

Browser Battles

[Web Smart Newsletter](#) originally published July 2002 - Updated July 2006. By Eric Holter.

Browser Battles

Web development would be great if weren't for browsers. Of course without them we would still be back in the days of Lynx - anyone remember that old text-only browser? Browsers have made the web what it is today, but they have also made developing sites for the web an extremely tricky business. This month's newsletter is part one of a two part series on browsers. Part one will review the history of the Netscape/Internet Explorer browser war and where the battle lines are drawn today. Next month we'll highlight current design issues that can be problematic with regard to browser compatibility.

Way, way back in 1993, Marc Andreessen (the creator of Netscape) was working on Netscape's predecessor "Mosaic." The web was already in existence, but the language of the web, HTML, was still in its infancy. During that time, the Mosaic browser added a new feature that led to the web site rush of 1995-96: images. The Mosaic browser could display graphics within a web page. Suddenly non-geeks started to get interested in the web and that started it all. It also marked the first shot in a browser war that has continued to this day. Suddenly web page authors had a problem. Now that they could use graphics as navigation buttons, how would all the web users with Lynx text-only browsers view the page? Or what about those people using a 24-baud modem with the image feature turned off? Fortunately there was a simple element to solve this problem: the "ALT" tag. If a web page author added "ALT" text to an image, the text-only browser would display the text in the graphic's place. Of course not all web page authors knew about this, or bothered to add the "ALT" text if they did.

Browser compatibility issues like this were destined to become the bane of a web developer's existence.

Enter Netscape

Marc Andreessen, seeing the potential for the web to become much bigger than it was at the time, saw an opportunity to build a much better browser than Mosaic, and so he did. Netscape, then known as Mozilla, was released in October of 1994. It was cool. It allowed you to change the colors of links and visited links. You could add background colors and graphics. No more gray web pages with blue and red links! Of course, if you were still using Mosaic, you were out of luck as far as these cool new features went. Most web page authors (yet to be referred to as designers) would simply add text or a graphic saying "best if viewed in Netscape" to their pages.

There was a problem though. The World Wide Web Consortium is an organization that determines the approved HTML coding standards. While the W3C intended to add most of the features that Netscape included in its browser, they had not yet decided on, or released the new HTML standard that included some of these new elements. Netscape had jumped the gun. Although technical purists objected, everyone else was thrilled at the new features, and didn't really care whether or not the W3C approved of what Netscape did. It wasn't too long before Netscape 2.0 was released with another set of new formatting features, such as tables, frames, and the dreaded "blink" tag. Things were moving forward quickly, and not much thought was given to how some of these new features would affect the older browsers that many people still used. This trend would grow more and more problematic in the future with the inclusion of JavaScript, Java Applets, and many other complex web page elements.

Now that the look and feel of a web page could be (somewhat) controlled, designers began to take a real interest in the web. Many designers were unaware of browser compatibility issues, but this was not a major problem since there was only one dominant browser anyway. Additionally, the typical web user did not have high expectations for how a web page displayed in their browser. If a site wasn't generic gray it was considered well designed.

The cold war begins

"What! Something is happening in the world of computers, and I don't own it," shouted Bill Gates, and soon Microsoft Internet Explorer 1.0 was born. Anti-trust issues aside, the world wide web was now engaged in a civil war between two entrenched factions, the Netscape Rebels, and the I.E. Empire. Suddenly there were two big players each trying to outdo the other's capabilities. Two very divergent paths emerged. The HTML standard controlled by the World Wide Web Consortium was left in the dust. Just about every aspect of how a web page worked was different between Netscape and I.E. For example, some of the more advanced programming features introduced by Netscape used JavaScript. I.E. decided they would use VBScript instead. Now what were developers to do? If we wanted a pop-up window, did they code it in JavaScript or VB script? For a while they were forced to choose which browser was best for a particular site, or else code two versions of each page to satisfy both browsers.

As if these divergent paths weren't difficult enough to contend with, market forces caused Netscape and I.E. to release new "beta" versions of their software on an almost monthly basis. Traditionally, when a company releases a "beta" product, it is usually to a small set of test developers for the purpose of finding and rooting out bugs and problems before it is released to the public. In the case of Netscape and I.E. they used the public as a beta test group, releasing buggy, unstable software to general users. So now developers not only had to determine how we would support the various browsers, but how to avoid crashing the buggy beta releases that were in constant circulation. Often they would have to code a web page with a "hack" to get around a certain bug in a browser. Of course in the next release the bug might be fixed, making the "hack" in the existing site broken in the "fixed" browser.

Before blaming everything on Netscape and I.E., we need to realize that they were just responding to a market that was demanding new features faster than they reasonably could be produced. They wanted cool layouts, better graphics, rollovers, animation, and all the bells and whistles that Netscape and I.E. were making possible. One of the most important features demanded by web page designers and users was the ability to control the type and layout of sites. Netscape and I.E. both responded by providing various ways to control these elements. Of course, each browser implemented the features with slight differences, and where a piece of HTML might work fine in Netscape, it might not work in I.E, and vice versa.

The theoretical ability to control design was increasing, and so web site users started getting used to (and ultimately expecting) cool layouts with awesome interactive features. Designers were under constant pressure to "wow" clients and site visitors with their designs. The more they tried to impress with intricate and original design elements, the harder it became to implement the designs, especially in both browsers.

