WEB EVOLUTION

A thesis submitted to the University of Manchester
for the degree of Master of Science
in the Faculty of Engineering and Physical Sciences

2008

By
Alex Qiang Chen
School of Computer Science

Contents

List of Tables

List of Figures

Abstract

The World Wide Web (web) is a heterogeneous environment that is in constant evolutionary change. This includes technological changes such as JavaScript, the management of data structures used to present the web content such as the Extensible HyperText Markup Language (XHTML), and guidelines such as the Web Content Accessibility Guidelines (WCAG). A lag was noticed between the time these standards and recommendations were introduced to when they were adopted by the developers. This causes a disconnection between the actual user experience, and what was expected by the technology stake-holders. In this study, we investigate the relationship that surrounds these issues, especially those involving the web user interface.

Different sets of data were collected to look at the current and long term slices of websites, and the correlation between the top websites and a set of randomly selected websites. Our results show a trend that new standards and recommendations get adopted faster by the top websites than the random websites. The time taken for this adoption varies between the different types of standards and recommendations; for example, the top websites on average get adopted one year faster than the random websites for a major (X)HTML standards, while it will take on average two years for a graphical format to get adopted. An initial decline in JavaScript usage was noticed for the past year (2007-2008), although a continuous increase in Asynchronous JavaScript and XML (AJAX) usage was observed. Cascading Style Sheets (CSS) took nine years to get adopted by $> 50\%$ of the random websites, however a healthy growth was predicted to continue. After ten years, it was observed that $< 10\%$ of the websites conform to the WCAG. By understanding these evolutionary trends we can inform and predict web development into the future.

Declaration

No portion of the work referred to in this thesis has been submitted in support of an application for another degree or qualification of this or any other university or other institute of learning.

Copyright

Acknowledgements

I would like to thank Dr. Simon Harper for his time, guidance and suggestions during this entire project, Darren Lunn for his advice and guidance when writing this thesis, and everyone in the Human Centred Web group.

Chapter 1

Introduction

The web is a medium that provides an environment where files (this can be in the form of graphical formats, plain text, or audio) are interlinked, and can be accessed publicly via the Internet. It is the largest existence of hyper linked hypertext documents, and it is constantly changing and growing. From the beginning when the web was created, HTML was defined as the data structure format to be transmitted over the network, and Hypertext Transfer Protocol (HTTP) was created for traversing hypertext links [8]. Although the first web browser for Windows operating system (Mosaic web browser) was released in 1993, the widespread use of the web really began only around 1995; this was around the same time when Microsoft released the Internet Explorer as part of Windows 95 [7]. Could this be responsible for the widespread use of the web or was it just a coincident?

At the rate that the web is evolving, it is difficult to keep abreast of the changes to the web content and technologies. Thus the guidelines, recommendations and standards were frequently being revised and generated by the World Wide Web Consortium (W3C) to improve accessibility to web content, and to provide better web experience. Often the standards, guidelines and recommendations take time to be accepted, and were slow to be adopted by web developers, authors, and user-agents. A lag was noticed between the time these standards and recommendations were introduced to when they were adopted. This causes a disconnection between the actual user experience, and what was expected by the technology stake-holders. Thus, understanding the evolution of the web is essential as it will help us to understand the relationship between the underlying standards, recommendations, guidelines and their adoption time. This study attempts to

understand these issues while focusing on the human factors surrounding the evolution of the web user interface.

A wide variety of technologies, standards, guidelines and recommendations are available for web content authors to use, and to conform. Early studies of web content reported that besides HTML, graphical formats such as GIF and JPEG formats were used for transporting images over the web even before 1997 [22]. More recently, the increase in popularity for the technologies such as client-side scripting, CSS and XHTML saw them included in some studies [19, 15]. Although a number of guidelines were generated by the Web Accessibility Initiative (WAI) to improve web content accessibility, webmasters often do not find it beneficiary to take up these guidelines. This is because of the small number of user population it will benefit, thus it does not return huge economical benefits [42]. A recent study presented that a small percentage of the federal websites in the United States of America (US) and the government service websites across Europe conforms to these guidelines. The study also reported that these guidelines were keen to be taken up by the Japanese as well [46]. Although these reports highlighted the poor adoption rates to these guidelines, but it also showed that more are beginning to adopt them. From our analysis, we found that on average less than 10% of the web conforms to WCAG which is still seems a little low, however these results do correspond to the results reported by Watanabe and Umegaki [46].

The information required to study the relationship between these issues can be found throughout the web. However the biggest obstacle when collecting information from the web is its size, hence an efficient method must be used. From an empirical study in 1999 it was reported that the size of the publicly indexable web was about 800 million pages [28]. Later in 2005, another study reported that the web had grew enormously by more than 14 times; this was more than 11.5 billion webpages [25]. These studies demonstrated that tracking the changes of the web can be difficult, and the web is evolving and expanding at an exponential rate, however this did not deter researches to be conducted on the evolution of the web. Commonly two types of method were used to track the changes to the web. The first method monitors the packets that passes through the corporates firewall in the proxy server [22]. This method can reliably capture the contents from the websites accessed by the server's users, but it is biased towards the needs and culture of the corporation. Thus it does not give a good representation of the web. The other method (the more popular method) uses a web crawler or

a web robot to crawler and fetch a snapshot of the targeted webpages, and the required data for further analysis [45, 23]. The coverage of this method depends on the quality of the webpages the web crawler was deployed to capture. Hence to overcome this issue, commonly some kind of webpage selection methodology (e.g. page ranking [18] and tagging [3]) will be used to select the web pages. A web robot was chosen for this study due to the scope, comprehensiveness and flexibility of webpages it can be programmed to examine, and capture.

This study will identify the recommendations to questions such as 'Do we rely on technology or guideline adoption?', 'Do we need technical intervention?', 'Will technical interventions be adopted into user-agents?', or 'Should we be led by users, by engineers, or by history?' [27]. To do this, a long term slice of a number of popular and randomly selected websites for the last ten years (1999-2008) were captured. This will allow us to investigate if the random websites, in-general, follows the trends of the popular websites, and to identify the lag between them. Two larger sets of popular and randomly selected websites were also captured from the current web to validate the analysis done for the long term slice of the websites for the last ten years. Our analysis showed that CSS took nine years to be adopted by more than 50% of the random websites, but a healthy growing trend was predicted for the next year. An initial decline in JavaScript usage was noticed for the past year (2007-2008) although an increase in AJAX usage was observed. For a major (X)HTML standards, the top websites on average adopts one year faster than the random websites, while it will take on average two years for a graphical format to get adopted. This report provides the background material in correlation to the research focus to be undertaken in this study. A detailed discussion relating to the issues of the results were followed. The analysis and discussions covered in this study will contribute as recommendations to inform and predict the web development into the future.

1.1 Synopsis

The structure of the remainder of this study is as follows:

Chapter 2: Background provides the necessary technical knowledge required to understand the discussions and work presented in this study. Four major categories were used for easy illustration. They include W3C standards, graphical formats, client-side scripting, and guidelines. A presentation of

the related work was followed to demonstrate the novelty of this study conducted by comparing its purpose with the related previous studies.

Chapter 3: Research Methodologies gives a detailed explanation of the various methods used when conducting this study. Four major stages were used to successfully capture the websites, and extract the required data for analysis. The existence of some missing data from the Internet Archives requires our data to be normalised as it causes an inconsistent volume of data captured. Finally the regular expressions that were used together with our methodologies to extract the necessary data for analysis were presented and discussed in more detail.

Chapter 4: Results And Discussions presents the results and discussions from our analysis. A general trend was noticed that the Alexa top websites does give a good representation of the random web when analysing the W3C standards and the graphical formats trends. From our web content accessibility conformance analysis, no increase to the conformance of WCAG 1.0 guidelines was forecasted. A continuous growth in CSS, Flash, AJAX and JPEG usage was predicted, however a fall in JavaScript usage was forecasted. Finally discussions relating to the further analysis were presented to understand more about the reasons behind the trends such as AJAX, JavaScript, CSS, Flash, and JPEG.

Chapter 5: Conclusion And Future Work concludes that the evolution for both the W3C standards and graphical formats can reliability be represented by the Alexa top websites to give a good idea of how the web was evolving. An initial roll off for JavaScript usage was noticed even though an increase in AJAX usage was found. However from our further analysis, these trends may be due to the websites converting from JavaScript to Flash technology. Finally, history together with the take up of technologies by the users, and the interventions from engineers are the suggested approach to lead the future of the web. This was followed by the discussions of four suggested related future work.

Chapter 2

Background

The elements that contribute to the issues discussed earlier changes concurrently with the evolution of the web. To analyse these issues a background of the related work, foreseeable client-side web technologies and guidelines were provided to support the study.

2.1 W3C Standards

The W3C has been introducing and re-interventing a number of HTML, and styling standards to make web content more accessible and presentable, allowing web designers, developers, and content authors the freedom to express themselves to a wider range of audience. HTML allows the data that are presented over the web to be structured, and interpretable by most popular user-agents. Styles can be added to the structured data by using Cascading Style Sheets (CSS) to control the appearance of both plain text and graphics (see Appendix A.2). However due to the constant rectification to the W3C standards and recommendations, and the introduction of the newer ones, this creates an indirect problem to web content accessibility. In this study, we attempt to understand the trends relating to these issues so that recommendations for future work can be identified. Refer to Appendix A to read more on (X)HTML and CSS.

2.2 Graphical Formats

As mentioned earlier, web content includes a few elements, and graphical formats is one of the important ones. These technologies not only beautify the webpages,

but were also used by advertisers to send their message across the web, and by some to explain difficult to explain subjects. The still graphical formats covered in this study includes JPEG and SVG, the animated graphical formats includes GIF and PNG, and the animated graphical formats with sound includes Flash and SMIL. Refer to Appendix B to read more about the different graphical formats.

2.3 Client-side Scripting

To make websites more interactive/dynamic and to reduce network traffic, client-side scripting is used not only to add interactivity to them, but also allows webmasters to balance the computation work load between the server's and the client's machine; hence this allows more effective use of the network traffic. There are several type of client-side scripting available, only the two more popular ones will be covered in this study; JavaScript and Visual Basic Script (VBScript). Refer to Appendix C to read more about client-side scripting.

A popular method of using client-scripting is the AJAX model. It consist of four elements; JavaScript, the asynchronous technology, XML, and web applications on the server-side. On the client-side JavaScript is used to computed, call the asynchronous technology (XMLHttpRequest object) and parse the returned result from the server. The asynchronous technology is a method that request for a service from the server and return the output results (if any). The returned results are structured as an XML documents and parsed by the client's machine using JavaScript. On the server-side a web application is required to compute these requests [5]. Since this method do not require the webpage to refresh, it is an alternative for many web developers/authors to provide better user experience on their websites, but it is not without its pitfalls. Thus understanding the usage of AJAX will provide a more comprehensive understanding of the evolution of the web.

2.4 Guidelines

Making the web content accessible by all can be a challenging task to the web community. This includes the accessibility by assistive technologies to interpret the content, but most of these technologies are user-agent specific, hence often they are slow to adopt to the latest guidelines. Furthermore, webmasters often

do not find it beneficiary to take up these guidelines since only a small user population will benefit from it, thus it does not return huge economical benefits. In this study only the guidelines set by the Web Accessibility Initiative (WAI) were discussed. Refer to Appendix D to read more about the different WAI guidelines.

2.5 Related Work

The attempt to study the evolution of the web has been around for a number of years with one of the earliest dating back to 1997. Many have tried to study the evolution of the web for different purposes, these includes the improvement of search engine crawlers, identifying web communities for identifying emerging and evolution of trends, and to assess accessibility for people with disabilities using the web. Our study was a general purpose investigation for the usage of client-side technologies. It covers a broad length of time to identify trends that can be used as recommendations for future developments and researches surrounding the web user interface and HCI. The related work were grouped into five sub-sections according to their purpose of the work as follows.

2.5.1 Analysing Web Contents

Many studies had been conducted on the evolution of web contents, and these are interesting information to validate our results, and further justify our proposed work. To begin lets look at the first related study by Douglis et. al. [22]. It was an attempt to quantify the rate, nature, extent of changes to web resources, and 'Can we detect and exploit changes in semantically distinguishable elements of HTML documents?' The information sampled were captured over a three years period from Digital where the traces were collected from the proxy server that passed the requests through the corporate's firewall, and from AT&T's 950,000 records. It was reported that of all the accesses, 69% were images, a fifth was HTML, and the rest were applications/octet-stream (arbitrary untyped data used by applications). As for all the resources, 64% were images, a quarter were HTML, and the rest were applications/octet-stream. The analysis on the types of images were broken down into GIF and JPEG, and the rate of change ratio were examined and presented against HTML, GIF, JPEG, and Octet-stream. When attempting to distinguish the elements in a HTML document, this was done for the attributes

href, the elements img, and the rest were on the types of data available. It was concluded that the rate and nature of change to the web resources, frequency of access, and information lifespan depends a lot on the content type and top-level domain, but not the size. The study laid a good background for our work, however, due to technological advancements, this analysis is required to be done at a much greater depth. Besides this the length of the period conducted for the study focused on a shorter length compared to our proposed study. Finally due to the time when this study was performed, no analysis on the accessibility guidelines were conducted parallel with it as this type of guidelines were not available during the time of the study.

