I will discuss in detail later. He offers a question that I hope to help answer in my research: *Who takes advantage of these options in dealing with compatibility?*

The World Wide Web Consortium (W3C 2000) provides additional depth in the use of Cascading Style Sheets (CSS), differentiating its role in

Web design versus the role of HTML. The author emphasizes that designers should use HTML only to structurally mark up their Web pages. Often, designers try to control their sites' layouts using HTML—doing so can lead to cross-compatibility problems. The W3C states that CSS should be used for layouts because it provides consistency that HTML alone cannot achieve.

The W3C article is very basic, providing the reader with a concise overview of the interactivity between HTML and CSS. It makes solid arguments about general points. On the other hand, it does not attempt to discuss the particulars of CSS that Jones details. Also, it fails to compensate for Web clients whose browsers do not understand CSS.

Mulder and Brandt (1999) present a helpful and more detailed article on browser and platform compatibility. The article's best asset comes in the form of a table that presents browsers and platforms versus various Web technologies. The Y-axis of the table hosts almost every browser available, dividing them between platforms. For instance, Internet Explorer 5.0 has three different rows on the Y-axis (one for PC, one for Mac, and one for Unix). It includes data for PC, Mac, Unix, Linux, Television (WebTV), NextStep, and OS/2. The X-axis hosts columns that are reminiscent of Holzshlag's table, only with more generalized entities. Instead of having columns for each individual HTML tag, this table has 13 columns with data for technologies other than HTML such as Java, plug-ins, Style Sheets, and XML. The table illustrates which technologies work in which environments. This table will be very helpful in studying more general decisions that designers make: *Is it worth it to use CSS?* Furthermore, The authors do a solid

job of explaining four major compatibility problems: offset, canvas size, text size, and form elements. In general, the work of Mulder and Brandt is a great reference for designers.

Once again, however, the accuracy of the research is somewhat questionable because the methodology is never made clear by the authors. Also, the main table could have been more detailed and gone into the particular tags that Holzshlag analyzes.

The HTML Goodies design site, authored by Joe Burns (No Date), targets a less-skilled audience than Mulder and Brandt. He takes a much different approach at browser compatibility and ignores platform compatibility for the most part. Burns looks at the different offerings between browsers as an advantage. As opposed to saying, *Avoid doing X because not all the browsers support it*, Burns's attitude is more like *If you use Internet Explorer, you can do X and it's really cool!* Burns's methodology is good because he goes through each step with the reader. His approach presents a good perspective on what the different browsers can do, but this information is not especially helpful for my purposes because it does not deal with how to address compatibility issues.

Siciliano and Boles (2000) focus on interoperability significantly more than Burns. While their work focuses primarily on Dynamic HTML (DHTML, *technically defined as the use of HTML, CSS, and JavaScript on any given Web page) techniques, it supplies good insight on how to deal with browser compatibility that can be related to all types of Web design, not simply DHTML. While it fails to discuss platform issues, it does provide a

cross-browser compatibility checklist as well as a very important chart entitled “The Pain Meter” by Scott Isaacs. The chart is a cost-benefit analysis of the choices designers can make in creating Web sites. On the one hand, a designer can make a highly interactive, graphics-heavy site which can be very attractive for the user; on the other hand, such a site requires a significant amount of extra effort in order to be compatible to the various browsers, and often the technology will not be usable on many browser types: “If you want advanced features, be prepared for a LOT of work to get pages readable by weaker browsers.”

The concepts Boles and Siciliano discuss are very significant to my research. Although the “Pain Meter” is hardly empirical, its value for my purposes is great in establishing different levels of technology for use on the Internet.

In a lecture by Isaacs (No Date), he illustrates several aspects of DHTML and the surrounding compatibility issues. Much of the lecture is not directly relevant to my research, but he makes an important reference to what Jones discussed. Isaacs makes the important point that requests in “Browser Detecting” should be checked on the server side, rather than the client.

Several Web development sites provide articles explaining interoperability techniques. Anderson and Kunicki (2000), supply many useful notes about minor cross-browser HTML problems. For instance, Netscape displays text input boxes very differently from Internet Explorer. Netscape renders the size of an input field using the browser’s default fixed

width font value as a guide while Internet Explorer uses the HTML's current font size as a guide. Usually this will not create a significant error; perhaps the two browsers will render the text boxes with only a few pixels of difference between them. However, it is possible that an entire table could crash on itself if the table's width cannot accommodate one of the two text boxes.

Another Web site I found particularly useful in learning tidbits of cross-environment problems and solutions was webreference.com. Shiran (2001) explains many interoperability solutions—mostly with JavaScript — for a plethora of problematic situations. He divides his brief articles into easy-to-find tips to facilitate the design of cross-compatible sites through JavaScript. Although he does not encompass other mechanisms for compatibility, his work on JavaScript is solid and he seems to be an authority on the JavaScript aspect of compatibility. Shiran focuses not only on Netscape and Internet Explorer problems, but also investigates issues with Macintosh and Windows differences.

Steinman (1998) focuses on DHTML compatibility across browsers. He presents solutions not unlike those of Shiran, but disregards operating system issues for the most part. On the other hand, he gives a solid discussion of CSS and compatibility.

The Macromedia Web site (no Date) provides a convenient article on how to detect whether or not a client has the Flash plugin on his/her browser. The article is easy to follow, but the product of the text is disappointing. The solutions provided by Macromedia are somewhat

inefficient, as they require an extremely large amount of code in several languages in order to assure the adequate detection of Flash.

The preliminary readings helped me in several ways. First, they provided me with a solid awareness of the prime concerns regarding the interoperability of Web sites. Mulder and Brandt (1999) and Holzshlag (1998) have authoritative tables that are easy to reference. Second, the older resources present a unique historical perspective that the newer resources do not discuss. This historical perspective has helped me understand the evolution of the different browsers available for use on the Web. Finally, several readings have introduced me to technologies that can be used as a solution to compatibility issues, particularly Shiran's articles and Jones's (1997) discussion of CSS.

Methodology

For my research I attained both qualitative and quantitative data. Quantitative data came from my own analysis of a variety of Web sites, selected at random. The sites were all available on the World Wide Web; I reviewed no intranet sites, as intranet designers have considerably less issues to deal with regarding interoperability. I gathered qualitative data through interviews of Web content managers, developers, and designers, chosen by opportunistic sampling. In this section, I will go over these two aspects of my methodology and explain how the qualitative data is useful for interpreting the quantitative statistics.

Quantitative Aspect

I collected quantitative data from three different locations. The first was a Windows environment that supported Internet Explorer 5.5, Opera, and Netscape Navigator versions 3.04 Gold, 4.72, and Netscape 6. The second was a Macintosh environment with Internet Explorer 5 and Netscape 4.74. The last environment was a Unix platform utilizing the Lynx browser. I studied a total of 75 sites for quantitative data. I browsed the Yahoo Web site's general categories to randomly select 45 of the sites (roughly 3 sites from each of the Yahoo categories). I also reviewed 5 sites created by people I interviewed. The other 25 I selected out of my own scrutiny and personal Web experience. I chose larger sites such as cnet.com and gm.com that I felt deserved recognition in this research.

There are several types of data that I sought from each site I examined. I collected the data by seeking answers to the following four questions: (1) *What method(s)—if any—does each site employ in addressing interoperability issues?*; (2) *In what environments do the site's method(s) work and in what environments do they not work?*; (3) *Who does the site cater to and how large of an audience does the site have?*; (4) *W hat does the site offer? Services, academic info, business info, sales?* For the second question, one could argue that the term *work* cannot be considered a quantitative type of data. For this research, I will deem the term to mean *I think that the site's designer or content manager is satisfied with its appearance in this particular environment*. This call requires me to use common sense and design rationale. I will point out any ambiguous sites where it is not clear whether or not the site *works.*

The most crucial part of the analysis was evaluating the first two quantitative questions I discussed earlier. The other two questions are be used to give a fuller meaning to the first questions. In analyzing the data, I placed the results from each site into a spreadsheet. The spreadsheet revealed design trends with compatibility issues for the sites I examined. Additionally, I used the spreadsheet to search for patterns that may occur. I expected to find that sites with smaller audiences tend to pay less attention to compatibility.

