

Cross Browser Incompatibility: Reasons and Solutions

ABSTRACT

A good Web site is more than just something to look at, it is functional interactive and flawless. As technologies are becoming smart so we need to be smarter enough to utilize them. With the rapid evolution of web technologies, the complexity of web applications has also grown up. Specially making a web application that works well with cross browser is a great challenge. Clearly, cross-browser means something works with all versions of all browsers to have existed since the web began. By this paper we have pointed out some reasons why applications behave or appear differently in different browsers because if you know the cause, you can get a solution.

KEYWORDS

Software testing, browser compatibility issues, cross browser, script, web automation tool.

1. INTRODUCTION

A few years ago, there weren't many different versions of web browsers to choose from. Now there are over 100 different browsers, not counting different versions of the same product. Quite a number of these names we have never heard of before. The W3C has been battling for years for software companies to produce "compliant" browsers. The idea of compliancy, amongst other things, is to guarantee that a web page looks and function the same under any browser. Many browsers circulating now are not "compliant". Unfortunately, the W3C has not yet able to set a proper baseline, accepted by all the browsers [18]. The good news for developers is that the vast majority of web surfers use prominent browsers as Internet Explorer, Firefox Netscape, Chrome, Opera - approximately 95%. The bad news is that there are over 200 flavors of Internet Explorer, Firefox and Netscape, Chrome. Web pages can look totally different among the different versions. Figure 2 shows usage of different versions of the same browser, a survey of w3schools [1, 4, 6].

As we know a site is developed using one browser much more than the others. Testing across a variety of browsers will reveal issues the developer may be unaware of. Web browser compatibility testing sounds rather technical and confusing - something you ought to let your web developer deal with. The problem is that if your website is not compatible with the plethora of browsers available, it will affect your business reputation. Indeed, before you know it there could be Tweets about you, your website could be discussed in Facebook and who knows how many blog posts slagging you. Think, your company has decided it's time to add that little "e" to the front of its service or product. And thinking that you have tested the site by the latest browsers was clearly inadequate. Assumed wrongly- that if IE8 could see the page, other versions of

Internet Explorer would also be able to do so. Problems like one well-known newspaper site returned several errors when you check morning news, another website found from a search engine was laid out like an explosion in a typesetting factor in Firefox, but looked OK in Internet Explorer. These issues are common and we face every day. Clearly, you can't check all the different browsers easily on your own computer. For instance, if you install Internet Explorer 8, it merely upgrades version 7.

	Internet Explorer	Firefox	Chrome	Safari	Opera
2010	29.70%	44.10%	19.20%	3.90%	2.20%
2009	37.20%	46.40%	9.80%	3.60%	2.30%
2008	46.00%	44.40%	3.60%	2.70%	2.40%
	Internet Explorer	Firefox	Mozilla	Safari	Opera
2007	56.00%	36.30%	1.20%	1.80%	1.60%
	Internet Explorer	Firefox	Mozilla	Netscape	Opera
2006	60.60%	29.90%	2.50%	0.20%	1.50%
2005	68.90%	23.60%	2.80%	0.40%	1.50%
	Internet Explorer		Mozilla	Netscape	Opera
2004	76.20%		16.50%	1.70%	1.60%
2003	84.90%		7.20%	2.60%	1.90%
	Internet Explorer	AOL		Netscape	
2002	83.40%	5.20%		8.00%	

Figure 1: Comparison of usage of different browsers

2010	Total	IE 9	IE 8	IE 7	IE 6
October	29.70%	0.40%	17.30%	7.20%	4.80%
2010	Total	Chrome 8.0	Chrome 7.0	Chrome 6.0	Chrome 5.0
October	19.20%	0.30%	5.60%	12.30%	0.60%
2010	Total	S 5	S 4	S 3	
October	3.90%	3.10%	0.70%	0.10%	
2010	Total	O 10	O 9.5	O 9	Other
October	2.20%	2.10%	0.10%	0.00%	0.00%
2010	Total	FF 4.0	FF 3.6	FF 3.5	FF 3.0
October	44.40%	0.70%	36.20%	4.50%	2.40%

Figure 2: Usage of different versions of the different browsers.