Two studies were published in 2000, one of which was by Brewington and Cybenko [11] that estimated how fast the web was changing and growing, and to formulate scheduling solutions for search engines. Their objectives were to observe the rate of change of the web, and to develop an exponential probabilistic model for the intervals between the changes of a webpage. This was done by using the web clipping service called The Informant[1] to download over 200 gigabytes of HTML data since the beginning of 1999. From this study, an increase in the importance of space-saving technology such as CSS, XML and the usage of second generation tags like <table> and <form> tags were discussed. It was reported that a correlation between the age of a web page and its style (e.g. the number of images and the distribution of content lengths depends upon age) was noticed. A display of the peak and troughs of last-modifications times within a week was presented. An attempt to analysis the frequency of change against the number of images and size of a web page was presented. This data was found consistent with an earlier report by Douglis et. al. [22]. The study went on to model the changes on a single page, the growing web, and the probability of distribution using lifetimes. The observations of basic technology were briefly discussed in this study, and it did not cover enough depth and length to give any logical trends. For instance, the analysis on images did not go a step further to identify the different types of graphical file formats. Hence, this study was not detailed enough to give any conclusive explanation of technological trends around 1999 and 2000.

The second study was by Cho and Garcia-Molina [18]. They started to analyse the evolution of the web so as to study how to build an effective incremental

[1] `http://informant.dartmouth.edu`

crawler for a new version of the Stanford's WebBase crawler. The objective of their study was to investigate how web pages evolve over time so that they can identify and analyse various design choice for an incremental crawler. The evolution of web pages were done to look at how often a web page changes, the lifespan of a page, and how long does it takes for the web to change by fifty percent. At the end an attempt to describe the changes of web pages using a mathematical model was described. Between 17^{th} February 1999 and 24^{th} June 1999 (slightly more than four months), 720,000 pages from 270 sites are crawled. An active crawling approach is used to collect more statistics from pages of interest periodically to see if they have changed. The top 270 sites were identified from the snapshot of 25 million web pages in the WebBase repository at Stanford. To measure the popularity of these sites, the modified PageRank metrics was used. At each site, 3,000 pages were crawled starting from the root page of the each selected sites. Due to the short period of time covered, all the analysis conducted will require further investigation to validate the claims. However, it was reported that about 23% of web pages changes daily, about 47% of web pages changes between more than a day to four months, and about 30% takes more than four months to change. The experiment on the lifespan of a page does not seem convincing due to the short analysis period, but good deductions were suggested for the results. From this time window, it was observed that more than 70% of the pages had a page lifespan of more than a month. Generally government and educational websites have similar trends (i.e. they have a longer page lifespan and are most static) when compared to the .com, .net and .org domain websites (this should be expected). This study focused mainly around the development of an effective incremental crawler, however it does include attempts to cover a superficial evolution of the web. The ideas were good, but it does not cover the type of web technologies and guidelines adopted by these websites, and the time span investigated was too short to identify web trends.

Many of the web evolution works were done to aid the understanding of the web so that improvement can be made to search engines; mainly for crawlers. Fetterly et. al. did a large scale study for the evolution of web pages in 2003 [23]. This was to find out about the dynamic nature of the web is interesting and important to improve the freshness of results returned by search engines so that more of their efforts can be concentrated on crawling and indexing pages which have changed. The study was conducted to answer questions like how fast does

the web change? What were the nature of the changes? Is the change correlated to any other property of the page? How consistent are mirrors and near-mirrors of pages? 151 million HTML pages were crawled between 26 November 2002 and 05 December 2002, and this was later repeated ten more times over a span of ten weeks for this study. Besides analysing the results as a whole (overall), the results were also broken down into their respective top-level domain for each analysis (this includes .com, .org, .edu and .gov). The analysis includes document length, words per document, and different aspects of rate of change to the document. Although the evolution of web pages were analysed, however this study really focused only on the changes (as in text changes) to a document. Thus no analysis were conducted on the evolution of web's client-side technologies, or for the conformance of the web content to any of the major web accessibility guidelines were conducted, and the period covered for this type of investigation was short.

A study by McKeever discussed about the evolution of web content management (WCM) systems life cycle and key market trends for WCM systems [31]. The purpose of the paper was to provide small, informal websites with a model so that they will be able to transform into large, rapid changing websites. Two models were presented to provide a clearer understanding of web content management and its underlying activities. It was reported that due to the rapid development of the WCM systems over the last six years, this had enabled websites to transform from basic HTML text based websites, with webmaster dependence, to sophisticated multi-tiered architecture websites maintained by distributed authors. The market value for such models were also reported to have risen. This study describes a model used by businesses to manage web content life cycle and evolution of market trends. However, it does not cover the depth and nature required by our proposed study.

More recently in 2004, Ntoulas, Cho and Olston did another evolution of the web to assist how search engine should cope with it [35]. Their measurements focus toward the potential interest to search engine designers. It was intended that from the study the evolution of link structure over time, the rate of creation of new pages, new distinct content on the web, and the rate of change for the content of existing papers under search-centric measures the degree of change. 154 websites from October 2002 to October 2003 for a total of 51 weeks were downloaded every week. The selection of websites were picked from the top five

ranked pages from a subset of topical category in the Google Directory. At each site, from the root page, pages are downloaded in a breath-first order for either all reachable pages in each site, or all pages until a maximum limit of 200,000 pages per site is reached. Analysis was done on the fraction of new pages between successive snap shots. The number of pages that existed since the first snap shot till the end of the 51 weeks period, and the number of new pages in a weekly granularity was also done. Other analysis done in this study includes the creation of new web content and the evolution of link-structure for a website, and changes done to existing webpages. Finally an attempt to predict the degree of change on pages and individual sites from the trends of changes of the pages. This study is quite similar to Fetterly et. al. [23], and Brewington and Cybenko [11] with a slight twist. Again it does not bring out the technological evolution and trends, and the time span investigated was only for 51 weeks, hence it does not cover the depth of our proposed study.

2.5.2 Size of the Web

Studies were done to characterise the size of the web and the distribution of information to understand how the web is evolving for many reasons. One of the earliest of these kind of studies was reported by Lawrence and Giles [28], this study attempts to understand the coverage of search engine against the size of the indexable web. The information were divided into eight categories as follows: Scientific/educational, pornography, government, health, personal, community, religion, and societies. Their aims were (1) to analyse the coverage of the indexable web by search engines, and (2) the distribution of information across the eight categories on the publicly indexable web as of February 1999. It was estimated that the publicly indexable web was more than 800 million in 1999. The content that were categorise as scientific/educational was about 6%, and about 83% of the servers contained commercial contents such as company home page which were classified into the remaining seven categories. Metadata were also analysed on the home page (the root of the domain) of each server using the meta tag. It was reported that most of the webpages do not encode any details that identifies the content of the page. Further analysis were done for existence of the keyword and description tags where only 34.2% of servers contained such metadata at that time. Finally the coverage of eleven search engines were presented with Northern Light dominating the coverage (16.0%) with respect to the

estimated web size, and AltaVista and Snap following closely behind with both covering 15.5%. This type of study plays an important role to our study as it gives us an inside to the web usage and the content coverage during that time (for this case February 1999). However, it does not cover the analysis depth required to answer our questions. Furthermore it was only focused for a period of time while our study spans over a much longer period and at greater depth. The study do not tell us any technological trends, but it does give us an idea of the usage of technologies used during that time.

As part of a project by the OCLC office of research to develop and implement a methodology for characterising the size, structure, and content of the web (the results are made publicly available). O'Neill et. al. did a study in 2003 for public websites between 1998 and 2002 to answer questions such as "how big is the web?", "what does it contain?" and "how is it evolving?" [36]. A random number generator to produce random samples of IP (Internet Protocol) addresses was taken from the IPv4 address space; a 32-bit address. To identify if a website exist, the detection of a response code of 200 and document response to the connection was used. Once this is true, the software developed by the OCLC was used to capture the website (Additional steps were taken to ensure IP addresses were unique). Quite surprising it was reported that the accounted approximated size of the public sites was about 1.4 billion web pages as of June 2002 (this seems a little low). It was reported that the web had actually shrank in size, the distribution of public websites was dominated by the US and it is increasing, but a decline in countries such as UK, Canada, Brazil, and Germany were reported. However an increase in public websites for countries such as Japan and Netherlands has been noticed. A similar trend for languages such as Japanese and Dutch was also noticed. Further analysis on metadata was done based on the assumption that if the web page has one meta tag, it is equivalent to one metadata element. The reported mean number of meta tags has shown an increase from 2.75 to 3.14 per sites, and from 2.27 to 2.75 per page. The nature of this study was of close nature and results to Lawrence and Giles study [28]. However, when referring to some of the data, Lawrence and Giles study seems more convincing, and was cited by more. Again this study do not answer our questions because no analysis on technologies, and guidelines were conducted, but it gives us a good inside to the longer period studied of four years (1998-2002).

More recently Gulli and Signorini tried to estimate the current size of the

web and their overlaps in 2005 [25]. This was to update and revise the estimated size of the indexable web that Lawrence and Giles has made in 1999 [28]. The method suggested by Bharat and Broder was used to estimate the over laps of the web in this study [9]. This was conducted by indexing the whole DMOZ.com directory (> 4 million pages) and in blocks of 20 terms. When estimating the size of the indexable web, claims from Google, MSN, Yahoo! And Ask/Teoma were used. The study presented that the coverage of the search engine was around 76.2% for Google, around 69.3% for Yahoo!, around 61.9% for MSN, and around 57.3% for Ask/Teoma. After averaging the relative size estimated and absolution size claimed by the search engines, it was reported that the indexable web is approximately 11.5 billion pages. This study covers the coverage of the indexable web of the search engine with a slight twist to web evolution, however it does not cover in the depth required by our proposed study.

2.5.3 Web Communities

As mentioned earlier, another use of conducting a study on the evolution of the web was to identify web communities and the type of trends that evolved from it. In 2003, Toyoda and Kitsuregawa did a study to understand when and how topics emerged and evolved in the web [45]. A description on how the global behaviour of web community evolution based on four Japanese web archives was presented. Information were extracted from four Japanese web archives crawled between 1999 to 2002 (this constitute 119 millions pages in total). To examine the archives and communities, two parts were used. (1) Extracts whole communities and their relevance from each archive. (2) To provide ways to locate the evolution of communities (e.g. has emerged or growing), a web community evolution viewer was developed to examine how communities evolved over the three years (1999 - 2002). It was reported that from 1999 to 2000, about 60% of URL disappeared, and thereafter between 2001 and 2002, about 30% of URL disappeared in four months. These results were similar to an earlier study report by Cho who reported that more than 70% of the lifespan of a page lasted for more than one month; in their four month observation [18]. Examining the evolution of web community charts were covered via size distribution, types of changes, growth rate, and how communities split, merged, emerged, and dissolved. Finally a demonstration to their features of their web community evolution viewer was presented. This study gives another perspective to understand evolution on a specific topic through

identifying a web community. This is useful for gathering historical topics on the web, on consumer products, tracking and observing social and cultural trends, the emergency of new topics, and quality web communities on a specific topic. However this study does not answer our questions as it was trying to identify web communities and specific topics over the web. No analysis were done for the trends of adoption for web technologies and guidelines, and evolution of the web.

2.5.4 Website Structure

Some research analyse the evolution of the web to improve website structures, and the correlation of these structures and the trend of the web. To overcome the neglect of the problem with evolution and modification of sites, Ricca and Tonella created a set of analysis for a website that enters maintenance, and need to evolve while retaining, and possibly improving its quality [41].

Graphs and several know analyses can be used to model the structure of a website. The proposed analyses that were derived from those used with traditional software systems were divided into two categories (analysis of the structure and analysis of the evolution). To do this the ReWeb was developed to analyse websites. It classifies the websites into four levels by examining the HTML contains, pages with and without frames, client-side scripting languages, applets, and dynamic contents. Fifteen websites from the different types of top-level domain were chosen for the analysis. These websites were downloaded every day for over three months. For each website, the number of webpages, links, and lines of codes were analysed. Detection for JavaScript and dynamic objects were also done for each website. The analysis done to detect web technologies such as JavaScript and dynamic objects were quite basic, and no further analysis were conducted on whether exactly what type of technology was used. The other results reported focused mainly on the website structure/graph. Great ideas were used to analyse the web pages, however no further pursue on the actual type and version of the technologies were present. Although this study do conduct basic analysis on the different technologies, it does not detail enough when identifying the technological trends, and the period conducted for the study was too short for this type of analysis.

From a study in 2001, Cherkasova and Karlsson attempted to understand the nature of web traffic to a website for three different websites to provide a proper method for designing and provisioning the current, and future web services [16].

The web servers log file data from these websites were analysed to characterise websites access patterns, the dynamics, and the evolution of them over time. For two of the websites analysed, data were collected over a duration of five months, and a third website for a month. Most of the analysis conducted were related to web traffic, however it was reported that a trend was noticed that websites were using more graphical contents per HTML page. In this study only basic web content investigations were conducted, and it focused mainly on the traffic of a website, thus it does not cover the depth and nature of our proposed study.

A study by Amitay et. al. was conducted to explore whether websites of two unrelated or competing corporations, universities, or two different web directories exhibits similar structure patterns, despite being designed by different webmasters [3]. It aims to categorise websites by functionality so as to detect what the website is and not what is it about. This was done by crawling 325 websites from the web graph data provided by AltaVista in late 2001. From this data, websites were categorised into eight different functionalities based on (1) aggregate structural properties of the websites, and (2) connectivity patterns between the website and the rest of the web. However the reported results were based mainly on the two classifiers that they have developed. This type of method is usually employed to focus on a topic or a type of community on the web. Very little analysis was done on the technologies used by the websites. Thus it does not answer any of our proposed arguments.