There was one major problem that I faced in the data collection. It is difficult to tell whether or not a particular site has database-driven content. It is safe to assume that most large-scale sites with constantly updated information utilize some sort of mechanism to generate HTML on the fly,

where the server adds pre-made templates to the content. I initially had hoped to come up with a statistic that revealed what percentage of these database-driven sites also used server-side user agent detection to generate cross-compatible content. Unfortunately, I was unable to determine such a statistic. However, several of my interview subjects suggested that whatever that statistic may presently be, server-side detection is on the rise. They felt that many large-scale sites are beginning to follow the trend of using server-side detection along with database-driven content to generate interoperable HTML. I will discuss this mechanism in greater detail later.

Qualitative Aspect

Interviews were the source of qualitative data for my research. I conducted 11 interviews with subjects from two main locations: the Raleigh/Chapel Hill/Durham Triangle area as well as the San Francisco/Oakland/San Jose Bay Area. Most of the interviews were conducted face-to-face and a few were conducted over the phone. I contacted one interviewee entirely by email, as she gathered data from several sources within her Web department and sent their answers directly to me.

I used the interviews to further supplement the quantitative statistics. Although they covered the same basic questions as the quantitative statistics, the interviews were more in-depth than the hand-gained statistics. Interviewees were asked why they use the compatibility method(s) they do,

which could not be deciphered by myself through quantitative data analysis alone. *See the appendix for more information as to the specifics of the interviews.*

Additionally, interviews afforded me the opportunity to ask my own how-to questions that I had difficulties discovering answers to on my own. Where my literature review failed in providing me necessary information, my interview subjects succeeded with solid explanations. The primary example that springs to mind is the problem of using JavaScript to detect the client's use of *Flash* on Internet Explorer in a Macintosh environment. I had known that there was a problem with that sort of detection, but it was great to have a face-to-face source explain the actual reason why the problem exists.

Using both qualitative and quantitative methods for this research solidified my work, protecting me from potential biases that I might have encountered had I relied on only one of the two methods for all my data collection. If I were to have focused solely on interviews, then I would risk the possibility that an interviewee may alter facts about his/her company *X* in order to make *X* look good. Because I had a relatively small number of interviews, I might have assumed that most companies like *X* use the same strategy because of its success. This would have been a huge mistake. If I used quantitative data collection to check the interview facts, I could tell if *X* truly is the norm or not.

Similarly, quantitative statistics alone are easy to misinterpret. I might assume that *X* and all its competing companies use a particular strategy in order to save time when in reality the companies may actually be

using the strategy because it reduces work on their server. Using the two methods together considerably strengthens my thesis as they cross-check each other (no pun intended).

The What, not the How

The problem of making Web pages' appearances consistent is not a new one. With time, more and more browser types have become popular and maintaining consistency in presenting information on the Web has become all the more difficult. Designers have a wide variety of options in choosing what methods they use to address this problem. The number of these options also seems to increase as technology grows. As a result of these changes in technology and browser versions, it has become all but impossible to nail down a guideline or set of rules for designing fully compatible Web sites.

This research is not an attempt to create a standardized guideline; rather, my goal is to tie together the multitude of loose ends that have been created by the wide variety of environments in which Web pages can be viewed. The loose ends I refer to include the strategies that Web designers employ. Tying together these loose ends will serve as an initial step toward establishing rough guidelines for cross-compatible Web design. There is but one simple a priori rule that serves as the foundation of my own research: *the more complex one's Web site is, the more difficult it is for one to make the site cross-compatible.*

I realized early on in my research that the academic world has not yet solidly established itself in the field of Web interoperability. Consequently, this paper aims to serve as a pillar of foundation for further research in the area. Hence, I have chosen to sacrifice some detail in favor of a larger

breadth for the scope of my work. An entire paper could be written solely on the use of JavaScript as a solution to interoperability problems. This paper, however, tackles a much wider scope including other solutions in addition to JavaScript.

At the conclusion of each interview I conducted I asked the subject for any comments s/he might have on my research. One of the interviewees said that he was very interested in my topic, but most of the questions I had asked him seemed "a little Internet 101," meaning that the questions I had asked were rather novice. Taken slightly aback, I responded by explaining that if I had gone into each meticulous facet of every type of interoperability solution, I would be writing a one thousand page doctoral dissertation and not a fifty-page masters thesis! The point here is that this paper serves as a foundation from which other academics can delve further into the topics I have exposed. My research is more of a 2001 "State of the Union" address for the Web, generalizing the *what* of interoperability, than it is a detailed manual explaining *how* to make Web content cross-compatible.

The Means

My research has brought forth five general ways to achieve cross-compatibility. They are not mutually exclusive, and in fact they are frequently intertwined with one another to achieve a solution. This section provides a brief overview of each method:

- Star Wars-Safe
- 4.0 Standard
- Hybrid Web design
- Server side detection
- Client side detection
- Cascading Style Sheets

Star Wars-Safe

Creating Web pages using simple HTML makes a site accessible to the largest audience while keeping maintenance undemanding on the content creators. This tactic includes the use of basic, clear-cut HTML that all browsers can understand and avoids newer, complicated HTML that might be browser-specific or unreadable by older browsers. DHTML and CSS, unreadable in several environments, are not included. The use of text-based images is o.k. so long as the corresponding *alt* values adequately substitute for the images in text-only browsers. The same simple HTML works in all environments, regardless of what user-agent the client is viewing from. Uncomplicated JavaScript functions (such as image rollovers) can be included as long as they do not corrupt the page when viewed in older environments. For instance, if a site has a mouse rollover function that

creates a significant graphical change on the screen, the graphical change may not be viewable on some browsers.

The site that I felt best employed this strategy was the official site of the Star Wars movie series, starwars.com. The site is visually attractive, and the same HTML works safely across environments. It uses simple JavaScript rollovers, but the rollovers do not affect the site's rendering in older environments. For the duration of this paper, I will use the term *Star Wars-safe* to reference those simple HTML pages which are safely viewable cross-environment. I inadvertently coined the term as I collected my data—whenever I found a site that effectively utilized simple cross-environment HTML, I noted that the site was "safe, like Star Wars."

For the most part, Star Wars-safe sites are usable on browsers designed for disabled users. However, they are not necessarily strictly compliant with the Americans with Disabilities Act (ADA) Web standard.

4.0 Standard

While Star Wars-safe sites attempt to accommodate the largest user base possible, 4.0 Standard sites concern themselves only with Internet Explorer 4+ and Netscape 4+ image-enabled browser users. Additionally, they cross-check their work only on Macintosh and Windows operating systems. By checking for Windows/Mac/IE4+/NN4+ compatibility, 4.0 Standard sites maintain anywhere from 90 - 99% interoperability with their Web audience. These sites often use DHTML as well as text-based images in their interface, which can leave pages unreadable by Personal Digital

Assistants (PDAs) and older or text-based browsers. Many 4.0 Standard designers, such as Z Promotion and Design, build sites on the premise that "the audience is usually IE and Netscape 4."

Hybrid Web Design

Star Wars-safe and 4.0 Standard sites will generally create one version of each Web page that the sites make available for Web users. Hybrid Web sites, on the other hand, will have two or more versions of documents within the site (Jones 1997, 78). At minimum, these sites have two versions of their home page. For example, a hybrid home page might have one version designated for 4.0 Standard clients and one for lower-level users (3.0 or less browsers). Some hybrid sites host two versions of every page, which makes content editing quite tedious because every edit made to the content must be carried out twice. At most, a hybrid site will have two or more duplications of the *entire* site where each duplication is created for a particular user agent.

The key advantage to hybrid Web strategy is that a designer can feel fairly confident that her work will be cross-compatible. I once employed this tactic at a Web site I used to work for, where over 90% of the user base was made up of Windows/Internet Explorer 5.5 clients. We wanted a DHTML solution to spice up the home page, but did not want to leave the small percentage of lower-level clients with a dysfunctional interface. The DHTML solution we came up with worked only on Internet Explorer in a Windows environment, which meant that any other users would be left with jumbled interfaces. As a result, we decided to create a DHTML home page for

Windows/Internet Explorer users and an alternate DHTML-free home page for all the other users, using JavaScript detection to send them to the alternate page. It served as a good hybrid Web example.

Server Side Detection

If a content provider wishes to utilize hybrid Web design, she must first know what kind of user agent the client is. One way to attain such information about the client is through server side detection. When the user types a URL into his browser, the browser sends an http request to the URL's host server. Within that request exists information about the client, namely what kind of operating system, browser and version the client is running. The server can then reply with a document compatible for that type of user agent, provided that the content producers have made the site's content compatible for that client type.