Browser Compatibility Support Issues

Macintosh and Windows

As the design specifications for web pages became more precise and user's expectations grew, another battle in the web browser war began. The more detailed a web page's design and the more feature rich its functionality, the larger the divide between the browsers. Not only did this gulf exist between browsers, but also between platforms. How Netscape worked on the Macintosh was significantly different than how it worked on Windows. There were differences in the browsers themselves, but also in the systems. For example, fonts tended to display smaller on Macs, and larger in Windows (even using the same browser). Color tended to display darker on a PC than on a Mac. Certain features available in I.E. 4.0 on the PC might not be available for Mac users (ain't that always the way).

A little math

If only we had to deal with two browsers and two platforms. Unfortunately this is not the case. To get an idea for how complex these issues can become, let's try and itemize the number of browsers. We'll use only browsers currently in use for this exercise.

[Note: As of July 06 the browser world is even bigger!]

Netscape's most current release is version 6.2.3 (7.0 is soon to be released - be afraid, be very afraid). While Netscape 4.x browsers are used only about 3% of the time, we'll include them for the sake of being thorough. Netscape skipped version 5 (not sure why). Before we count Netscape as having two browsers available, we need to keep in mind those pesky ".2.3" suffixes. Netscape 6.0, for example, has been out for almost two years. The first release was an absolutely horrible browser. Just about everyone avoided it until they released 6.01. In the past two years, seven versions of the 6.0 browser have been released (6.0, 6.01, 6.1, 6.2, 6.2.1, 6.2.2, and 6.2.3). The 4.0 series has about twenty distinct releases. Each of these versions has its own subtleties and bug "fixes" from previous releases. So that's a

total of twenty-seven Netscape browsers. Each of them has the potential of not supporting, or supporting in different ways, the various features used to display content on a given web page.

I.E. has been a bit more restrained in their browser releases. They have had only eight major releases since 5.0.

That gives us a grand total of thirty-five browsers between Netscape and I.E., each one having differences in how they handle tags and display layouts. But there's more. Don't forget that we have the two major platforms, Mac and Windows. This means that we need to double our count, making the total seventy distinct browsers. That's a lot of browsers! Wait, there's more! We can't forget that each of the two platforms has more than one operating system. It's not just Macintosh, it's Macintosh system 9, and system X (plus all the intermittent ".2.1" releases as well). Then there's Windows 95, 98, 2000, NT, and XP. I guess that means we can't just double our original count, we need to multiply it by at least seven - hold on, I need my calculator now! that's 245 browsers!

How can we cope?

The browser war has been intense and frustrating. Every web developer has horror stories about trying to make their sites work in most of these browsers. In the early days the battle lines were divided pretty evenly. That made things very hard. The upshot was that we often had to develop to a lower level, to keep the browser problems to a minimum. Today we are able to do more. Fortunately some of the older, more problematic browsers (Netscape 4.x for example) are so rarely used, that we can confidently ignore them. This still leaves us with plenty of problems addressing issues of browser compatibility. In general Netscape usage has declined. As a result, Netscape is less motivated to add its own unique features and instead complies with existing standards. I.E. has even made some attempts at complying with standards. On the other hand, the advanced features that current web sites demand open up other and sometimes more significant browser compatibility issues.

Browser Compatibility Standards

The first step in handling browser compatibility issues is clearly defining which browser a given web site will support. For us, when a browser's use drops below 3% we no longer actively support it. That does not mean that a site won't work in an older browser. It just means we no longer test with that browser and if display issues arise we do not automatically fix them.

While we form our standards based on the most actively used browsers, and in 95-97% of cases our sites will look great, that doesn't help when our client just happens to be one of the few web users still using an antiquated browser. While we always include our browser compatibility standards in our proposals, it's also a very good idea to ask a prospect or client what browser they use (and perhaps more importantly, which browser their boss uses) before beginning a project. Otherwise you will be facing a real problem toward the end of a project when the client calls saying the site looks awful on their computer (not the time to begin educating them on browser compatibility issues).

Supporting Older Browsers

Supporting the most widely used browsers and using the most efficient means of coding makes perfect objective sense for web projects. However, the decision to limit browser support might not make sense to those people who still happen to use older browsers (this is particularly problematic if the CEO happens to be one of those people). This is why it is extremely important to communicate and discuss browser compatibility standards with your client before a project begins. It's also a good idea to include browser compatibility standards in any proposal or project contract. And please, always ask the question "what browser does the president of the company use?," and maybe what browser her Mom uses.

So what happens when one of the 3% hits the website? Our policy is to make sure the site is still usable (visitors can access information) in the older browsers, even though there may be significant display issues that might make the site look bad. In some cases, we have employed the technique of using a pop-up window to alert the user of an older browser, letting them know that the site was built for 5.0 and up. The window provides a link to download a current browser.

Handling future browsers

One other note regarding communicating browser standards relates to future browsers. At Newfangled, we include the following text in our technical standards, "Newfangled does not guarantee compatibility with future releases of the various browsers. If new browsers do not display information accurately fixes can be made at our hourly rate." While this may seem obvious (how can you support a browser that

doesn't yet exist), it's a good idea to let a client know that future releases of a browser may act differently theoretically "breaking" the existing site. When Netscape 6.0 was first released this was a major problem (that particular browser was unusually buggy, but the Netscape 6.01 release fixed most of the bugs).

What does the future hold?

While browser compatibility issues are by far the most frustrating part of web development, things have improved. The good news is that with every month that goes by, there are fewer older browsers being used. To our great relief the current trend in browsers is toward cross-browser compatibility and adherence to the existing web standards. Hopefully with less pressure for new releases of browsers to offer new and unique capabilities, the newer browsers will remain compliant with existing standards. One can dream, can't he?