2.5.5 Web Content Accessibility

Finally the evolution of the web was also studied to understand the impact of it on web content accessibility. Hackett et. al. did a study in 2005 to analyse the effects of technological advances in web design have on accessibility for people with disabilities [26]. From the study a method to determine how changes in web design have affected accessibility over time was developed. To do this they have utilised a proven metric (Web Accessibility Barrier score) to assess accessibility, and to employ the complexity algorithm to assess the complexity of a website. Two categories of websites were compared. (1) A random sample of general websites that was selected for each of six years between 1997 and 2002, with a different sample used each year. (2) For the US government websites, the same websites were followed through all the years (1997-2002).

A list of random websites was based on the Top 500 ranked websites by Alexa[2] on 28 July 2003. However websites that could not be located, websites that contained explicit adult content and non-English websites were not included. This was left with 221 potential websites to be included to the random websites population. For comparison, a total of 22 US government websites were selected from the 100 Top Government Site[3] in 2003. The Wayback Machine (by Internet Archive) was used for each of the 221 random websites, and the 22 government websites to capture the archives (at least one archive) of these websites between 1997 and 2002. Both accessibility and complexity were evaluated. When evaluating accessibility, each of these websites was assessed using the Web Accessibility Barrier score that looks at 25 checkpoints based on WCAG and Section 508 guidelines. Complexity was assessed using a method where the summation of the number tags, scripts and objects to the product of each of their points to provide the complexity score.

From their results, generally the US government sites kept their complexity score rather low and were much better in Web Accessibility Barrier score when compared to the random sites. Quite often increasing the complexity of a web page will cause developers to include complex components to a web page, but this does not have to contribute to increasing barriers to accessibility. The US government website had managed to remain accessible despite increasing their complexity by limiting the number of scripts used in web page design. The type and popularity of web technologies used we not considered in this study, hence it cannot answer our questions. Without analysing the take up of technological evolution together with the adoption of the guidelines, our proposed questions cannot be answered.

[2] `http://www.alexa.com`

[3] `http://www.100topgovernmentsites.com`

Chapter 3

Research Methodologies

Our methodologies were employed together with the recommendations discussed in [27]. As discussed earlier, there are two methods commonly used to capture information from the web (see chapter 1). Our chosen method uses web robots to capture the data from our targeted webpages. A web robot is a web application that will crawl a set of selected webpages and return with the captured source code of the webpages and the necessary data for further analysis. However, this will only capture the current version of the webpage. In order to capture the historical data of a website for the past ten years, the Internet Archive[1] data which keeps a snapshot of Alexa's[2] web crawled history [1] was used. In this project all the web applications used to carry out our tasks were written using PHP: Hypertext Preprocessor (PHP). PHP version 5 was chosen due to its simplicity and capability as a server-side scripting language.

A separate application was used to conduct the web mining processes to collect useful data from the captured webpages for specific analysis. From the analysis, trends can be identified, and together with the other results, answers to the questions that motivate this research can be determine.

Four sets of data were captured for this study. Two sets were used to look at a long term slice of websites for the past ten years, another two larger sets were used to give an in-depth look of the current web, and to verify the analysis done for the two sets of historical data. In the next few sections the methodologies that were employed to select the websites, capture the source code of the webpages, and web mining processes will be discussed.

[1] `http://www.archive.org`
[2] `http://www.alexa.com`

3.1 Methodology for Selecting Websites

As mention earlier, four sets of data were captured for this study. Two sets were used to look at a long term slice of websites for the past ten years, one was for the top twenty websites, and the other was a set of five hundred randomly selected websites. Another two larger sets were used to give an in-depth look of the current web, and to verify the analysis done for the two sets of historical data. For these two sets of of data, one was for the top five hundred websites, and the other was a set of five thousand randomly selected websites.

Our top twenty websites URLs were taken from the Alexa global top 500[3] on 13 June 2008, and our top five hundred websites URLs were taken from Alexa global top 500[3] on 24 July 2008. Since our Alexa top twenty websites look at the long term slice of websites for the past ten years, the archives of these websites were captured from Internet Archives[1] servers. Hence our top twenty websites were not the top twenty websites from Alexa global top 500[3] list, but the top twenty websites in that list with at least one archive from Internet Archive[1] servers between year 1999 and 2000. Our Alexa top five hundreds websites were taken directly from the same list as this set of data looks only at the current web. Thus our Alexa top five hundred websites data is a superset of our Alexa top twenty websites data as shown in equation 3.1.

$$Top20websites \subseteq Top500websites \tag{3.1}$$

The both sets of randomly selected websites were selected from Google Directory[4]. The Google Directory[4] uses the data from the Open Directory Project[5], but employs the Page Ranking algorithm [37] to rank the websites in each directory [35]. Depending on the number of targeted websites required, this amount will be divided into x number of websites and spread out as equally as possible across the fifteen directories at Google Directory[4]. In each directory, the most populated subdirectory will be chosen, and in each subdirectories, the first x number of websites will be selected. Due to the Page Ranking algorithm, the top x websites of that subdirectory will also be ensured to be quality websites from that subdirectory. Similarly, the random five thousand websites is a superset of the random five hundred websites as shown in equation 3.2.

[3] `http://www.alexa.com/site/ds/top_sites?ts_mode=global&lang=none`
[4] `http://www.google.com/dirhp`
[5] `http://www.dmoz.org`

$$Random500websites \subseteq Random5000websites \tag{3.2}$$

Additional effort were taken to ensure that when selecting our sets of random websites, these websites should not be part of our top websites list. This concept is illustrated in equation 3.3 where the *Randomwebsites* refers to our random 500 websites and random 5000 websites, and the *Topwebsites* refers to our Alexa Top 20 websites and Alexa Top 500 websites.

$$Randomwebsites \cup Topwebsites \tag{3.3}$$

3.2 Capturing the Webpages

A web robot was employed to carry out the capturing of information from the targeted websites in this study. The web robot was customized to fetch only the target webpage's source code along with the required external scripting and styling source code files. This information were stored on the local machine that was used to deploy the web robot. The suggested general modules of the web robot is presented in figure 3.1. First the selected domains were stored in the queue where it will be feed into a scheduler to manage the web robot process, and carefully not to overload the network traffic. Once the scheduler gives the go ahead, the URL will be fed to the downloader to proceed with its task. Finally the downloader will store the captured data on the machine's local hard disk.

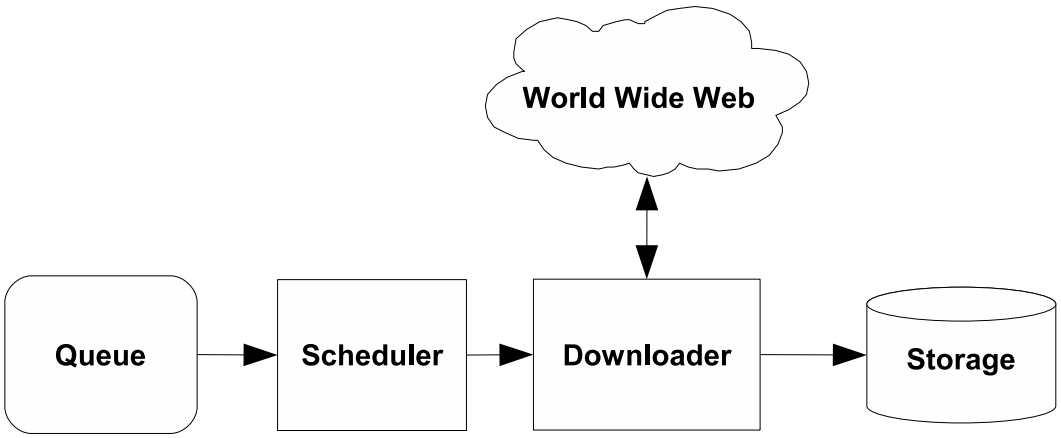

Figure 3.1: Modules in the web robot

When storing the downloaded data, each sets of data (i.e. Alexa Top 20 websites, random 500 websites, Alexa Top 500 websites and random 5000 websites) were stored in a separate folder. Each data file was given a unique file name as illustrated in figure 3.2 to avoid over writing of existing data files. The file name of the data file consist of three parts; the time stamp, the URL, and the extension (Ext). The time stamp is a combination of the date and time in the format of YYYYMMDDhhmmss when file was stored, and the URL is the actual data captured's URL with some of the string patterns replaced as seen in table 3.1. Finally the extension of a data file was either assigned, reassigned or a mirror from the original file's extension as shown in table 3.2.

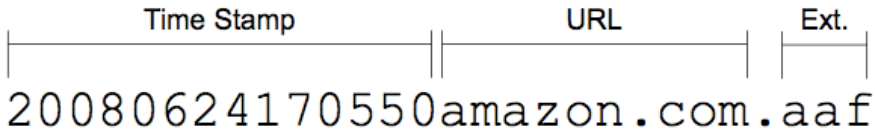

Figure 3.2: Structure of captured file name

String pattern	Replacement
http://	'blank'
http://www.	'blank'
http://web.archive.org/web	'blank'
'space'	'blank'
: , ; / ? %	'blank'
+ =	-
&	N

Table 3.1: Captured data files URL characters replacement

Actual file extension	Assigned file extension
Original webpage	.aaf
External JavaScript file	used given (e.g. .js)
External CSS file	used given (e.g. .css)

Table 3.2: Assigned file extension for captured files

After the targeted web pages were captured, the next part will involve a process called web mining. This process involves retrieval of the required data

from the captured webpages for further analysis. In the next section this will be covered in greater depth.

3.3 Web Mining

Web mining is a stage that involves finding for the interesting information required from the captured webpages for further analysis. Each webpage and its required external files were either parsed using the PHP native Document Object Model (DOM) HTML parser, or searched using the Perl's regular expression syntax to retrieve the required information. Now let us examine how the different types of analysis and data were collected.

3.3.1 Types of HTML Standard Detection

Two methods were employed to detect the type of HTML standards used by a webpage. (1) Whenever available, the document type (DOCTYPE) was detected using the following Perl's regular expression syntax. Commonly the DOCTYPE will be printed at the top of the webpage's source code. Three examples listed below are some examples of the possible methods used by a webpage to declare the document's intended HTML standard. The first example given was used to declare a HTML 3.2 Final webpage, the second example shows the DOCTYPE for HTML 4.01 Transitional webpage, and the third example is the DOCTYPE for XHTML 1.0 Strict webpage.

Example 1:

```
<!DOCTYPE HTML PUBLIC "-//W3C//DTD HTML 3.2 Final//EN">
```

Example 2:

```
<!DOCTYPE HTML PUBLIC "-//W3C//DTD HTML 4.01 Transitional
    //EN" "http://www.w3.org/TR/1999/REC-html401-19991224/
    loose.dtd">
```

Example 3:

```
<!DOCTYPE html PUBLIC "-//W3C//DTD XHTML 1.0 Strict//EN"
    "http://www.w3.org/TR/xhtml1/DTD/xhtml1-strict.dtd">
```

From the above examples some similarities in declaring the DOCTYPE of a webpage can be observed, hence the following syntax was devised and used to capture the DOCTYPE from a webpage source code. This syntax will not only return the version of the HTML standard, but also the document's sub-specification (Transitional, Strict...).

```
/!DOCTYPE\sHTML\sPUBLIC\s\"-\/\/\w+\/\/DTD\s([\sa-z0-9\.]+)
    \/\/EN\"/i
```

Since this method (declaring the DOCTYPE) was not strongly enforced by most major user-agents, hence HTML documents can be written without it and still being parsed properly by them. Therefore another method was used whenever the DOCTYPE was not available. (2) This time the HTML document were parsed, and the type of tags used were examined to determine the type of HTML standard used. However as it was not possible to know what was the actual HTML standard the web author/developer was intending to use, and since some of our documents were historical data, we will assume that if a document uses HTML syntax that is so simple, and it was not possible to differential between the different versions of HTML standards, then we will assume it to be a HTML 2 document. So by default we will assume a webpage to be a HTML 2 document, unless a different standard was detected. The HTML elements that were used to defined for the different HTML standards in this project were listed in table 3.3.

3.3.2 WCAG 1.0 Conformance Detection

To check if a webpage is compliant to the WAI's WCAG 1.0 guidelines, we will search for a display of conformance to these guidelines on a webpage. For this process, since it was not intended to create a validation tool and due to the scope of this study, thus the above methodology was employed. A couple of techniques were used for this detection, the first technique looks for any display of WCAG 1.0 conformance logo, and the type of logo for the specific guideline level of

HTML Standards	Elements
HTML 3 [20]	listing, plaintext, style, xmp
HTML 3.2 [40]	font, div, script
HTML 4.0, HTML 4.01	abbr, acronym, applet, basefont, bdo, button, col, colgroup, center, del, embed, fieldset, frame, frameset, iframe, ins, label, legend, noframes, noscript, object, q, s, span, tbody, tfoot, thead, u
XHTML 1.0	-
XHTML 1.1	rb, rbc, rp, rt, rtc, ruby

Table 3.3: HTML standards and elements used during our web mining process for HTML standards detection [39, 32, 21]

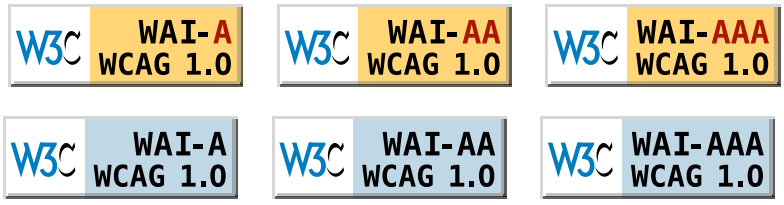

Figure 3.3: WCAG 1.0 Conformance Logos

conformance supplied by W3C. The W3C provides the logos in two colours, the original colour and a blue colour as seen in figure 3.3.