There are several ways to perform server side detection, and I will not attempt to be at all comprehensive in describing them. To keep it simple, the server can be programmed to complete the duty in a wide variety of computer languages, depending on which ones the server supports. Many sites possess more than one page that needs detection (for instance, the personal example I mentioned earlier needed detection on only one page, for the rest of the site was 4.0 Standard). If the whole site is in fact hybrid, then the site's developer has four options in remembering the type of user agent (also known as maintaining state):

1) To maintain state, she can choose to repeatedly detect the user agent every time the user requests a page.
2) She can maintain state by sending out a cookie to the browser in the reply. In this case, the server reads the cookie, as opposed to the userAgent, portion of the reply to decipher the client.
3) She can include *hidden* attributes in a form in the reply's HTML thus designating the user agent. This tactic would be used effectively in content dominated by forms.
4) Probably least effective, the developer can customize all the page's hyperlinks to have Common Gateway Interface (CGI) methods included in their URLs. In most cases, this fourth option is unnecessarily complex. To no surprise, I did not find one site that employed CGI to maintain state of the user agent.

Client Side Detection

The other way of detecting what kind of environment from which the user is viewing the Web site occurs on the client side. Whereas several different mechanisms are used for server side detection, by far the most popular means of client side detection is performed through JavaScript. Most browsers understand JavaScript, making its use very reliable.

There are two major ways of using client side detection to achieve compatibility. The first is done entirely on the client side, with no help from the server. The client requests a document, and the server returns a document that is pre-armed for multiple environments. For this example, let us assume that the developer is designing for 4.0 Standard compatibility and uses JavaScript as the ammunition. She wants to have layered DHTML that works in both Internet Explorer and Netscape Navigator. This is a compatibility problem because Netscape, Netscape 6, and Internet Explorer 4+ understand elements within a document differently. As a solution, she

uses JavaScript to detect the browser and version. She employs different JavaScript commands when accessing the elements according to the browser type. If the client is Internet Explorer, the JavaScript executes a command using *document.all* to reference an element. If the client is Netscape Navigator 4+, the JavaScript command includes *document.layers* to reference the element. The DTHML document's elements are then accessible in both Explorer and Netscape. Many other browsers, however, are not compatible with this solution.

Should the designer want the content to be accessible to other types of user agents (such as Lynx users) without the jumbled DHTML meddling with the interface, she can additionally use a second method of client side detection, combining JavaScript with hybrid Web design. This strategy requires two versions of the page: one, an enhanced version of the JavaScript-armored DHTML page described above; two, a Star Wars-safe version of the same content. Upon the request, the client receives the enhanced DHTML page. The enhancement uses a new JavaScript function to detect the user agent before the page has fully loaded. If the browser is *not* Netscape 6, Internet Explorer 4+, or Netscape Navigator 4+ then the user is transported to a Star Wars-safe version of the same page. Otherwise, the user remains on the DHTML page. This solution provides good interoperability, but requires significantly more content management to maintain hybrid Web content.

Cascading Style Sheets

The last general category of achieving compatibility regards the use of Cascading Style Sheets. Sites of this type are Star Wars-safe for the most part, but additionally include style sheets to provide consistency across environments. They avoid JavaScript and any other technologies that may produce cross-compatibility problems. The key advantage of designing CSS content is that even if a browser does not support CSS, the page will usually render much more nicely than a DHTML page in a pre-DHTML browser. CSS pages are not as widely compatible as Star Wars-safe pages, but they service a much larger audience without the significant problems that might occur with the use of DHTML in older browsers.

The methods described above are intentionally generalized. There are countless other smaller-scaled tactics that can be employed to make a site compatible. For instance, when creating a colored table with text inside it, a designer should probably not keep the table's background a dark color and the text within the table a light color (assuming that the body of the Web page has a light background color). Should a client who uses an older browser that does not render table backgrounds visit the page, then the text will be difficult to see, as it will blend in with the body's color. Analysis of strategies like these can be found around the Web at your own peril; however, such details are beyond the scope of this paper.

Relevant Statistics

Quantitative data collection revealed several present trends in Web design. To salvage some sort of explanation for these trends, I used the knowledge I attained through my subject reading as well as the interviews I conducted. Serving as the culmination of my research, this section examines ten general issues and design decisions regarding interoperability on the Web:

The Star Wars-safe strategy
Flash
Client side detection
CSS
Hybrid Web design
Server side detection
The Macintosh platform
3.0 and lower level browsers
Monitor size
Text only browsers

The Force is Strong in this One

Over one quarter of the sites I examined (27%) consisted of simple HTML that worked satisfactorily across environments, fulfilling the Star Wars-safe requirement. Not surprisingly, these sites tended to have large user bases with consistently large hit counts. Having a Star Wars-safe site for large Web presences like yahoo.com and hotmail.com is beneficial in several ways. First, the simplicity of their sites makes them accessible to virtually all user agents. Second, they have a reduced load on their servers because the content they send to their clients is minimal. They include less images and JavaScript, having text instead. This lowers the kilobytes of information that

pass through the server during each response. Furthermore, I assume that they do not employ server side detection because that too would further strain their already-busy machines.

I presume that most of the other Star Wars-safe sites with smaller user bases elected to use this method in order to keep their site design and maintenance simple, avoiding the difficulties associated with more complicated compatibility options. One of my interview subjects manages a site that gets around a half million hits a week. In explaining his rational for maintaining only one site for all his clients, he exclaimed, “I’ve only got four people to work with!” He simply did not have the manpower to design a hybrid site that could send customized pages to different users. He felt the Star Wars-safe solution was his best option.

His site’s implementation is a loose rendition of the Star Wars-safe description in the sense that while it is usable in virtually all environments, it caters toward 4.0 and above Internet Explorer and Netscape users running Windows. He continued to explain his rational by citing his site’s WebTrend report, which provides data about the site’s visiting clients (attained through the server logs). Over 80% of his clients were viewing from the same browser and platform—Internet Explorer on Windows—and over 95% of users were on a minimum version of 4.0 on Internet Explorer or Netscape on Windows. Having such a distinguishable user base allowed him to make the site optimal for these users. At the same time, the site makes sure that no content is too complex for weaker browsers by excluding style sheets and DHTML. Lower-level clients might have an inferior rendering of the site’s pages, but

none of the content will be lost or confusing due to dysfunctional style sheets or DHTML.

When I asked him about the disregard for the "other" users, as minimal as it was, he defended himself by stating that his site is a *marketing* site, and that the types of clients the site is marketing to are expected to have good browsers.

Who can You Flash?!

Another marketing site, nsync.com, demonstrates the implementation of a similar idea. Most of the NSync site is entirely Flash-based, leaving non-Flash users without content. Perhaps the exclusion of weaker browsers can be used as an elitist strategy. If a lower-level client without the Flash plugin visits nsync.com, the user may understand a hidden message not entirely unlike this: *Flash is hip and so is NSync. If you are hip, then you will have Flash and you will be cool enough to listen to NSync.*

Of all the sites I viewed, I deemed 27% of them as *marketing* sites and found that over half of all the marketing sites employed a significant amount of Flash in their content (the Flash sites made up 15% of the total number of sites I researched and every Flash site, not coincidentally, was a marketing site). I define *marketing sites* as sites that serve as marketing tools much more than as typical information services (e.g. search portals). Only half of the Flash sites provided an alternative for non-Flash users. More often than not, that alternative to using Flash came in the form of a page consisting solely of

a hyperlink from which to download Flash, clearly eliminating a large number of users from being able to view content!

The senior developer at eluxury.com, a site that relies heavily on Flash, clarified that his clients were “a higher level audience” and that his users mostly have newer computers that come equipped with Flash. Nevertheless, eluxury uses a combination of client and server side detection to maintain usability for non-Flash users. Every time a user requests a page, the response includes a JavaScript function that detects the presence of the Flash plugin. If the plugin exists, then a cookie is set on the client. The next time the client makes a request, the server examines the cookie to determine whether or not to include Flash in the next response. The response is generated on the fly so that the server decides what to incorporate into the database-driven HTML. If the cookie says that the user has the Flash plugin, then the server includes Flash in the response. Otherwise, the server includes additional HTML content to replace the Flash segments that would not be compatible with a non-Flash client. This is an example of a logical solution that works fairly well but requires a significant amount of programming on the back end. Many Web sites, such as the marketing site mentioned earlier, cannot afford to include such a solution. Other sites may choose not to implement this solution because it would only service a small market share.