Make sure you test your site thoroughly with efficient web browser compatibility testing. Owners do not emphasize any form of browser compatibility testing. They also don't know as a result that their reputation has been harmed, or that they've lost business to someone else. It is worthwhile to give attention to your website fall foul of the varying standards and ways in which browsers load [7, 8].

This document describes, and is written to conform to, author guidelines for the journals of AIRCC series. It is prepared in Microsoft Word as a .doc document. Although other means of preparation are acceptable, final, camera-ready versions must conform to this layout. Microsoft Word terminology is used where appropriate in this document. Although formatting instructions

may often appear daunting, the simplest approach is to use this template and insert headings and text into it as appropriate.

2. Challenges

Decide number of browsers to be considering for cross browser testing. As we know it is not easy to test our web application on unreasonable number of browsers because complexity and hence the time and cost of testing a web application is directly proportional to the number of browsers on which the application is going to be tested.

$$C = (B * O + A) * T$$

C=Complexity

A=Third party components

B=Different Browsers used for AUT

O= Number of OS

T= Types of testing Performed

3. Causes to Incompatibility

There is no certified tool or way which can guarantee to make an application sure for cross browser. But still research shows some areas if worked out properly during designing of a web site can assure to work well with cross browser.

- HTML Tags
- CSS
- Font Rendering
- DOM
- Scripts
- Addons
- Third Party entities

Now let's have a look of these one by one.

3.1. HTML Tags

Back during the heyday of the Browser Wars, both Netscape and Microsoft tried to get a competitive edge by running ahead of the HTML standards, inventing their own tags and attributes. Problem is further complicated by browser-specific "HTML extensions."

The best way to minimize these problems is to pay attention when building your Web pages. Avoid using HTML extensions and be careful about using cutting-edge features of the language that may not yet be supported by all the major browsers. Another issue with HTML tags is that Different Browser Versions. The major difference between two versions of the same browser is their support for newer portions of the HTML language. A new browser is generally better at displaying Web pages with additional features than an old one. That presents a problem because after some length of time a new version of browser comes. But a significant minority of people would still use the previous version [10, 15].

A practical rule of thumb is to design your pages to work for the last two versions of the major browsers.

While a handful of people use browsers older than that, their numbers aren't large enough to justify the sacrifice required to support them.

An HTML error is some spot on a Web page where you've violated the official rules of HTML. For example, you may have two tags that overlap one another. In practice, the major browsers are robust and forgive many of these HTML errors. But not all browsers forgive the same errors. One browser may display Web page without error, but another browser may be seriously affected by the same error.

For example, if HTML code on a page forgets to close a single TABLE tag, which is a violation of the HTML rule. Internet Explorer forgives this error and displays the page correctly but on the other hand Netscape 6 cannot recover from the errors and doesn't display the page at all! Flaw in Netscape 6 is that it will ignore both Internet Explorer and Netscape Navigator 4 proprietary HTML tags as well as tags from older authoring tools. Thus a page may not display in Netscape 6 as it does in Internet Explorer 4 and Netscape Navigator 4. So unfortunately it is also true that always it doesn't work to update with latest version and feel confident. Therefore html error plays a very important role in browser compatibility testing.

3.2. Cascading Style Sheets

CSS is a style sheet language used to describe the presentation semantics (the look and formatting) of a document written in a markup language. Although the CSS1 specification was completed in 1996 and Microsoft's Internet Explorer 3 was released in that year featuring some limited support for CSS1. Then came CSS2 and many of browsers additionally implemented parts of CSS2. As of August 2010, no browser has fully implemented even CSS2[14]. Think if a web designer uses the features of CSS3, how many of these are going to work on cross browser. So all these points are worthwhile to build up a good web application. Infected There are different rules of CSS in Internet and Firefox. So if it works well with IE, many times it spoiled the Firefox users. The below list need to fit perfectly in the below box with height 245px. It should look same both in Internet Explorer and Mozilla Firefox.