The respective logo's level is usually displayed if a page is conform to a certain level of conformance in the WCAG 1.0 guidelines. Presented below are the sample HTML codes provided by the W3C for the display of the above logos as a proof of conformance on a webpage.

Level A Conformance:

```
<a href="http://www.w3.org/WAI/WCAG1A-Conformance"
      title="Explanation of Level A Conformance">
  <img height="32" width="88"
          src="http://www.w3.org/WAI/wcag1A"
          alt="Level A conformance icon,
          W3C-WAI Web Content Accessibility Guidelines 1.0"></a>
```

Level Double-A Conformance:

```
<a href="http://www.w3.org/WAI/WCAG1AA-Conformance"
```

```
        title="Explanation of Level Double-A Conformance">

    <img height="32" width="88"
            src="http://www.w3.org/WAI/wcag1AA"
            alt="Level Double-A conformance icon,
            W3C-WAI Web Content Accessibility Guidelines 1.0"></a>
```

Level Triple-A Conformance:

```
    <a href="http://www.w3.org/WAI/WCAG1AAA-Conformance"
          title="Explanation of Level Triple-A Conformance">
      <img height="32" width="88"
            src="http://www.w3.org/WAI/wcag1AAA"
            alt="Level Triple-A conformance icon,
            W3C-WAI Web Content Accessibility Guidelines 1.0"></a>
```

The above codes gave an illustration of what was expected within a HTML code that display these conformance logos, however web developers/authors may change these codes slightly to meet their design specifications. Therefore two checks were used to detect the display of the conformance logos that were generic to the two logo colours, and for more flexibility to the codes used. The following regular expressions were used to detect for the presence of WCAG 1.0 level A conformance logo.

```
(1)  /href\s*=\s*[\'\"]*http\:\/\/www.w3.org\/WAI\/WCAG1A-
        Conformance[\'\"]*/i
(2)  /src\s*=\s*[\'\"]*http\:\/\/www.w3.org\/WAI\/wcag1A/i
```

The next two regular expressions were used to detect for the presence of WCAG 1.0 level AA conformance logo.

```
(1)  /href\s*=\s*[\'\"]*http\:\/\/www.w3.org\/WAI\/WCAG1AA-
        Conformance[\'\"]*/i
(2)  /src\s*=\s*[\'\"]*http\:\/\/www.w3.org\/WAI\/wcag1AA/i
```

The following regular expressions were used to detect for the presence of WCAG 1.0 level AAA conformance logo.

```
(1)  /href\s*=\s*[\'\"]*http\:\/\/www.w3.org\/WAI\/WCAG1AAA-
         Conformance[\'\"]*/i
(2)  /src\s*=\s*[\'\"]*http\:\/\/www.w3.org\/WAI\/wcag1AAA/i
```

Some websites may choose not to display the conformance logo, but display their web content accessibility conformance in plain text. Thus another technique was used to cope with this issue. Two methods were configured for this experiment. This first method was a pessimistic view that scans for the last 100 characters on each webpage for a display of conformance, and the second method was an optimistic view that scans the entire webpage for a display of conformance. Words such as 'accessibility' or 'WCAG', or just for the display of level of conformance were checked. When checking for the level of conformance, except for level A conformance, both level double A and level triple A conformance were conducted. This was because checking for level 'A' conformance alone, it can be easily mixed up with the letter A. The following regular expression was used to check for the display of text for the accessibility conformance.

```
/(Accessibility[-_\s,]*|wcag(\s?1\.0)?[-_\s,]?|\s(AA|AAA)\s)/i
```

From the two techniques presented, different forms of results will be gathered when searching for the accessibility conformance of a webpage. Thus the final results used for this analysis will be in the form of either the webpage was conformed or was it not to the accessibility guidelines.

3.3.3 Graphical Format Usage Detection

In this part of the thesis, the detection for different types of graphical format will be covered in detail. There are many ways a graphic can be formatted and prepared to be portable over the web. Only the more popular formats, and the formats suggested by W3C will be covered in this study. To detect the usage of the different types of graphical formats, the extension of the different formats must be include in the regular expressions used. However for some graphical formats multiple file extensions may exist. So lets first look at the different types of graphical formats we will be covering in this study, and its possible file extension(s) used for portability in table 3.4.

Now that the foreseeable different types of file extensions for each graphical format was identified, now lets look at the few ways that a web developer/author

Graphical Formats	File Extensions
GIF	.gif
JPEG	.jpg, .jpeg, .jpe
PNG	.png
Flash	.swf, .flv, .f4v, .f4p, .f4a, .f4b
SVG	.svg
SMIL	.smil

Table 3.4: Graphical formats and file extensions

can include a graphic to a webpage. The generic method include either attaching the graphic within an HTML code or in a style coding as a background. Hence from the webpage's source code, the following regular expressions can be used to detect a graphical format for the two generic method and the file's extension. First for GIF graphical formats,

```
(1)  /[\'\"]*([-_~\:a-z0-9\/\.]+\.gif)[\'\"]*/i
(2)  /\((\s*[-_~\:a-z0-9\/\.]+\.gif\s*)\)/i
```

For JPEG and its possible file extensions,

```
(1)  /[\'\"]*([-_~\:a-z0-9\/\.]+\.jpg)[\'\"]*/i
(2)  /\((\s*[-_~\:a-z0-9\/\.]+\.jpg\s*)\)/i
(1)  /[\'\"]*([-_~\:a-z0-9\/\.]+\.jpeg)[\'\"]*/i
(2)  /\((\s*[-_~\:a-z0-9\/\.]+\.jpeg\s*)\)/i
(1)  /[\'\"]*([-_~\:a-z0-9\/\.]+\.jpe)[\'\"]*/i
(2)  /\((\s*[-_~\:a-z0-9\/\.]+\.jpe\s*)\)/i
```

For PNG formats,

```
(1)  /[\'\"]*([-_~\:a-z0-9\/\.]+\.png)[\'\"]*/i
(2)  /\((\s*[-_~\:a-z0-9\/\.]+\.png\s*)\)/i
```

For Flash and its possible file extensions,

```
(1)  /[\'\"]*([-_~\:a-z0-9\/\.]+\.swf)[\'\"]*/i
(2)  /\((\s*[-_~\:a-z0-9\/\.]+\.swf\s*)\)/i
(1)  /[\'\"]*([-_~\:a-z0-9\/\.]+\.flv)[\'\"]*/i
(2)  /\((\s*[-_~\:a-z0-9\/\.]+\.flv\s*)\)/i
(1)  /[\'\"]*([-_~\:a-z0-9\/\.]+\.f4v)[\'\"]*/i
```

```
(2)  /\((\s*[-_~\:a-z0-9\/\.]+\.f4v\s*)\)/i
(1)  /[\'\"]*([-_~\:a-z0-9\/\.]+\.f4p)[\'\"]*/i
(2)  /\((\s*[-_~\:a-z0-9\/\.]+\.f4p\s*)\)/i
(1)  /[\'\"]*([-_~\:a-z0-9\/\.]+\.f4a)[\'\"]*/i
(2)  /\((\s*[-_~\:a-z0-9\/\.]+\.f4a\s*)\)/i
(1)  /[\'\"]*([-_~\:a-z0-9\/\.]+\.f4b)[\'\"]*/i
(2)  /\((\s*[-_~\:a-z0-9\/\.]+\.f4b\s*)\)/i
```

For SVG,

```
(1)  /[\'\"]*([-_~\:a-z0-9\/\.]+\.svg)[\'\"]*/i
(2)  /\((\s*[-_~\:a-z0-9\/\.]+\.svg\s*)\)/i
```

Finally for SMIL,

```
(1)  /[\'\"]*([-_~\:a-z0-9\/\.]+\.smil)[\'\"]*/i
(2)  /\((\s*[-_~\:a-z0-9\/\.]+\.smil\s*)\)/i
```

If any of the above regular expressions was true, the presences of the respective graphical format would be assumed to be used on the webpage. The next subsection discusses about the method used to detect the usage of styling and client-side scripting on a webpage.

3.3.4 Client-side Scripting and Styling Usage Detection

Detecting the use of client-side scripting and styling will allow trends for the respective standards to be identified. For JavaScript, this will help to understand the usage of it and how it has effected the growth or decline in popularity of other standards and recommendations (E.g. AJAX...). To detect the usage of client-side scripting, the HTML codes will be parsed and the "<script>" tag will be examined. Under the "script" elements, both the "type" and "language" attributes will be searched using the regular expression "/javascript/i" for the existence of JavaScript. If this exist, or none of these attributes were defined, then an assumption will be made that the client-side scripting language was JavaScript. However if at least one of these attributes was defined and JavaScript was not found, then VBScript will be assumed.

Three methods were employed to conduct the detection of CSS. (1) Detect within the HTML code if the element "style" is used. (2) All the elements defined

in a HTML document will be checked if the attribute "style" was used. This method was employed to detect if CSS was applied to the individual HTML tags for specific display control. (3) The next method checks for the attachment of external CSS files by using the following regular expression.

```
/<link\s[-_~\/\.\:a-z0-9\s]*rel\s?=\s?[\'\"]?stylesheet[\'\"]
    ?[-_~\/\.\:a-z0-9\s]*type\s?=\s?[\'\"]?text/css[\'\"]?/i
```

As long as one of the above methods was detected, an assumption will be made that the presence of CSS usage exist. Now lets look at a related model that uses these standards; AJAX, and the methods we have applied to detect for its usage.

3.3.5 AJAX Usage Detection

AJAX growing popularity may be benefited from the fruits of other web technologies such as JavaScript. Thus analysing the usage of AJAX will help us to understand its usage trend and how it was affecting the other related web technologies. On the client-side, AJAX uses JavaScript and the asynchronous technology to communicate with the server. Although there are numerous ways in which one can determine the presence of AJAX, our method employed consist of the following two techniques. (1) The detection of "XMLHttpRequest" in JavaScript, and (2) the usage of HTML element "iframe". The detection of HTML element "iframe" was used because this element uses the asynchronous technology, and can be dealt with by JavaScript. Once the external JavaScript file's codes were concatenated with the JavaScript codes embedded within the HTML code, the following regular expressions can be applied to search for the "XMLHttpRequest" within them.

```
/XMLHttpRequest\(/i
```

To search for the "iframe" element, the following regular expression was used; although parsing the HTML code will do the job as well.

```
/<iframe\s/i
```

As long as one of the above method was detected on a webpage, an assumption would be made that the presence of AJAX exist.

3.4 Overall Process

The processes and methods required to collect the necessary data can be a fatigue due to the volume of websites analysed in this study. This overview wraps up the overall processes that is required to go through when collecting each set of data. Four general stages will be required to complete the entire process for each set of data. As shown in figure 3.4, in the first stage the web robot will be sent out to extract or select the targeted URLs to be captured. Then in the second stage the web robot will be deployed to capture the source code of the targeted websites along with the necessary external files. In the third stage, this consist of two parts: an automated, and a manual verification of data process to ensure they were captured correctly. If an error was detected, stage two and three will be required to be repeated before one can proceed to the final stage. The purpose of last stage is to collect the required information from the captured source code so that further analysis can be done. None of the stages in figure 3.4 should be bypassed to ensure repeatability. To ensure integrity of the data captured, stage three should be done thoroughly.

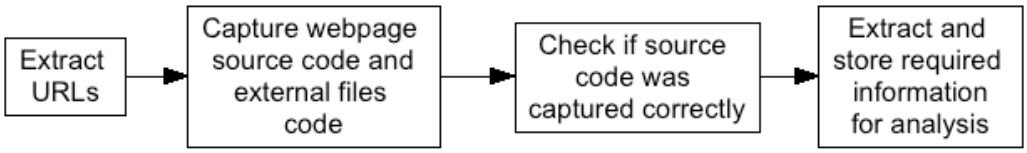

Figure 3.4: Overall Process Flow

3.5 Conclusion

Different methods were applied in this study to understand the evolution of the web. Four major stages were used to successfully capture the selected websites, and to extract the required data for analysis. From the discussion above, due to the existence of some missing data from the Internet Archives, the historical data sets had to be normalised due to the inconsistent volume of data captured between the intervals. Percentage was later applied to all the data captured for normalisation before analysis were conducted. Besides the dip in historical data available from the Internet Archives between July 1999 and January 2000

as shown in the next chapter on table 4.1, most of the data retrievable were of acceptable volume. Finally regular expressions and parsing the HTML codes were techniques used together with our methodologies to extract the necessary data for our analysis required by this study.