There is a significant problem in Flash detection that deserves mention here. According to many of the developers I interviewed, there is not an efficient means of detecting Flash on Internet Explorer in a Macintosh environment. Several Web sites attempt to use JavaScript to write code in

Visual Basic (VB is a Microsoft language) that detects Flash. I will not dive into the technicalities behind this JavaScript/VB combination, but I noted two sites that attempted to utilize the two languages in order to detecting Flash. I showed their code to one of my subjects and he was certain that they would not successfully detect Flash in *all* environments. At the time of this writing, eluxury was attempting to write such a code that would successfully detect Flash in all Mac/Internet Explorer environments.

Other Client Side Detection

More than half of the Flash sites I studied used JavaScript to detect the plugin, making up 8% of *all* the sites I reviewed. One quarter of the total number of sites used some sort of client side detection to make their pages more compatible. There are too many combinations of patterns to discuss here, but the most popular was a simple script that detected if the client was running Netscape Navigator or Internet Explorer. Only one site went so far as to detect Opera and WebTV. Usually the IE/Netscape code was used to make DHTML interoperable. Almost every case of detection would serve the purpose of properly accessing elements (see the *elements* discussion earlier), as Internet Explorer and Netscape have slight differences in how they position CSS elements within the browser window.

Not Everyone has Style

A surprisingly low 27% of the sites I reviewed utilized style sheets. According to Sparklejet, a design company that relies heavily on their use,

style sheets can reach almost all of your Web audience when used properly. By *properly*, he alludes to the use of cross-browser CSS only, and straying from the temptation of applying cool effects that are browser-specific in nature. He cited that the five following environments make up 99% of most Web audiences:

Windows	**Macintosh**	**Unix**
IE 4	IE 5	NN 4
IE 5		
NN 4		

The proper use of style sheets will render compatible pages in all these environments, which seems to be a successful solution.

The one problem that I found with regard to cross-compatible style sheets regards using the *text-decoration* attribute for hyperlinks when the user's mouse hovers above the link. If hyperlinks are styled to have no underline in their normal state, but underlines when they are hovered, inconsistencies occur between the two major browsers. Internet Explorer renders the code normally. Netscape, on the other hand, underlines the text regardless of whether or not the user hovers the mouse over it. Nearly half of the CSS sites used the underline-only-on-hover mechanism for hyperlinks. The acceptance of this particular error in consistency has become something of an industry standard.

Static Hybrid Architecture

Considering the difficulties associated with maintaining multiple forms of the same content, it comes at no surprise that very few of the sites I

visited maintained some form of a hybrid structure—a mere 13%. Most of these sites had only one hybrid page. None of the designers I spoke with felt that hybrid design was a logical solution for compatibility. Usually, sites that utilized hybrid design were Flash sites. The rock band, Megadeth, has a flash site that is completely hybrid, providing content for both Flash and non-Flash viewers. The probable reason that the site is able to use this sort of design is because the content changes are few and relatively simple to update. Maintenance probably requires a very minimal number of staff members. Obviously, the smaller a site's size, the easier it is to create hybrid content.

Server Side Activity

With the exception of small/medium-sized Web presences such as megadeth.com, it appears as though hybrid design is only efficient if the multiple content is generated on-the-fly from a single content creation mechanism. Small sites do not have large enough user bases to consider employing such a mechanism. Large-scale sites, however, can use it, provided they have the manpower. "Database-driven pages are an intelligent solution," stated a representative from the Fluid design company, "but they result in fairly large time costs on the development-implementation side of things." The eluxury example I cited earlier is a good example of on-the-fly content generation.

Server side detection and generation will become more prevalent in the near future. WebslingerZ, a design group, strongly advocates the use of on-the-fly generation for their larger customers. They estimate that 75% of

their new development is done with Cocoon, a server side Java/XML (Extensible Markup Language) application. Cocoon facilitates content maintenance, as each Web page on a site requires only a single XML file. From the XML file, Cocoon parses together a Web page that is customized for each individual client, depending on the client's environment (Apache 2001).

The growing popularity of remote Internet connections—particularly with the rise of the Personal Digital Assistant (PDA)—will inevitably make server side detection and page generation a more attractive option for Web sites in the next few years. Additionally, devices for disabled Web surfers are becoming more popular. The disabled are a somewhat untapped reservoir of potential clients on the Internet, and I am confident that many more commercial sites will soon use server technology to appropriate compatible Web content for these users. Furthermore, client side detection is not usable in many lower-level browsers, so reliable detection can only take place on the server side. An efficient solution for PDA and disabled users is achievable only through server side detection and database-driven content generation.

S.--'O.S.'!

Only one of the sites I reviewed was not usable in a Macintosh environment, suggesting a solid awareness of Macintosh-related compatibility problems. Either designers stray from using code that is problematic on the Macintosh or they take precautions to ensure good usability on the platform. On occasion, designers may have reason to ignore the Macintosh platform, depending upon who their user base is. For

example, ea.com (EA Sports) recently released a new site that offers video games for Windows users. The games do not work on Macintosh, so the site sends Macintosh users to a page explaining that the site caters to Windows users. A developer at EA informed me that they are presently converting their games to Macintosh format and that they will soon be making the site Macintosh-compatible as well.

Where the Wild Things Are

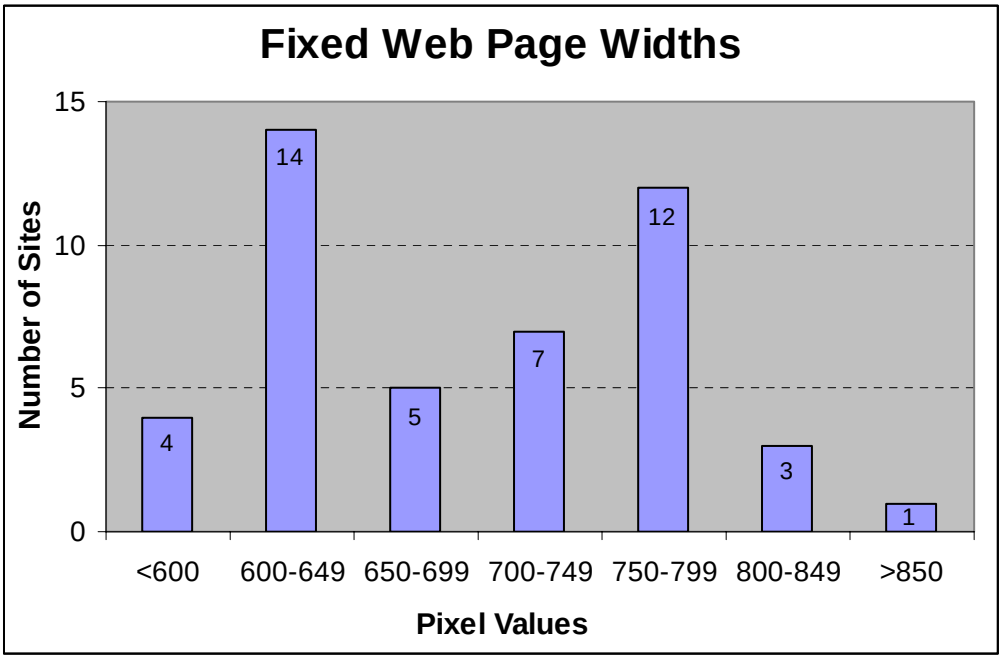
Many of the sites I examined broke down in lower level environments. 17% of all the sites were seriously dysfunctional on 3.0 browsers. Usually these pages had JavaScript error messages upon loading. Most other sites tended to have minor problems, such as jagged table borders or text that leaked outside of their intended realm. One subject I interviewed explained that "people viewing through older browsers are used to seeing messed up pages, so [having poor lower level renders] is o.k."

Monitor Solutions

It is interesting to first note that none of the people I interviewed discussed solutions to problems with monitor resolution. They seem to take the same reasoning on this issue as they do with regard to older browser users: *people who have poor resolution are used to seeing pages with strange color rendering.*

There are, however, two general ways to address the wide variety of monitor size that users have. First, designers can opt to create Web pages

that are small enough to fit into most windows without requiring the user to scroll from side to side to view all the page's content. 61% of the sites I viewed had set table widths. The width sizes on these sites varied from 468 pixels to 860 pixels. The median width was 651 pixels and the average was 675 pixels. The designers I spoke to agreed that the industry standard on what size to use was constantly growing. The past couple years a safe mark would be sizing a page at around 600 pixels. Now, as users tend to have larger monitors, designers have chosen to increase that number. Most of the designers I spoke with now design for a minimum width of 700 pixels. The developers at Tatu now design for 700 pixel wide screens. HyperArts used to design for a 640 pixel width, but have also graduated to 700 pixels. The following is a histogram depicting the various sizes I found in my data collection:



The second means of attaining screen size interoperability is a little more complex, requiring more intricate compatibility testing during development: designers can employ a percentage width for their sites. Through this technique, the tables on the site adjust in size according to the client's available screen size. One great advantage of Flash is that it can be rendered according to percentage widths so that a Flash presentation can occupy the client's entire window. Including these Flash-designed sites, 38% of all the sites I examined utilized percentage widths.