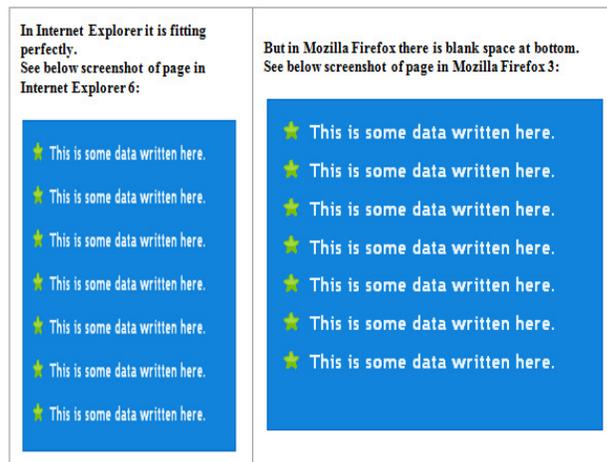


Figure 3: CSS problem

This is a common height problem in CCS, for Firefox and Internet Explorer.

3.3. Font/Page Rendering

Font rendering terms refers how the font will be displayed on the screen with respect to the operating system and browser. Actually some browser believes that the goal of the algorithm should be to preserve the design of the typeface as much as possible, even at the cost of a little bit of blurriness, on the other hand some think that the shape of each letter should be hammered to look perfect[2].



Figure 4: Font rendering comparisons

As you can see, Google Chrome 4 builds have some problems with Font Anti-Aliasing, while IE shows a bit more blur than anyone else. Here Firefox 3.6 Beta 5 gave the best results with both Serif and Sans-Serif fonts.

Page rendering refers to the appearance of your page in different browsers. It might be different in appearance and sometimes it might expose a defect [5].

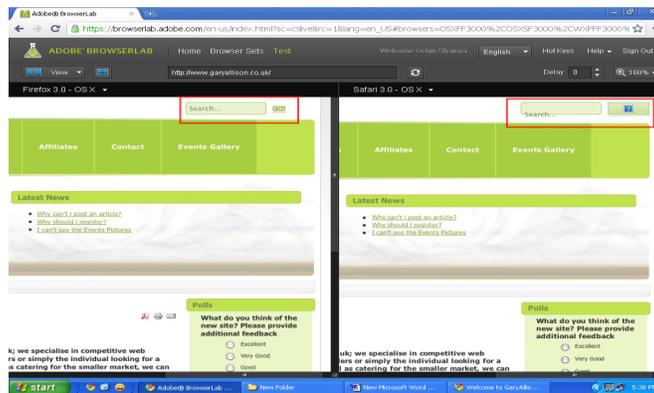


Figure 5: Page rendering

Using Acrobat Browser Lab we can see that in Safari3.0 'Search' option is misplaced and image is also not displayed, however it is working fine in Firefox 3.0.

3.4. DOM

The history of the Document Object Model is intertwined with the history of the 'browser wars' of the late 1990s between Netscape Navigator and Microsoft Internet Explorer, as well as with that of JavaScript and JScript,

When a browser receives HTML code from a web server, it creates a structured overview of the HTML document. This overview is known as DOM. And through its structure JavaScript can access the various parts of the document. The DOM is usually visualized as a tree-like structure. In fact, IE and other browser handle it differently [17]. IE5 and Netscape 6 can be considered the first two browsers to begin supporting the modern DOM, or DOM level 2. More modern browsers that followed such as Firefox offer complete support for DOM 2 (though not 100%). However, there are differences between Internet Explorer's DOM and Mozilla's DOM. The tree structure of DOM is generated differently in both. The most important difference is that how they handle white-spaces text nodes. When XML generates, it often contain white-spaces between the nodes. Actually Internet Explorer and Firefox both generate the structure overview entirely different. So java script unable to work in one browser and it works fine in the other one. So Dom plays an important concern in the cross browser concept.

3.5. Scripting Languages

A scripting language is used to coordinate and combine components that are written in some other language. Script languages are easier and faster to code in than the more structured and compiled languages such as C and C++. However, a script takes longer to run than a compiled program since each instruction is being handled by another program first (requiring additional instructions) rather than directly by the basic instruction processor [11-13]. Fact is that there are so many scripting languages available today. IE supports both VB Script and JavaScript while Netscape & Mozilla supports only JavaScript. So VB script cannot be considered a cross browser scripting language but infect neither can JavaScript. Let's dig out the concept by taking the example of JavaScript. Both Microsoft and Netscape browsers support JavaScript, but sometimes in slightly different ways. JavaScript developers have always pushed outside the envelope of browser capabilities; thus, there have been JavaScript errors as for as long as the scripting language has been around. Some very popular web browsers are two to three years behind in development, as far as correctly supporting JavaScript is concerned.