Chapter 4

Results and Discussion

Using the methodologies discussed in chapter 3, our web robot was deployed to capture the selected websites between 24 June 2008 and 24 July 2008. The capturing of webpages process, or stage two and three as referred to in section 3.4 took longer then what was predicted initially. This was due to the missing data from the Internet Archive[1] and the no longer existing websites that were chosen. Due to these issues some of the current websites data and archives were not retrievable. Hence these missing data causes an inconsistent volume of data captured for the different sets of websites and the historical data. Thus all the data collected had to be normalised before analysing them.

In this chapter, possible analysis, trends and conclusions from our results were presented. The presentation of our discussions will be structured into the four categories discussed earlier in chapter 2, followed by the further analysis done to enhance our understanding for some trends. Before analysing the results we had collected, lets cover the issues relating to the data collected and the results from our web mining processes.

4.1 Issues Relating to Captured Data

The data captured for this study were not all perfect. Some of the historical data were not available from Internet Archive's[1] servers, or the websites selected that were no longer existent. Due to this, our historical data sets require some form of normalisation after the data was captured. For this study, percentage was applied to normalised the captured data for further analysis. Hence for all

[1] `http://www.archive.org`

the analysis conducted in this study the data analysed were normalised data and not the raw data.

Table 4.1 list the total number of websites retrievable for each sets of data during our capturing process. One would noticed that besides the historical data collections, there were also missing data from Alexa top five hundred websites where we attempted to capture the current web. The missing data in our Alexa top five hundred websites data was due to four no longer existing URLs provided by Alexa global top five hundred[2] on the 24 July 2008.

Year	Alexa top 20 websites	Alexa top 500 websites	Random 500 websites	Random 5000 websites
Jan 1999	17	-	320	-
Jul 1999	15	-	76	-
Jan 2000	14	-	176	-
Jul 2000	19	-	452	-
Jan 2001	15	-	444	-
Jul 2001	18	-	380	-
Jan 2002	18	-	380	-
Jul 2002	17	-	452	-
Jan 2003	20	-	479	-
Jul 2003	20	-	459	-
Jan 2004	18	-	409	-
Jul 2004	20	-	483	-
Jan 2005	20	-	487	-
Jul 2005	20	-	463	-
Jan 2006	20	-	478	-
Jul 2006	20	-	473	-
Jan 2007	20	-	467	-
Jul 2007	16	-	440	-
Jun 2008	20	496	500	5000

Table 4.1: Total number of websites obtainable during capturing process

When retrieving archives from Internet Archives, it was noticed that between July 1999 and February 2000, very little archives were available. As seen in table 4.1 very few data were captured between these times for both our data for Alexa top twenty websites and random five hundred websites. Could this be due to the Millennium bug [6] or may be it was just a mistake at Internet Archives?

[2] `http://www.alexa.com/site/ds/top_sites?ts_mode=global&lang=none`

To deal with the inconsistency of the data capture, some form of normalisation was required to be applied to these data before further analysis were done. In this study, percentage was applied to all the data before conducting any further analysis. Thus all the results from the analysis presented in this chapter are in the form of percentages.

4.2 W3C Standards

Since 1994 the W3C has been constantly revising and introducing new recommendations for the web. Thus before discussing about the results from the analysis we had conducted, let us first look at the milestone of the web standards. Figure 4.1 presents the timeline of the major HTML and CSS standards when they became a web standard.

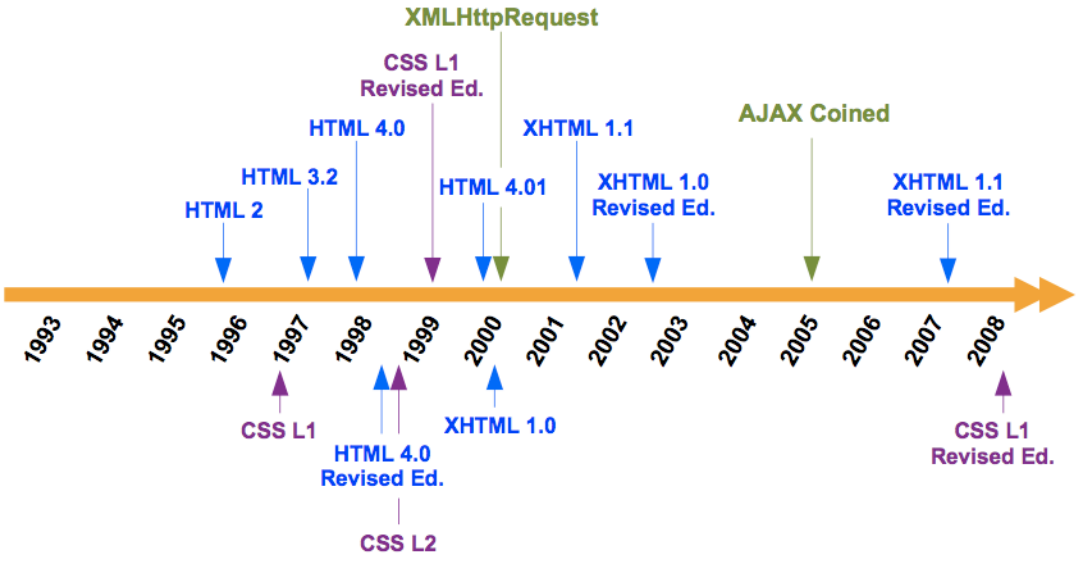

Figure 4.1: Web standards milestone

The information collected for the styling of web content analysis was done for CSS as in general to look at the general CSS usage. Figure 4.2 shows the results in percentage for the usage of CSS for all the four sets of data collected. A steady growth was noticed and we predict that this growth will continue for the next year. When analysing the trend for the usage of CSS to be more than

50%, one can notice from Figure 4.1 and figure 4.2 that it took the Alexa top twenty websites about four years to achieve to it, and the random 500 websites about nine years for it to cross the more than 50% mark. Although both showed similar trends, a four years lag was noticed between the top websites and the random websites. Hence for this type recommendations to get adopted by more than 50% of the web, an adoption time of about four to nine years is required, and less than four years for web technologies that surrounds it to get developed. When using Pearson correlation, a significant relationship was noticed between the Alexa top twenty websites and the random five hundred websites, $r = .89$, p (two-tailed) $< .01$.

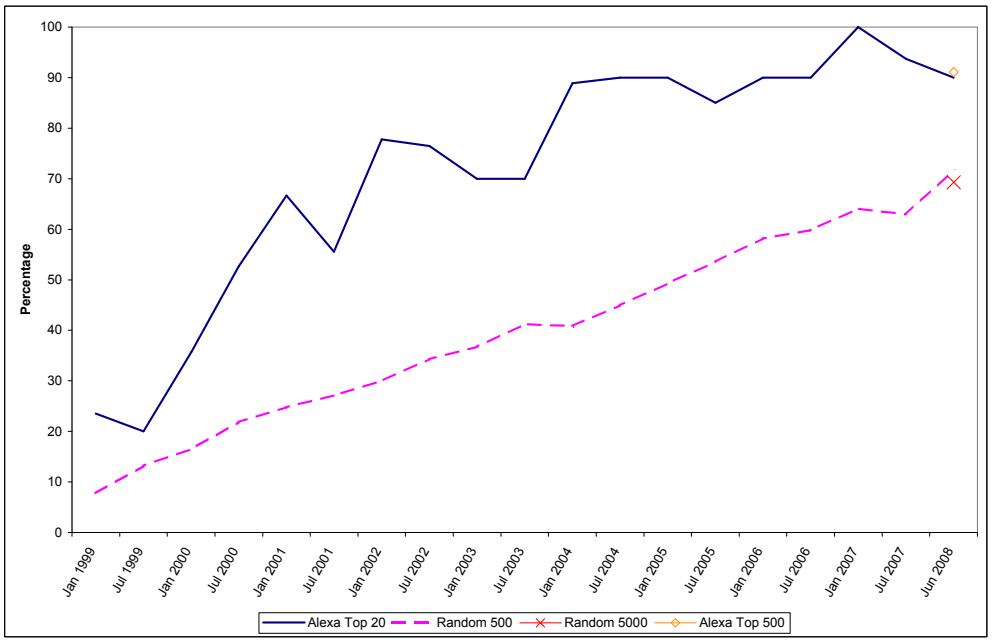

Figure 4.2: CSS usage percentage for Alexa top 20 and 500 websites, and a set of 500 and 5000 random websites

Styling of plain text can control the way how a webpage is presented in a user-agent, however the data structure of the plain text is equivalently important to realised its full capability. As discussed earlier, there are a number of W3C recommendations for HTML standards, and these standards timeline were also presented in figure 4.1. The individual standards will be discussed first before covering the further analysis surrounding these standards.

A gradual decline in HTML 2 usage for the last ten years was noticed from the graph in figure 4.3. This was seen for both the random five hundred websites

and the Alexa top twenty websites sets of data. Further analysis was done to determine if these two sets of data have any correlation applying Pearson correlation. A significant relationship between Alexa top twenty websites and random five hundred websites was noticed, $r = .52$, p (two-tailed) $< .05$. Both the random five thousand websites and the Alexa top five hundred websites results where close for the current web analysis. Thus a similar usage for this standard was forecasted.

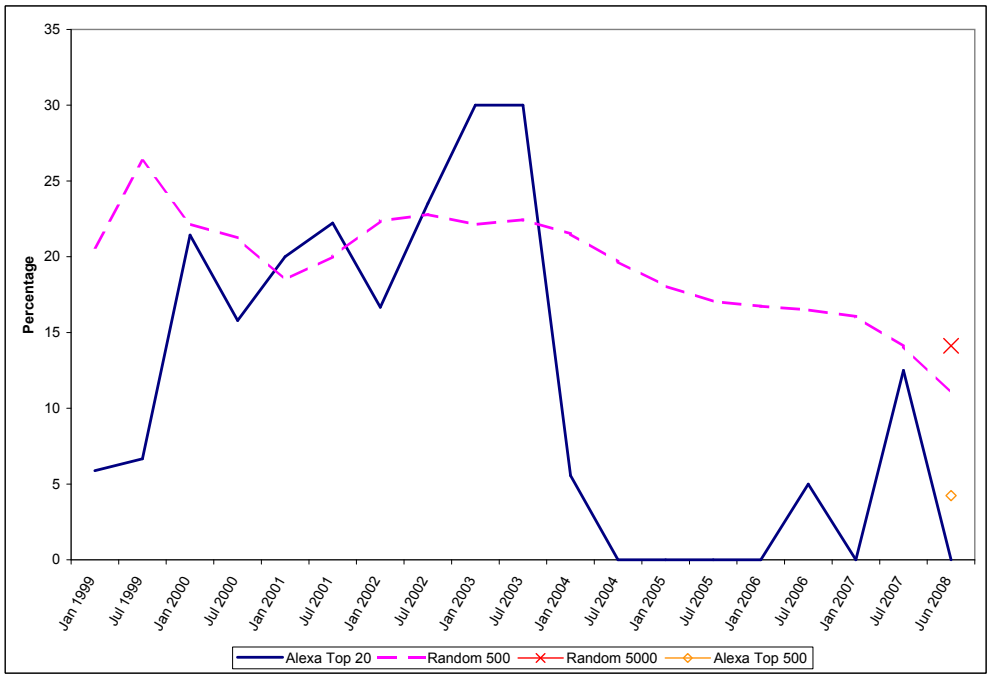

Figure 4.3: HTML 2 usage percentage for Alexa top 20 and 500 websites, and a set of 500 and 5000 random websites

Similar to HTML 2 standards, the HTML 3 standards usage also exhibited a decline for the last ten years. Figure 4.4 showed that a correlation between both the the random five hundred websites results and Alexa top twenty websites results. When applying Pearson correlation, a significant relationship was noticed between them, $r = .68$, p (two-tailed) $< .01$. From this experiment, both the random five thousand websites and Alexa top five hundred websites demonstrated very close percentages for the current web. This proves that HTML 3 is slowly loosing its popularity with web developers/authors.

The HTML 4 standards has been heavily used by the web as shown in figure 4.5, but a declining trend is predicted. A significant relationship between the

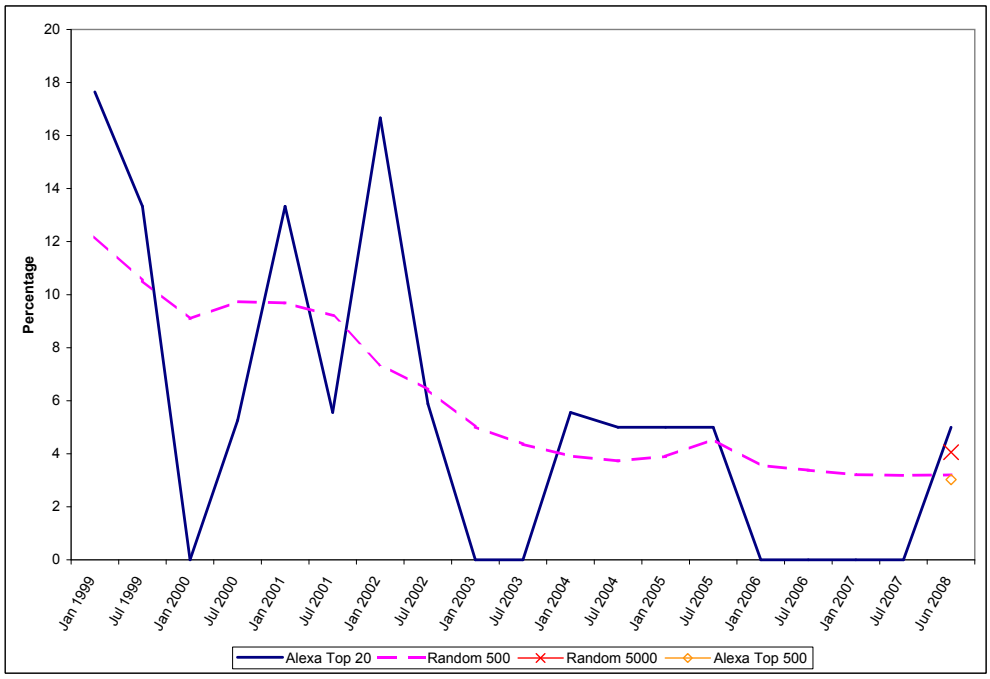

Figure 4.4: HTML 3 usage percentage for Alexa top 20 and 500 websites, and a set of 500 and 5000 random websites

random five hundred websites results and the Alexa top twenty websites results was noticed when applying Pearson correlation, $r = .62$, p (two-tailed) $< .01$. Our prediction for this trend was justified using both the Alexa top five hundred websites results and the random five thousand websites results presented.