Two of the Web sites I reviewed had significant problems on a 780 pixel wide screen, where content was not viewable and scrolling had been disallowed by the sites.

Font Size Problems

Sites are often troubled by browsers' options of increasing the user's font size. Most of the problems occur when the user elects to increase Netscape Navigator's font size two or more times above the default size. Although increased font size can lead to knotty renderings on Internet Explorer as well, it is less of a problem than on Netscape. 23% of the total sites faced interface problems when font size was increased. 10% of these problems were major, leading to significant site disruption and some loss of usability. Usually the major problems occurred on Netscape.

Many designers disregard the font size problem. Sparklejet noted that there are many users out there who do not understand browser logistics well—the company assumes that users will not change their settings. Tatu

often designs with a larger font size ahead of time to insure that the site will not break should the user select larger font sizes. HyperArts specifies *font* tags at "-1" in order to safely keep the sizes small so that tables will not break. Of all the questions I asked the interviewees, questions about font size problems were the most difficult for them to answer. A designer at WebslingerZ said that the company usually "bites its tongue" and lets the client have the option of using larger font sizes. The font size issue is truly frustrating because it can turn an attractive Web site into a mush of chaos with the click of a button.

Lynx to the Past

For the most part, the Internet giants with large user bases (such as Yahoo) maintain sites that are very accessible to text-only browsers. However, nearly one out of every three (29%) of all the sites I viewed was useless when viewed through a text-only browser. Many more of the sites were very difficult to navigate in a text-only environment. Design standards seem to allow for the neglection of such browsers on small- to medium-sized sites because they make such a small percentage of the user agents on the Web.

Cross-Examination

It appears as though most designers have a solid knowledge of basic compatibility principles. At the very least, they know to cross check their work on the latest Internet Explorer and Netscape Navigator versions. All of the sites I visited worked in the two major environments, Windows Internet Explorer 5 and Netscape Navigator 4.7. How do designers decide which environments to their Web pages should work in? This question is usually answered with simple math. The key part of the equation is figuring out who the site's audience is.

Who is Your Audience?

There are a couple conclusions that I reached with regard to how designers choose which environments to support. Large-scale sites tend to cater to as wide a user base as possible and usually employ Star Wars-safe tactics. Suppose that yahoo.com found that 99% of its users had 4.0 browsers and thus decided to employ DHTML throughout their Web site. This would leave 1% of the users without accessible content. This would be a terrible business decision because 1% of yahoo.com's user base is a *tremendous* number of people. To ignore such a large number of users would surely result in a significant decrease in hits and probably a drop in income.

Smaller sites can usually get away with designing for 4.0 Standard compatibility because ignoring 1% of their users will probably not significantly affect their financial situation. It is often more important for

these sites to have a more attractive interface than it is to cater to the lower-level users. Whatever size the site, a cost-benefit analysis should be performed in order to decide what is best for the site. WebslingerZ noted that target audiences vary greatly and that significant effort must be made to find out who that audience is before making design decisions. Server logs should be analyzed periodically to maintain an understanding of the audience's diversity.

On occasion, a company might be fortunate enough to have a specific user base. The developer from Z Production and Design pointed out a possible scenario: if you are designing for a venture capitalist, then you can probably feel safe about creating a 4.0 Standard site. Obviously a venture capitalist probably would opt not to invest in a company who is still running Mosaic to view the Web!

Cross-Check

Once a developer knows who to cater to, the developer must figure out a means of checking compatibility across environments. Sfgate.com is a large site that chooses to create pages accessible to as many users as possible. The site uses simple HTML with style sheets. Sfgate employs a three person Quality Assurance Group to cross-check all content across several environments. A source at sfgate explained that viewing content through Opera on Linux is a good primary option in checking content for errors. She commented that the Opera/Linux combination was great for picking pages apart because "everything breaks there!"

Sfgate's Quality Assurance Group was the most elaborate cross-checking scheme I encountered in my interviews. Most of the design companies I interviewed checked their work for 4.0 Standard compatibility, making sure to have both Macintosh and Windows machines available for cross-checks. The larger Web sites tended to pay more attention to smaller monitor sizes and lower-level users. None of the designers had started to consider 6.0 browsers in their designs. Netscape 6 is still a relatively new browser and Internet Explorer 6 is still in beta at the time of this writing.

Conclusion

All the designers I spoke with agreed that in the past year or two, the creation of interoperable Web content has become much easier as more and more users have 4.0 or greater browsers. "It's getting better all the time!" exclaimed one subject when I asked him about possible trends in compatibility. Today, most Web sites can get away with providing 4.0 Standard sites. In mid-April of 2001, just prior to the completion of this paper, the Star Wars designers rebuilt their site. Starwars.com is now a 4.0 Standard site that breaks significantly on older browsers (no longer is it Star Wars-safe!). This illustrates how quickly compatibility standards evolve on the Internet. Presently, a vast amount of larger sites opt for the 4.0 Standard.

However, I think that two factors will play large roles in determining future compatibility standards: Flash and PDAs. Should Flash continue to increase in popularity, compatibility will become less of an issue. With a Flash site, the prime concern regards whether or not the user has the plugin, and without that problem, many designers would have a much easier time worrying about interoperability.

On the other hand, if PDAs continue to skyrocket in popularity, compatibility will become a huge issue once again, because PDA content is rendered so differently than content in typical browsers. PDAs could potentially send compatibility back into the dark ages as it was a few years ago.

Another noteworthy factor is the increase in disabled users on the Internet. Technological innovations have made the Internet much more accessible to them. Because their browsers render pages more like text-only browsers and PDAs, they cannot access a large amount of Web content. Inevitably, larger Web sites will soon need to provide a resolution to the problem of accessibility for both disabled and PDA users.

Throughout this paper I have advocated the use of server side detection to generate customized database-driven content, if a Web presence can afford it. Hugh Cayless, a lead developer for the University of North Carolina-Chapel Hill, is one of many to claim that XML is presently the best option—and will continue on to be the future standard—for generating HTML content from the server on the fly, appropriately customized for each user. As discussed earlier, a single XML file can be parsed into different HTML formats, using style sheets (XSL) to customize a Web page's interface for each individual client. XML keeps maintenance simple, requiring only one XML file for each page of content. A site using XML need only create one XSL file for each type of user environment. This file can be used for every page on the site to appropriately organize content for each user. Content maintenance is kept to a minimum and more importantly, template changes to the Web site must only occur once for each XSL file. Constantly growing in popularity, XML is a first-rate solution that all large Web presences should consider implementing to achieve maximum interoperability.

At any rate, there is a fact which all designers, developers, and managers seem to agree upon: knowing who visits a site is the most crucial step in deciding how cross-compatible a site should be. Once a manager can identify her audience, she can start making decisions about what environments her Web site should cater to.

Appendix

The following pages include AA-IRB consent and approval information.

IRB Form

1. Project Description. What strategies and reasoning do Web designers implement in order to compensate for the countless combinations of computer platforms, browsers, and monitors that the world utilizes? My research aims to answer this question regarding Web design interoperability issues.

My data collection will come from two sources: qualitative data collection through interviews as well as quantitative data obtained through Web site analysis. The interview questions follow:

1. What measure(s) do you take, if any, to address interoperability problems, namely differences in browsers, operating systems, and monitor sizes/resolutions? *Style sheets, browser detecting, database-driven pages, etc.*
   (if NONE, go to #8. Else continue)
2. Why did you choose this method over others?
3. Where did you get the ideas for this method?
4. Do you test your Web pages in more than one environment? If so, how?
5. What browsers do you focus on?
6. What considerations do you make for operating systems and monitors?
7. Do you consider problems with text size? (GO TO #9)
8. Why don't you address compatibility problems?
9. Who is your site's audience?