Unfortunately, until all web browser developers agree on one set of standards for processing JavaScript and other scripting languages, we are going to continue to see JavaScript errors. All browsers process the scripts differently; therefore, all report JavaScript errors differently. For example Google Chrome uses the Open Source V8 JavaScript engine for processing and handling JavaScript code execution. The JavaScript handling procedures used in Chrome's V8 are quite differently than those in other browsers. Typically, if a browser encounters a piece of JavaScript code that it cannot process correctly, it will display one of those infamous JavaScript error messages. JavaScript scripts are usually not a single process or executable operation. Usually JavaScript code contains a series of commands or operations. All browsers seek to execute JavaScript processes or commands in the order they are written. However, other browsers will produce an error message that asks the user if he wants to continue processing the script, if it encounters a piece of code it cannot execute. Conversely, Google's Chrome simply continues to run the script, leaving the error prone code unchecked and normally produces no JavaScript error. Does this mean there NO JavaScript errors in Chrome? No, there are some, if not as many.

3.6. Plugin/Add-on

A plug-in or an add-on is a set of software components that adds specific capabilities to a software application. Your browser's basic functionality is to allow you to access web contents but if we want to listen music from a website then we have to equipped your web browser with the capabilities to access the music files but as we know this is something done by the music player. So you add this player to your browser to access the music on the web. For example If there are audio or video components on your page, test that the attributes of the plug-in are correct. This includes volume, replay, and auto-start [16]. For example

- Does the audio/video clip starts and stops at the correct places?
 - Is the audio/video quality acceptable?
- Actually installing plug-in is not a big deal; many of these are freely available on the web.

But there are certain issues associated with plugins. Most of plugins are ActiveX and ActiveX is treated in Internet Explorer and Firefox differently. Once downloaded on a user's computer, the ActiveX control becomes part of the operating system with the ability of tampering with every piece of hardware and software on the machine. Instead of restricting ActiveX's functions, Microsoft chose another security route. Every time Internet Explorer needs to download a new ActiveX control, it launches a pop-up window asking the user if he wants to proceed. In response to widespread criticism of the ActiveX vulnerability, Microsoft increased ActiveX security with the release of Internet Explorer 7, disabling all but the most common ActiveX controls - Windows Media Player, Flash Player, and Adobe Reader. So the problem is mostly Plugins are actually ActiveX and ActiveX are invented by Microsoft for Internet Explorer and not supported by Mozilla Firefox directly. Other browsers can use ActiveX but with the help of another plugins and these are handled differently.

3.7. Third Party entities

When launching a web application, remember that it won't operate in a vacuum. There are many third-party applications that could interfere with its compatibility, For example:

- Live chat
- Checkout processes
- Search plug-ins
- RSS Feeds
- Embedded videos or audio players
- Ad servers or embedded ads
- Web analytics packages
- Blogs, forums and message boards
- Social networking modules or toolbars (like Tweet streams or Digg).

4. Solutions

As we know that it cannot be practically feasible to test an application on your system manually because there are so many browsers available in the today's market and infect your system also not going to support you to have multiple versions of the same browser. For example if you have IE7 then you can update to IE8 but concurrently both versions cannot exist simultaneously on a system[3]. A good way to test your site on cross browser is to use cross browser automation tools.

Using these tools one can identify the problem and hence can take action in appropriate time. Here are some useful tools.

4.1. Adobe Browserlab

Adobe Browserlab is a web hosted service which allows designers to test designs on multiple browsers and operating systems [5]. In this, system are included a zoom function and measurement tools like rulers and guides, and the ability to move quickly to specific areas of a screenshot. Browser sets allow designers to customize, edit, and save a combination of browsers to test.