Since the release of the revised edition for the XHTML 1.0 standard in August 2002, an increase in usage for this standard was noticed around 2004 as shown in figure 4.6. There was also a significant relationship between the random five hundred websites and the Alexa top twenty websites results when Pearson correlation was applied, $r = .92$, p (two-tailed) $< .01$. From the graphs, based on the verification of the Alexa top five hundred websites and the random five thousand websites results, a growing trend was predicted to continue.

The recent release of the revised edition of the XHTML 1.1 standards in February 2007 as seen in figure 4.1 explains the reasons behind these poor adoption rates for this standard during the time when this study was conducted. From our data collected as presented in figure 4.7, and from our adoption trends of the previous HTML standards, a growth in usage for this standard was predicted. This prediction cannot be validated, and future work for this standard is required, and

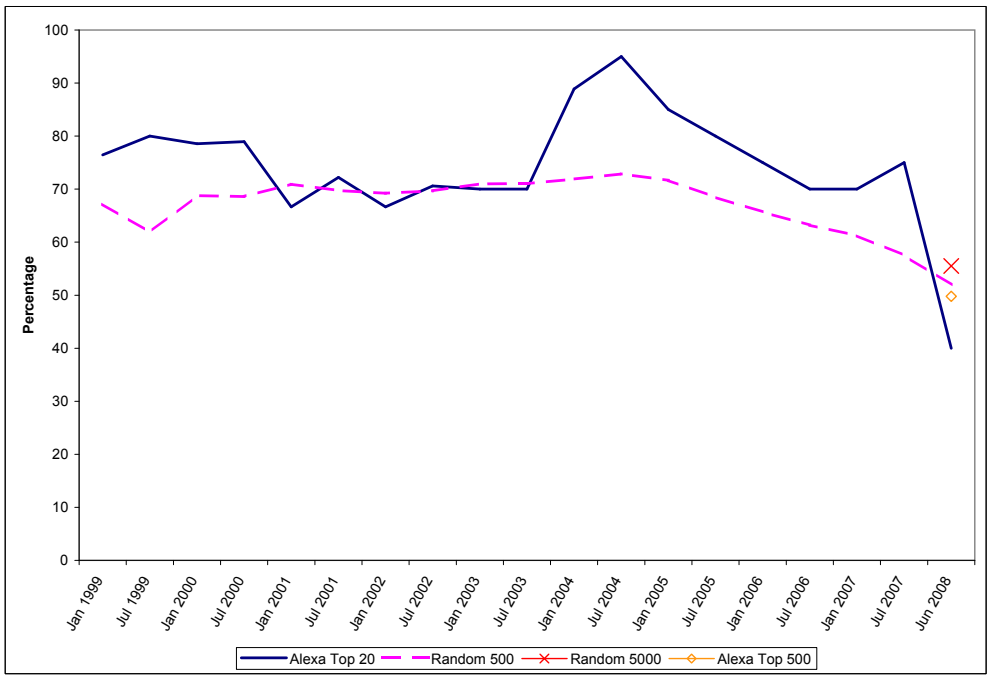

Figure 4.5: HTML 4 usage percentage for Alexa top 20 and 500 websites, and a set of 500 and 5000 random websites

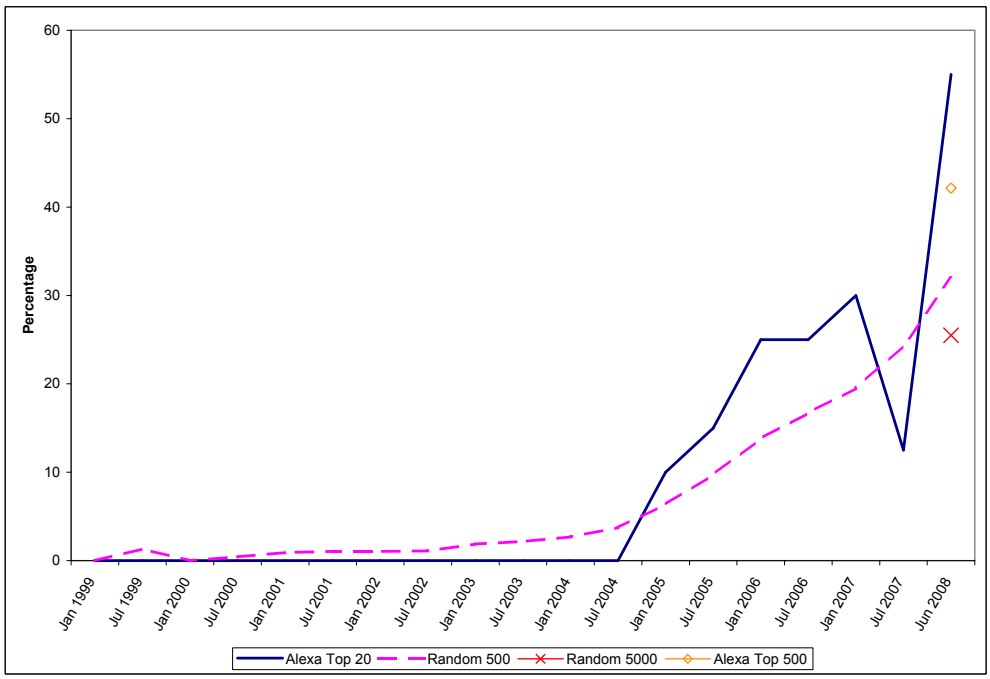

Figure 4.6: XHTML 1.0 usage percentage for Alexa top 20 and 500 websites, and a set of 500 and 5000 random websites

to prove if our prediction was correct.

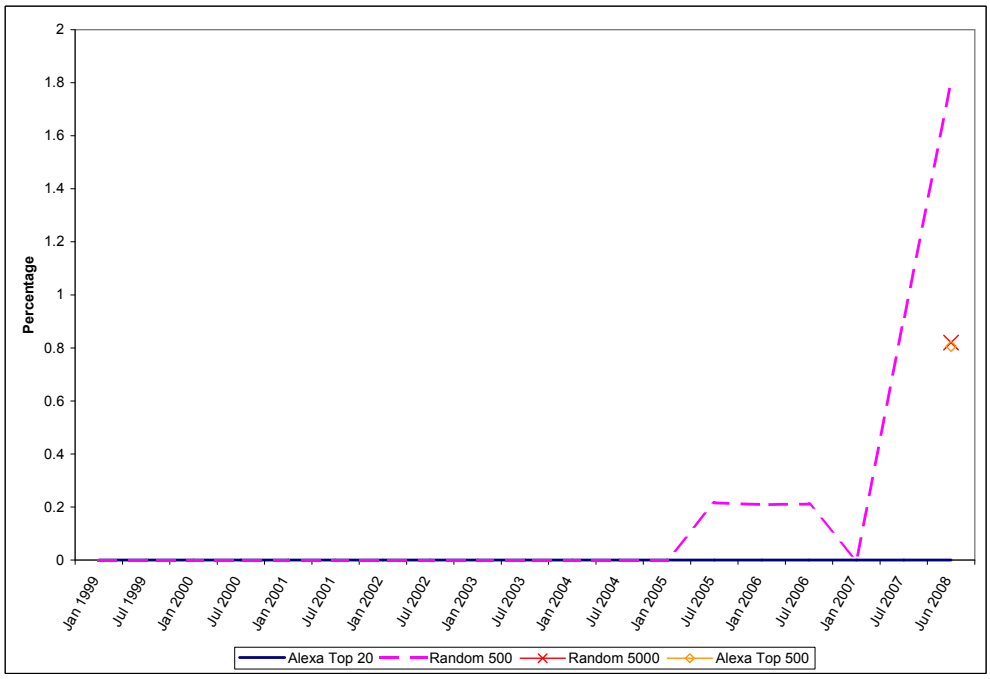

Figure 4.7: XHTML 1.1 usage percentage for Alexa top 20 and 500 websites, and a set of 500 and 5000 random websites

From the above analysis, besides the XHTML 1.1 standards, all the rest of the major W3C standards demonstrated a significant correlation between the Alexa top websites and the random websites results, p (two-tailed) $< .05$. Therefore we can conclude that from these results the Alexa top websites do give a good idea of how the web is evolving for the W3C standards in general.

Further analysis were done to understand the relationships between the W3C standards for the random five hundred websites, and the Alexa top twenty websites. Figure 4.8 shows the usage of HTML 2, 3 and 4 for Alexa top twenty websites over the past ten years. Both HTML 2 and 3 exhibits a gradual decline in usage, while HTML 4 was increasing before the year 2006, and from year 2007 a rapid roll off was noticed. One can expect for this type of trend to happen to the existing HTML standards when a new HTML standard is gaining its popularity; since only one HTML standard can dominate the web. Hence further analysis on the different types of W3C standards is required to understand the reason behind this.

The graph in figure 4.9 for the random five hundred websites for the last ten

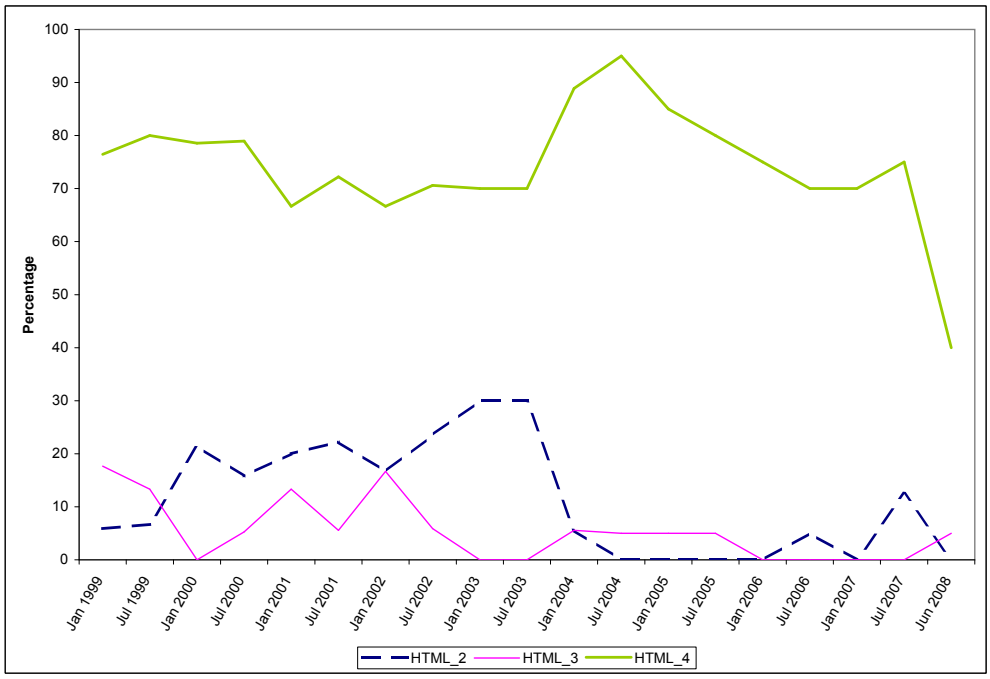

Figure 4.8: HTML 2, 3, 4 usage percentage for Alexa top 20

years also showed trends similar the Alexa top twenty websites results. Again from the discussions above, a decline in usage for these standards were expected. It was also observed from figure 4.8 and figure 4.9 that the influencing factor for their decline were not from either the HTML 2, 3 or 4, but by some other standards or recommendations.

Next lets examine how will HTML 4 perform when plotted against XHTML 1.0 and 1.1. From figure 4.10 it was observed that around the same time when HTML 4 began to roll off, a significant increase in the usage of XHTML 1.0 was also notice. The pattern of the declining HTML standard; HTML 4, exhibits a mirroring image of the XHTML 1.0's graph. Thus for the top websites, it was observed that a major shift in usage ($> 50\%$) from HTML 4 to XHTML 1.0 had already occurred. This converting trend is expected to continue as it can be verified by the discussion presented earlier for the individual HTML standards trends.