2. Participants. I will recruit approximately 20 Web designers and/or managers to be interviewed. The only inclusion criteria is (1) that they speak English and (2) have been involved with the decision-making surrounding browser/platform/monitor compatibility for a non-personal Web site (*browser/platform/monitor compatibility* simply means making Web pages work in all types of browsers, on different kinds of computers, and with different sized monitors).

Limited travel funds restrict me from interviewing anyone face-to-face outside of the San Francisco Bay Area and Raleigh/Durham Area. Therefore, I will choose the subjects based on convenience and geography. I will perform two Internet searches to find interviewees: (1) an Internet search for larger companies in the San Francisco and Raleigh/Durham Areas (2) a similar Internet search to find Web design companies within the two areas. I will then contact the various companies in search of interviewees who fit the inclusion criteria.

3. Risk. The participants are not at risk.

5. Illegal Activities. There are no illegal activities involved.

6. Deception. There is no deception involved.

8. Prior Consent. Prior consent will be attained with the attached consent form.

9. Privacy. Privacy and confidentiality will be maintained as the interviewee pleases. The consent form includes an option for the interviewee to check a box allowing me to cite his/her company name in my work. This checkbox is *optional*.

ACADEMIC AFFAIRS INSTITUTIONAL REVIEW BOARD
Request for Review of Research Involving Human Participants
COVER SHEET

1. AA-IRB Request Number 00-045(LIBS) Date Submitted 12/6/00

2. This project relates to or supersedes previous Request Number _____, approved on _____.

3. Principal Investigator (PI):
Name Harry Ahlas
Address 616 Craige
Phone/E-mail 914-3983 hahlas@email.unl.edu

Faculty Advisor (FA), if PI is student:
Name Gary Marchionini
Address 203 Manning Hall
Phone/E-mail (919) 966-3611 marchionini@ils.u

4. Project title: Browsers, Platforms, and Monitor Compatibility Issues with Regard to Web De

5. Project types. Check all that apply: ☑ New ☐ Renewal ☐ Protocol Change ☑ Specific Project

☐ This project involves a school system; system approval: ☐ is attached; ☐ is pending.

☐ Grant Proposal; campus routing: ☐ is complete; ☐ is ongoing; ☐ has yet to begin.
List funding agency:

6. PI/FA recommendatio. If PI is a student, FA must also sign this form:

☒ Exempt from further IRB review; Exempt Paragraph Number 2, AA-IRB Manual Section IX.A.2.
☐ Expedited review; Expedited Paragraph Number _____, AA-IRB Manual Section IX.B.2.
☐ Full review
If DHHS-funded and involving offsite agency or location(s), Single Project Agreements must be filed. Refer to AA-IRB Manual Section IX.E.

12/6/00
Signed and Dated by Principal Investigator

12/6/0
Signed and Dated by Faculty Advisor

7. Local review committee recommendation. Attach Working Forms.

☒ Exempt from further IRB review; Exempt Paragraph Number 2, AA-IRB Manual Section IX.A.2.
☐ Expedited review; Expedited Paragraph Number _____, AA-IRB Manual Section IX.B.2.
☐ Full review

Barbara M. Wilde
Signed and Dated by Chair, Local Review Committee

8. Departmental endorsement:

Marshall 12/8/00
Signed and Dated by Department Chair/Dean/Director

9. AA-IRB recommendation:

☑ Exempt from further IRB review; Exempt Paragraph Number 2, AA-IRB Manual Section IX.A.2.
☐ Expedited review; Expedited Paragraph Number _____, AA-IRB Manual Section IX.B.2.
☐ Full review

10. AA-IRB decision:

☑ Exempted; no further review needed unless protocol changes.
☐ Approved as Specific Project.
☐ Approved as Grant Proposal. Specific Project approval needed prior to data collection.
☐ Approved with special conditions, see attachment.
☐ Not approved.

David A Eckerman 12/13/00
Signed and Dated by Chair, AA-IRB

AA-IRB approval of this project expires _____

School of
Information and Library Science
The University of North Carolina at Chapel Hill

Consent form–*Browser, Platform, and Monitor Compatibility Issues with Regard to Web Design*

To Participant:

I am writing a research paper about interoperability—how people make Web pages look similar on different types of browsers (like Microsoft Internet Explorer and Netscape Navigator) as well as different computers and monitors. For my work, I need to interview 20 developers and managers like you to understand how and why you make your decisions regarding interoperability.

Before I can ask you any questions, I need you to sign this form which gives me permission to interview you and cite your words in my research. The interview should last approximately fifteen minutes; you may withdraw from the study at any time. I will not use your name in my paper, and my research poses no risk to you in any way at all.

If you have any questions regarding my work, you can contact me or my advisor, Dr. Gary Marchionini:

Harry Ahlas
616 Craige, UNC Residence Hall
Chapel Hill, NC 27514
(919) 914-3983
hahlas@email.unc.edu

Dr. Gary Marchionini
(919) 966-3611
marchionini@ils.unc.edu

If you have any questions about your rights as a participant, you can contact the Academic Affairs Institutional Review Board:

Academic Affairs Institutional Review Board
David A. Eckerman, Chair
CB #4100, 201 Bynum Hall
The University of North Carolina at Chapel Hill
Chapel Hill, NC 27599-4100
(919) 962-7761 aa-irb@unc.edu

[ ] Check this box if you do NOT want me to use your company's name, or any other names you mention during the interview, in my work.

[ ] Check this box to give me permission to use your company's name in my work. *You do NOT have to check this box if you do not want to.*

I give Harry Ahlas permission to conduct an interview with me and print what I say in my interview into his paper on Web interoperability.

_____
Sign Here

12/13/00

_____
Date

School of
Information and Library Science
The University of North Carolina at Chapel Hill

Consent form–*Browser, Platform, and Monitor Compatibility Issues with Regard to Web Design*

To Participant:

I am writing a research paper about interoperability—how people make Web pages look similar on different types of browsers (like Microsoft Internet Explorer and Netscape Navigator) as well as different computers and monitors. For my work, I need to interview 20 developers and managers like you to understand how and why you make your decisions regarding interoperability.

Before I can ask you any questions, I need you to sign this form which gives me permission to interview you and cite your words in my research. The interview should last approximately fifteen minutes; you may withdraw from the study at any time. **I will not use your name in my paper**, and my research poses no risk to you in any way at all.

If you have any questions regarding my work, you can contact me or my advisor, Dr. Gary Marchionini:

Harry Ahlas
616 Craige, UNC Residence Hall
Chapel Hill, NC 27514
(919) 914-3983
hahlas@email.unc.edu

Dr. Gary Marchionini
(919) 966-3611
marchionini@ils.unc.edu

If you have any questions about your rights as a participant, you can contact the Academic Affairs Institutional Review Board:

Academic Affairs Institutional Review Board
David A. Eckerman, Chair
CB #4100, 201 Bynum Hall
The University of North Carolina at Chapel Hill
Chapel Hill, NC 27599-4100
(919) 962-7761 aa-irb@unc.edu

[ ] Check this box if you do NOT want me to use your company's name, or any other names you mention during the interview, in my work.

[✓] Check this box to give me permission to use your company's name in my work. *You do NOT have to check this box if you do not want to.*

I give Harry Ahlas permission to conduct an interview with me and print what I say in my interview into his paper on Web interoperability.

Hugh A. Coyle
Sign Here

4/20/01
Date

School of
Information and Library Science
The University of North Carolina at Chapel Hill

Consent form–*Browser, Platform, and Monitor Compatibility Issues with Regard to Web Design*

To Participant: (Giles Hendrix - Please send signed fax back to 919-967-9243

I am writing a research paper about interoperability—how people make Web pages look similar on different types of browsers (like Microsoft Internet Explorer and Netscape Navigator) as well as different computers and monitors. For my work, I need to interview 20 developers and managers like you to understand how and why you make your decisions regarding interoperability.

Before I can ask you any questions, I need you to sign this form which gives me permission to interview you and cite your words in my research. The interview should last approximately fifteen minutes; you may withdraw from the study at any time. I will not use your name in my paper, and my research poses no risk to you in any way at all.

If you have any questions regarding my work, you can contact me or my advisor, Dr. Gary Marchionini:

Harry Ahlas
616 Craige, UNC Residence Hall
Chapel Hill, NC 27514
(919) 914-3983
hahlas@email.unc.edu

Dr. Gary Marchionini
(919) 966-3611
marchionini@ils.unc.edu

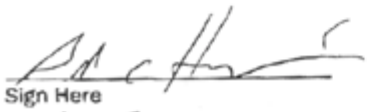
If you have any questions about your rights as a participant, you can contact the Academic Affairs Institutional Review Board:

Academic Affairs Institutional Review Board
David A. Eckerman, Chair
CB #4100, 201 Bynum Hall
The University of North Carolina at Chapel Hill
Chapel Hill, NC 27599-4100
(919) 962-7761 aa-irb@unc.edu

[ ] Check this box if you do NOT want me to use your company's name, or any other names you mention during the interview, in my work.