Figure 6: Adobe browserlab

4.2. Netrenderer

IE NetRenderer is free browsers compatibility tester. With IE NetRenderer you can check how a website is rendered by different versions of Internet Explorer [21].



Figure 7: Netrendering

4.3 Browsera

Browsera is meant to be used as a testing tool, not just a screenshot tool, and works at the site-level instead of the page level. Browsera crawl the pages on your site and actually test them for layout inconsistencies and scripting errors [19, 22].



Figure 8: Browsera

4.4 Litmusapp

Litmus shows you screenshots of your websites as they look across all major web browsers. After you finished testing, you can get a full compatibility report ready for review by your clients.

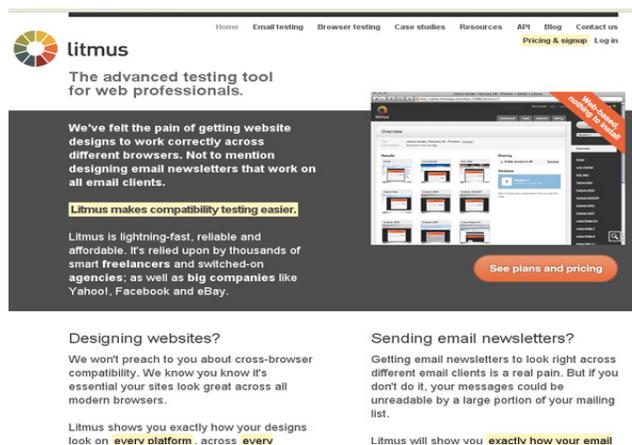


Figure 9: Litmusapp

4.5 Browsrcamp

Browsrcamp allows you to test the compatibility of your design with Mac OS X browsers. You can check compatibility of your design with Safari 3.1.2 [20]. Simply add your website URL and you will get your website screenshot.

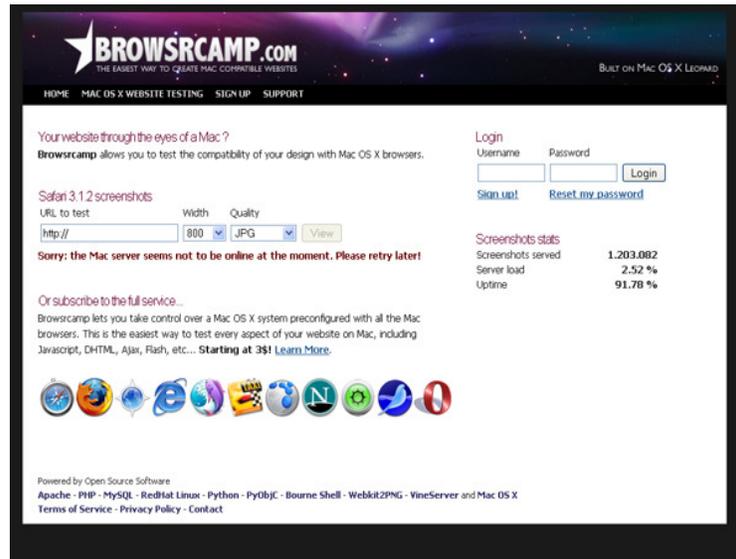


Figure 10: Browsrcamp

5. CONCLUSIONS

As the technology is changing rapidly, the application designers believe to give the best features and functionalities to the applications. But most of the applications are not stand alone. Users have different platforms and browsers. Designers cannot assume that their application will run fine and display and work for all the browsers without cross browser testing. Hence to avoid loss of business and reputation it is very important to pay attention to cross browser issues. In this paper we have pointed out a few important concerns regarding this. As if we know the problem it is easy to plan out the areas that should be cross verified. Indeed it is hard to find out compatibility of your application for each existing browser. Hence the help of cross browser automation tools is indeed a good way in this respect. Only then users will be able to get unbiased what the applications can deliver them.

ACKNOWLEDGEMENTS

Author express his sincere thanks to Dr. Rakesh Aggarwal, D.C.S.A., Kurukshetra University for his valuable guidance. Also I extend my regards to Dr. S. S. Tyagi, HOD, Department of CSE, F.E.T., Manav rachna Internation University for his throughout support and esteem guidance.