Finally in the last analysis, HTML 4 was again plotted against XHTML 1.0 and 1.1 for the random five hundred websites results. This was to check if the same trends exist for the random websites results. From figure 4.11 showed that

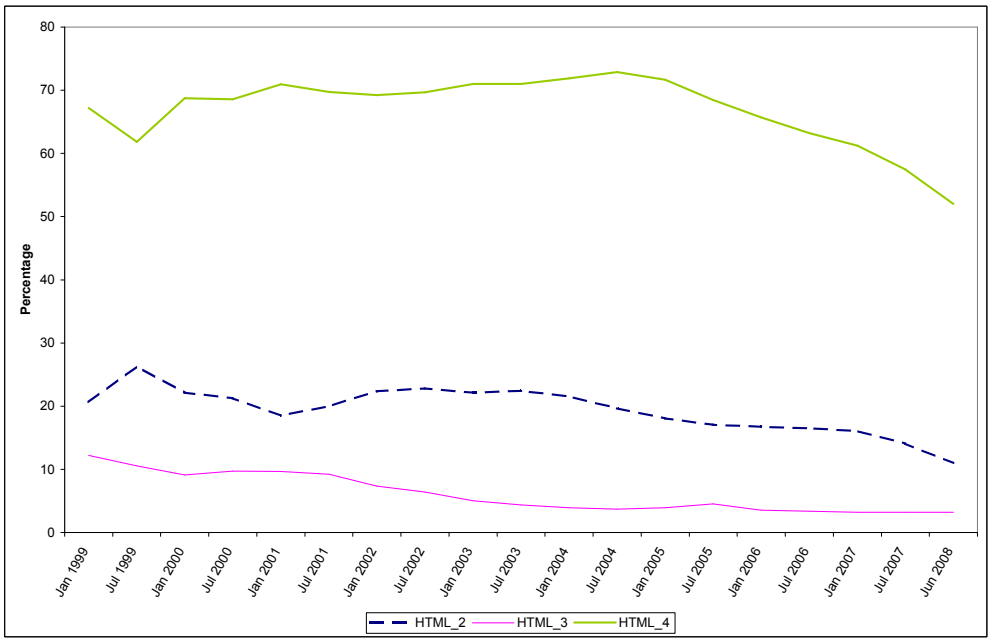

Figure 4.9: HTML 2, 3, 4 usage percentage for random 500

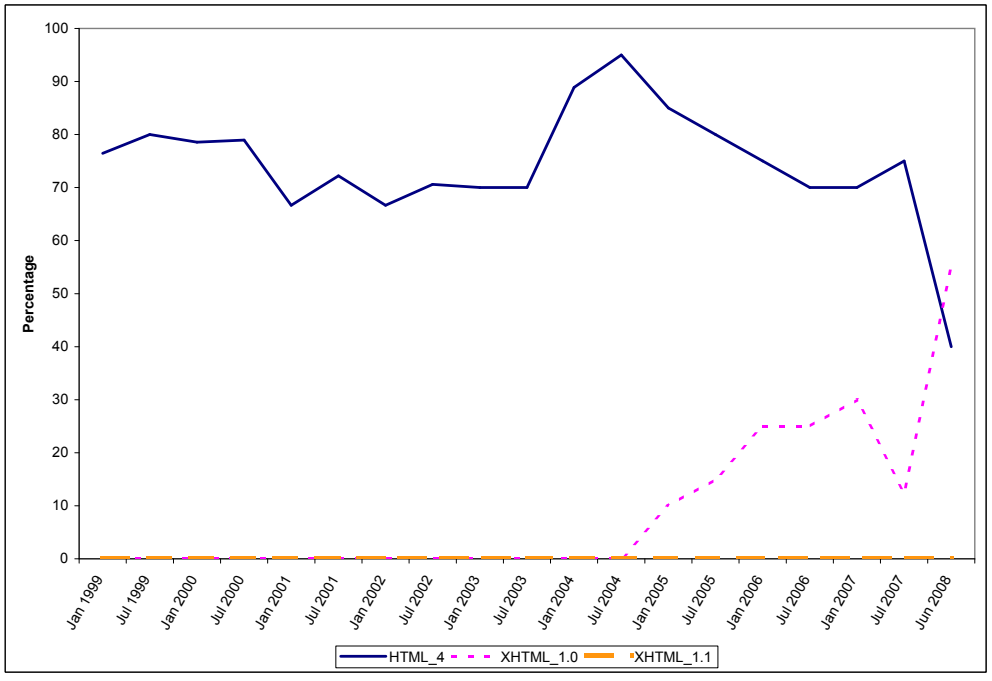

Figure 4.10: HTML 4, XHTML 1.0 and 1.1 usage percentage for Alexa top 20

this set of results also exhibited similar trends, as one would expected this from their individual standards analysis discussed earlier. Although the random five hundred websites have not been adopted as much as the Alexa top twenty websites to the XHTML 1.0 standards, but they were showing similar trends to the Alexa top twenty websites W3C standards.

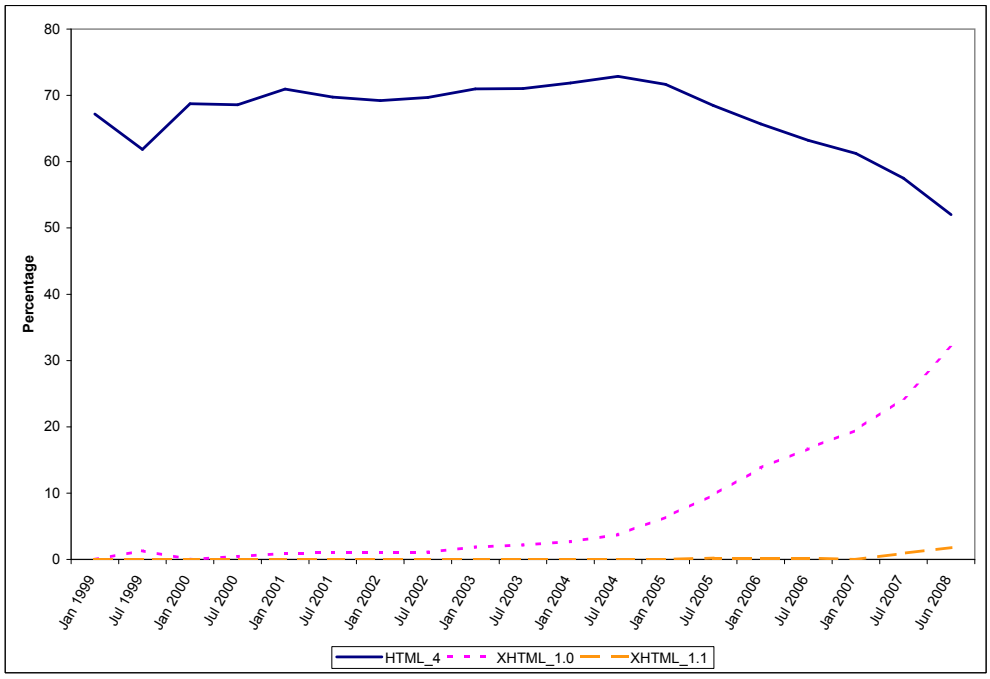

Figure 4.11: HTML 4, XHTML 1.0 and 1.1 usage percentage for random 500

To summarise the discussions presented in this section, the major W3C standards on average seems to be led by the Alexa top websites. From the discussions presented earlier, the Alexa top websites in-general does give a good representation of how the web is evolving for the major W3C standards. On average the Alexa top websites adopts to a new standard one year faster than the random websites, and a growth in XHTML 1.0, and 1.1 usage was predicted. A trend was also notice that the usage of HTML 4 is decreasing, and websites that were previously using it are replacing their webpages with XHTML 1.0.

4.3 Graphical Formats Results

There are many different types of graphical formats available to be used over the web, however as mentioned earlier, only JPEG, GIF, SVG, PNG, SMIL and Flash formats will be covered. The results collected for the individual graphical format will be discussed first, followed by the further analysis for these formats. To begin lets look the results from our analysis conducted on GIF.

Figure 4.12 shows the results for the usage of GIF for our four sets of websites. Although both the random five hundred websites and the Alexa top twenty websites results displayed a gradual take up trend, but no significant correlation was found between the two sets of results when Pearson correlation was applied. Using the Alexa top five hundred websites and the random five thousand websites results for verification, the usage of GIF was expected to remain unchanged for the near future.

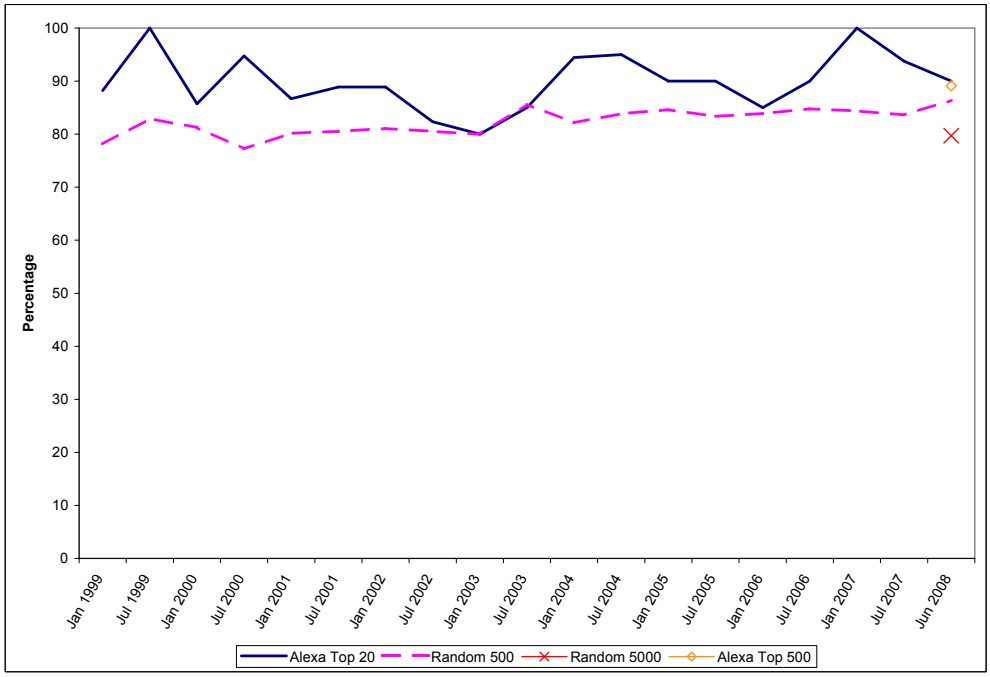

Figure 4.12: GIFs usage percentage for Alexa top 20 and 500 websites, and a set of 500 and 5000 random websites

In figure 4.13 shows a healthy growth in usage trend for JPEG in the past ten years. This trend was predicted to continue as verified by the Alexa top five websites and the random five thousand websites results. Although JPEG has

been implemented by web browsers since 1996, but it took three years for more than fifty percent of the Alexa top twenty websites to have an adopt it. The lag between the Alexa top twenty websites results and the random five hundred websites results were quite small, it took only around one year later for more than fifty percent of the random five hundred websites use the JPEG graphical format. To analyse if a correlation exist between the top websites and the random websites, Pearson correlation was applied. There was a significant relationship between the Alexa top twenty websites and the random five hundred websites results, $r = .67$, p (two-tailed) $< .01$.

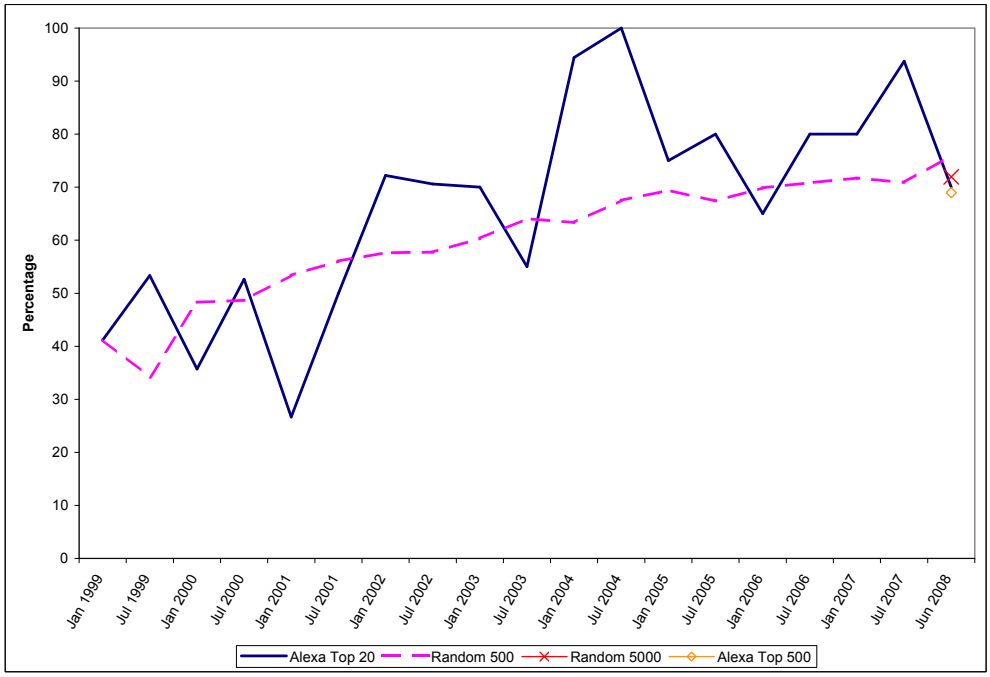

Figure 4.13: JPEGs usage percentage for Alexa top 20 and 500 websites, and a set of 500 and 5000 random websites

PNG became a W3C recommendation since October 1996, and the second edition was released in November 2003. As shown in figure 4.14, the random five hundred websites initial take up this graphical format much earlier then the Alexa top twenty website. However after the release of the second edition, the Alexa top twenty websites quickly pick up, and led the adoption trend for this graphical format. When Pearson correlation was applied to the two sets of data, a significant relationship between the Alexa top twenty websites and the random five hundred websites results was observed, $r = .93$, p (two-tailed) $< .01$. Hence

from this experiment, it can be seen that both the Alexa top twenty websites and the random five hundred websites learned from each others usage trends. From the results of the Alexa top five hundred websites and the random five thousand websites, it shows a growing trend for the usage of PNG. Based on our current web analysis for the Alex top five hundred websites and the random five thousand websites, this trend is predicted to continue. The increase in usage for this type of graphical format may not be the results of other technologies, but by the capability of this type of format itself (see Appendix B). Hence after the release of its second edition a significant increase in usage was observed.