[X] Check this box to give me permission to use your company's name in my work. *You do NOT have to check this box if you do not want to.*

I give Harry Ahlas permission to conduct an interview with me and print what I say in my interview into his paper on Web interoperability.



Sign Here

4. 17. 2001

Date

School of
Information and Library Science
The University of North Carolina at Chapel Hill

Consent form–Browser, Platform, and Monitor Compatibility Issues with Regard to Web Design

To Participant: MATT JALBERT

I am writing a research paper about interoperability—how people make Web pages look similar on different types of browsers (like Microsoft Internet Explorer and Netscape Navigator) as well as different computers and monitors. For my work, I need to interview 20 developers and managers like you to understand how and why you make your decisions regarding interoperability.

Before I can ask you any questions, I need you to sign this form which gives me permission to interview you and cite your words in my research. The interview should last approximately fifteen minutes; you may withdraw from the study at any time. I will not use your name in my paper, and my research poses no risk to you in any way at all.

If you have any questions regarding my work, you can contact me or my advisor, Dr. Gary Marchionini:

Harry Ahlas
616 Craige, UNC Residence Hall
Chapel Hill, NC 27514
(919) 914-3983
hahlas@email.unc.edu

Dr. Gary Marchionini
(919) 966-3611
marchionini@ils.unc.edu

If you have any questions about your rights as a participant, you can contact the Academic Affairs Institutional Review Board:

Academic Affairs Institutional Review Board
David A. Eckerman, Chair
CB #4100, 201 Bynum Hall
The University of North Carolina at Chapel Hill
Chapel Hill, NC 27599-4100
(919) 962-7761 aa-irb@unc.edu

[ ] Check this box if you do NOT want me to use your company's name, or any other names you mention during the interview, in my work.

[X] Check this box to give me permission to use your company's name in my work. *You do NOT have to check this box if you do not want to.*

I give Harry Ahlas permission to conduct an interview with me and print what I say in my interview into his paper on Web interoperability.

Sign Here

5 March 2001
Date

School of
Information and Library Science
The University of North Carolina at Chapel Hill

Consent form–*Browser, Platform, and Monitor Compatibility Issues with Regard to Web Design*

To Participant:

I am writing a research paper about interoperability—how people make Web pages look similar on different types of browsers (like Microsoft Internet Explorer and Netscape Navigator) as well as different computers and monitors. For my work, I need to interview 20 developers and managers like you to understand how and why you make your decisions regarding interoperability.

Before I can ask you any questions, I need you to sign this form which gives me permission to interview you and cite your words in my research. The interview should last approximately fifteen minutes; you may withdraw from the study at any time. **I will not use your name in my paper**, and my research poses no risk to you in any way at all.

If you have any questions regarding my work, you can contact me or my advisor, Dr. Gary Marchionini:

Harry Ahlas
616 Craige, UNC Residence Hall
Chapel Hill, NC 27514
(919) 914-3983
hahlas@email.unc.edu

Dr. Gary Marchionini
(919) 966-3611
marchionini@ils.unc.edu

If you have any questions about your rights as a participant, you can contact the Academic Affairs Institutional Review Board:

Academic Affairs Institutional Review Board
David A. Eckerman, Chair
CB #4100, 201 Bynum Hall
The University of North Carolina at Chapel Hill
Chapel Hill, NC 27599-4100
(919) 962-7761 aa-irb@unc.edu

[ ] Check this box if you do NOT want me to use your company's name, or any other names you mention during the interview, in my work.

[✓] Check this box to give me permission to use your company's name in my work. *You do NOT have to check this box if you do not want to.*

I give Harry Ahlas permission to conduct an interview with me and print what I say in my interview into his paper on Web interoperability.

Sign Here

2/7/01
Date

School of
Information and Library Science
The University of North Carolina at Chapel Hill

Consent form–*Browser, Platform, and Monitor Compatibility Issues with Regard to Web Design*

To Participant:

I am writing a research paper about interoperability–how people make Web pages look similar on different types of browsers (like Microsoft Internet Explorer and Netscape Navigator) as well as different computers and monitors. For my work, I need to interview 20 developers and managers like you to understand how and why you make your decisions regarding interoperability.

Before I can ask you any questions, I need you to sign this form which gives me permission to interview you and cite your words in my research. The interview should last approximately fifteen minutes; you may withdraw from the study at any time. I will not use your name in my paper, and my research poses no risk to you in any way at all.

If you have any questions regarding my work, you can contact me or my advisor, Dr. Gary Marchionini:

Harry Ahlas
616 Craige, UNC Residence Hall
Chapel Hill, NC 27514
(919) 914-3983
hahlas@email.unc.edu

Dr. Gary Marchionini
(919) 966-3611
marchionini@ils.unc.edu

If you have any questions about your rights as a participant, you can contact the Academic Affairs Institutional Review Board:

Academic Affairs Institutional Review Board
David A. Eckerman, Chair
CB #4100, 201 Bynum Hall
The University of North Carolina at Chapel Hill
Chapel Hill, NC 27599-4100
(919) 962-7761 aa-irb@unc.edu

[ ] Check this box if you do NOT want me to use your company's name, or any other names you mention during the interview, in my work.

[✓] Check this box to give me permission to use your company's name in my work. *You do NOT have to check this box if you do not want to.*

I give Harry Ahlas permission to conduct an interview with me and print what I say in my interview into his paper on Web interoperability.

Mimi Chon, Fluid, Inc.
Sign Here
2/11/01. 415.369.3200.
Date

School of
Information and Library Science
The University of North Carolina at Chapel Hill

Consent form–*Browser, Platform, and Monitor Compatibility Issues with Regard to Web Design*

To Participant:

I am writing a research paper about interoperability—how people make Web pages look similar on different types of browsers (like Microsoft Internet Explorer and Netscape Navigator) as well as different computers and monitors. For my work, I need to interview 20 developers and managers like you to understand how and why you make your decisions regarding interoperability.

Before I can ask you any questions, I need you to sign this form which gives me permission to interview you and cite your words in my research. The interview should last approximately fifteen minutes; you may withdraw from the study at any time. **I will not use your name in my paper**, and my research poses no risk to you in any way at all.

If you have any questions regarding my work, you can contact me or my advisor, Dr. Gary Marchionini:

Harry Ahlas
616 Craige, UNC Residence Hall
Chapel Hill, NC 27514
(919) 914-3983
hahlas@email.unc.edu

Dr. Gary Marchionini
(919) 966-3611
marchionini@ils.unc.edu

If you have any questions about your rights as a participant, you can contact the Academic Affairs Institutional Review Board:

Academic Affairs Institutional Review Board
David A. Eckerman, Chair
CB #4100, 201 Bynum Hall
The University of North Carolina at Chapel Hill
Chapel Hill, NC 27599-4100
(919) 962-7761 aa-irb@unc.edu

[ ] Check this box if you do NOT want me to use your company's name, or any other names you mention during the interview, in my work.

[X] Check this box to give me permission to use your company's name in my work. *You do NOT have to check this box if you do not want to.*

I give Harry Ahlas permission to conduct an interview with me and print what I say in my interview into his paper on Web interoperability.

Sign Here

Janet L. Foots
CEO
ONTHERAIL
Tatu

2-7-01
Date

School of
Information and Library Science
The University of North Carolina at Chapel Hill

Consent form–*Browser, Platform, and Monitor Compatibility Issues with Regard to Web Design*

To Participant: Kay Marie Jacobson

I am writing a research paper about interoperability--how people make Web pages look similar on different types of browsers (like Microsoft Internet Explorer and Netscape Navigator) as well as different computers and monitors. For my work, I need to interview 20 developers and managers like you to understand how and why you make your decisions regarding interoperability.

Before I can ask you any questions, I need you to sign this form which gives me permission to interview you and cite your words in my research. The interview should last approximately fifteen minutes; you may withdraw from the study at any time. I will not use your name in my paper, and my research poses no risk to you in any way at all.

If you have any questions regarding my work, you can contact me or my advisor, Dr. Gary Marchionini:

Harry Ahlas
616 Craige, UNC Residence Hall
Chapel Hill, NC 27514
(919) 914-3983
hahlas@email.unc.edu

Dr. Gary Marchionini
(919) 966-3611
marchionini@ils.unc.edu

If you have any questions about your rights as a participant, you can contact the Academic Affairs

David A. Eckerman, Chair
CB #4100, 201 Bynum Hall
The University of North Carolina at Chapel Hill
Chapel Hill, NC 27599-4100
(919) 962-7761 aa-irb@unc.edu

[ ] Check this box if you do NOT want me to use your company's name, or any other names you mention during the interview, in my work.

[ ] Check this box to give me permission to use your company's name in my work. *You do NOT have to check this box if you do not want to.*

I give Harry Ahlas permission to conduct an interview with me and print what I say in my interview into his paper on Web interoperability.

Kay Jacobson
Sign Here

02/08/01
Date

School of
Information and Library Science
The University of North Carolina at Chapel Hill

Consent form–*Browser, Platform, and Monitor Compatibility Issues with Regard to Web Design*

To Participant:

I am writing a research paper about interoperability—how people make Web pages look similar on different types of browsers (like Microsoft Internet Explorer and Netscape Navigator) as well as different computers and monitors. For my work, I need to interview 20 developers and managers like you to understand how and why you make your decisions regarding interoperability.

Before I can ask you any questions, I need you to sign this form which gives me permission to interview you and cite your words in my research. The interview should last approximately fifteen minutes; you may withdraw from the study at any time. **I will not use your name in my paper**, and my research poses no risk to you in any way at all.

If you have any questions regarding my work, you can contact me or my advisor, Dr. Gary Marchionini:

Harry Ahlas
616 Craige, UNC Residence Hall
Chapel Hill, NC 27514
(919) 914-3983
hahlas@email.unc.edu

Dr. Gary Marchionini
(919) 966-3611
marchionini@ils.unc.edu

If you have any questions about your rights as a participant, you can contact the Academic Affairs Institutional Review Board:

Academic Affairs Institutional Review Board
David A. Eckerman, Chair
CB #4100, 201 Bynum Hall
The University of North Carolina at Chapel Hill
Chapel Hill, NC 27599-4100
(919) 962-7761 aa-irb@unc.edu

[ ] Check this box if you do NOT want me to use your company's name, or any other names you mention during the interview, in my work.

[✓] Check this box to give me permission to use your company's name in my work. *You do NOT have to check this box if you do not want to.*

I give Harry Ahlas permission to conduct an interview with me and print what I say in my interview into his paper on Web interoperability.

Sign Here

2·7·2001
Date

School of
Information and Library Science
The University of North Carolina at Chapel Hill

Consent form–*Browser, Platform, and Monitor Compatibility Issues with Regard to Web Design*

To Participant:

I am writing a research paper about interoperability—how people make Web pages look similar on different types of browsers (like Microsoft Internet Explorer and Netscape Navigator) as well as different computers and monitors. For my work, I need to interview 20 developers and managers like you to understand how and why you make your decisions regarding interoperability.

Before I can ask you any questions, I need you to sign this form which gives me permission to interview you and cite your words in my research. The interview should last approximately fifteen minutes; you may withdraw from the study at any time. **I will not use your name in my paper**, and my research poses no risk to you in any way at all.

If you have any questions regarding my work, you can contact me or my advisor, Dr. Gary Marchionini:

Harry Ahlas
616 Craige, UNC Residence Hall
Chapel Hill, NC 27514
(919) 914-3983
hahlas@email.unc.edu

Dr. Gary Marchionini
(919) 966-3611
marchionini@ils.unc.edu

If you have any questions about your rights as a participant, you can contact the Academic Affairs Institutional Review Board:

Academic Affairs Institutional Review Board
David A. Eckerman, Chair
CB #4100, 201 Bynum Hall
The University of North Carolina at Chapel Hill
Chapel Hill, NC 27599-4100
(919) 962-7761 aa-irb@unc.edu

[ ] Check this box if you do NOT want me to use your company's name, or any other names you mention during the interview, in my work.

[ ] Check this box to give me permission to use your company's name in my work. *You do NOT have to check this box if you do not want to.*

I give Harry Ahlas permission to conduct an interview with me and print what I say in my interview into his paper on Web interoperability.

Sign Here

Date

School of
Information and Library Science
The University of North Carolina at Chapel Hill

Consent form–*Browser, Platform, and Monitor Compatibility Issues with Regard to Web Design*

To Participant:

I am writing a research paper about interoperability—how people make Web pages look similar on different types of browsers (like Microsoft Internet Explorer and Netscape Navigator) as well as different computers and monitors. For my work, I need to interview 20 developers and managers like you to understand how and why you make your decisions regarding interoperability.

Before I can ask you any questions, I need you to sign this form which gives me permission to interview you and cite your words in my research. The interview should last approximately fifteen minutes; you may withdraw from the study at any time. I **will not use your name in my paper**, and my research poses no risk to you in any way at all.

If you have any questions regarding my work, you can contact me or my advisor, Dr. Gary Marchionini:

Harry Ahlas
616 Craige, UNC Residence Hall
Chapel Hill, NC 27514
(919) 914-3983
hahlas@email.unc.edu

Dr. Gary Marchionini
(919) 966-3611
marchionini@ils.unc.edu

If you have any questions about your rights as a participant, you can contact the Academic Affairs Institutional Review Board:

Academic Affairs Institutional Review Board
David A. Eckerman, Chair
CB #4100, 201 Bynum Hall
The University of North Carolina at Chapel Hill
Chapel Hill, NC 27599-4100
(919) 962-7761 aa-irb@unc.edu

[ ] Check this box if you do NOT want me to use your company's name, or any other names you mention during the interview, in my work.

[X] Check this box to give me permission to use your company's name in my work. *You do NOT have to check this box if you do not want to.*

I give Harry Ahlas permission to conduct an interview with me and print what I say in my interview into his paper on Web interoperability.

Sign Here

3/30/01
Date

Bibliography

Anderson, Joel & Chris Kunicki. (No Date). *More real world Cross-Browser HTML Development*. WebTechniques. [On-line]. Available: http://www.webtechniques.com/archives/2000/08/kunicki/ [2001, Jan. 24].

Apache. (2001). *Cocoon*. [On-line]. Available: http://xml.apache.org/cocoon/ [2001, Mar. 31].

Boles, David & Siciliano, Rachael A. [2000]. *Writing Cross-Browser Dynamic HTML*. Cnet. [On-line]. Available: http://www.builder.com/Programming/Dhtml [2000, Oct. 17].

Burns, Joe. (No Date). HTML Goodies. [On-line] Available: http://www.HTMLgoodies.com [2000, Oct. 7].

Holzschlag, Molly E. (1998). Web by Design. Alameda: Sybex.

Isaacs, Scott. (No Date). Cross-Browser DHTML Authoring. Inside DHTML. [On-Line]. Available: http://www.insidedhtml.com/writeOnce/talk1/start.htm [2000, Oct. 14].

Jones, Joseph R. (1997). *Cascading Style Sheets*. New York: MIS:Press.

Macromedia. (No Date). *Using JavaScript to Detect Flash Player*. [On-line]. Available: http://www.macromedia.com/software/flash/ [2001, Jan. 23].

Mulder, Steve and Michael Brandt. (1999). *Sizing up the Browsers*. Wired Digital. [On-line]. Available: http://www.hotwired.lycos.com/webmonkey/reference/browser_chart/ [2000, Oct. 7].

Niederst, Jennifer. (1996). Designing for the Web: Getting Started in a New Medium. Sebastopol: O'Reilly & Associates.

Redman, Stephanie A. (1999). Taking the Leap into New Media. Cincinnati: F&W.

Shiran, Yehuda. (2001). *Doc JavaScript's Tip of the Day Archives*. WebReference.com. [On-line]. Available: http://www.docjs.com [2001, Feb 11, 2001].

Sinclair, Joseph T. (1999). Typography on the Web. San Diego: Academic Press.

Steinman, Dan. (1998). *The Dynamic Duo—Cross-Browser Dynamic HTML*. [On-line]. Available: http://www.dansteinman.com/dynduo/ [2001, Feb. 12].

Vitanza, Victor J. (1998). Writing for the World Wide Web. Needham Heights: Allyn & Bacon.

W3C. (2000). *User Interface Domain*. World Wide Web Consortium. [On-Line]. Available: http://www.w3.org/MarkUp/ [2000, Nov. 27].