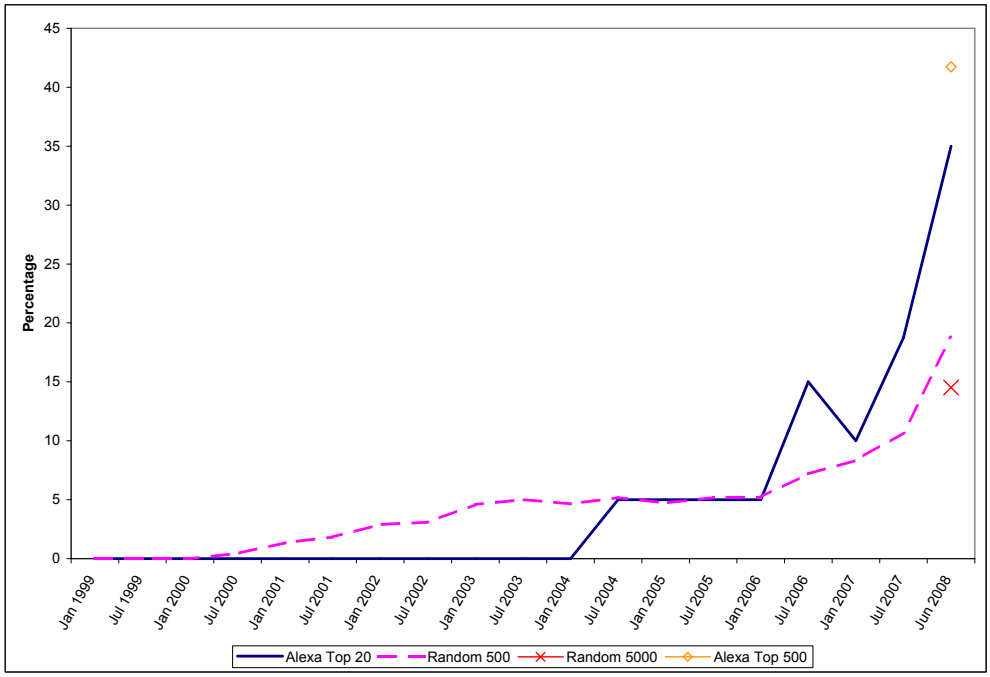

Figure 4.14: PNGs usage percentage for Alexa top 20 and 500 websites, and a set of 500 and 5000 random websites

The SVG format became a W3C recommendation from in 14 January 2003, but from the graphs shown in figure 4.15, a poor adoption rates were observed. A number of websites attempted to take up this format around July 2003 but quickly abandoned it. This could be due to a few reasons such as lack of software to create graphics for this type of format. Due to the poor results analysed, it was not justifiable to conduct a correlation test with these results. However looking that the Alexa top five hundred websites and the random five thousand websites results, together with the results for the random five hundred websites in June

2008, an slight increase for the usage of this graphical format was predicted, but no significant increase was expected since it is still in its first recommendation release. As observed from the other graphical formats trends, a significant increase will be expected after its revised or second edition is released.

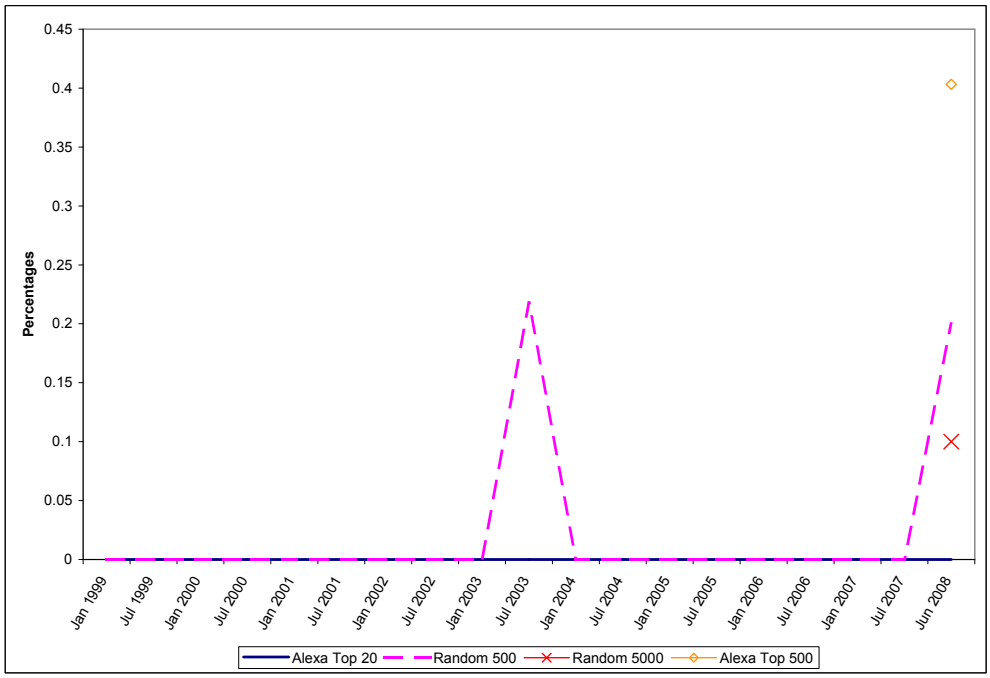

Figure 4.15: SVGs usage percentage for Alexa top 20 and 500 websites, and a set of 500 and 5000 random websites

In order to synchronise multimedia over the web, the W3C had introduced the SMIL format as a recommendation for synchronised multimedia. From figure 4.16 one can notice from the graphs that the usage of this type of format is very poor, and a few attempts by websites to take up this format can be observed. Using the data from the Alexa top five hundred and top twenty websites, together with the random five hundred and five thousand websites, no increase in usage was expected. Again due to the poor usage, no correlation test was conducted as it was not justifiable.

The last type of graphical format covered in this study is Flash. From figure 4.17, the usage of this for this type of graphical format for the last ten years were presented. A steady growth in the usage was noticed from Alexa top twenty websites and the random five hundred websites results. Based on the Alexa top five hundred websites and the random five thousand websites results to verify

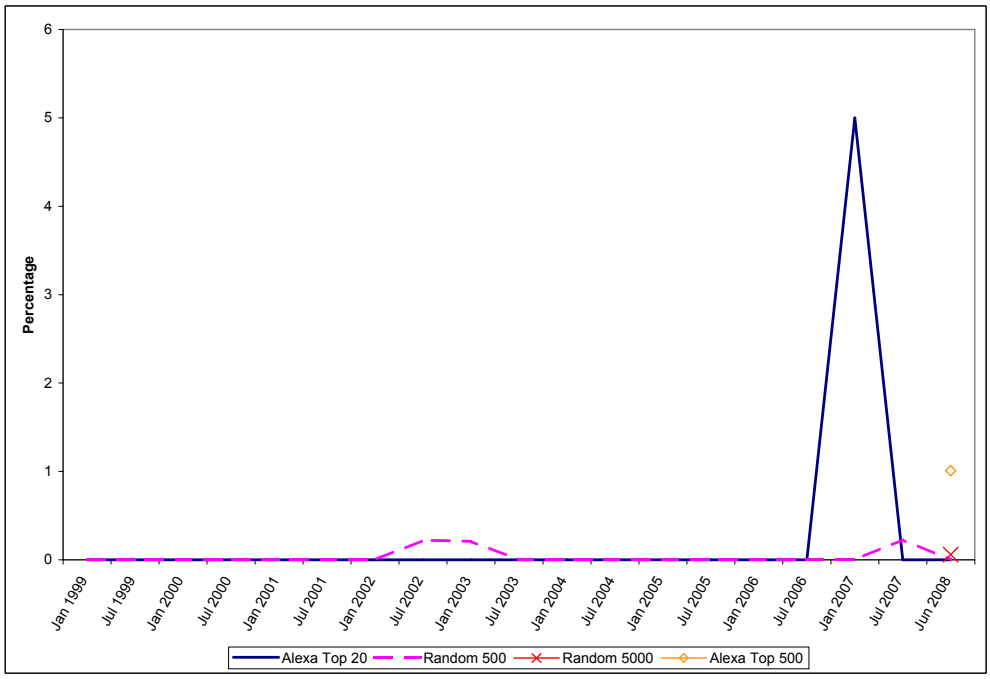

Figure 4.16: SMILs usage percentage for Alexa top 20 and 500 websites, and a set of 500 and 5000 random websites

these claims, a continuous growing trend was forecasted. Looking at the graphs a correlation between the Alexa top twenty websites and the random five hundred websites results was suggested. Hence Pearson correlation was applied, and a significant relationship between the Alexa top twenty websites and the random five hundred websites results was noticed, $r = .84$, p (two-tailed) $< .01$.

From the above results and discussions, based on the correlation results, besides GIF, all the other graphical format had a significant relationship was noticed between the Alexa top twenty websites and the random five hundred websites results. On the average it will take about two years for a new graphical format to get adopted. These results demonstrated that when analysing a graphical format usage trend, the Alexa top websites do gives a good indication of how the random web is evolving.

Further analysis was also done to see how the different graphical formats fair against each other. Figure 4.18 plots the results collected of the different graphical formats results from the Alexa top twenty websites. No relationship was noticed between the different graphical formats, but when observing the Alexa top twenty websites results, it was noticed that it is more likely for a graphical format to be

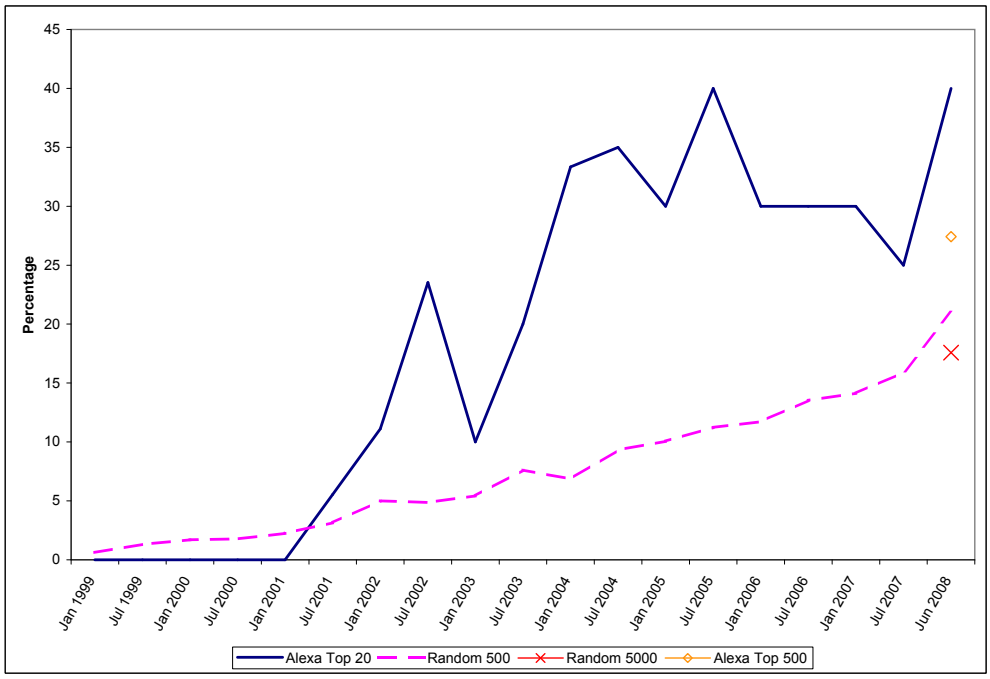

Figure 4.17: Flash usage percentage for Alexa top 20 and 500 websites, and a set of 500 and 5000 random websites

used side by side with another graphical format than to be replaced.

A similar analysis was also done for the random five hundred websites data set for the different types of graphical formats. As expected, a similar trend was noticed when the results were collected for the Alexa top twenty websites. However again no relationship was noticed between the different graphical formats, and it was more likely that a new graphical format will be used side by side with existing graphical format than to be replaced. However this set of results were more consistent since a larger set of websites were examined.

To conclude this section for the results and discussions for the different types of graphical formats, it can be observed that the Alexa top websites do give a good representation of how the web in-general was evolving for this type of analysis. On the average it will take about two years for a new graphical format to get adopted by the web from the time it was released. It is also more likely for a graphical format to get adopted, and to be used side by side with the older formats, then to be used as a replacement. Finally the Alexa top websites and the random websites do learn from each other trends when it comes to taking up new graphical formats as seen previously.

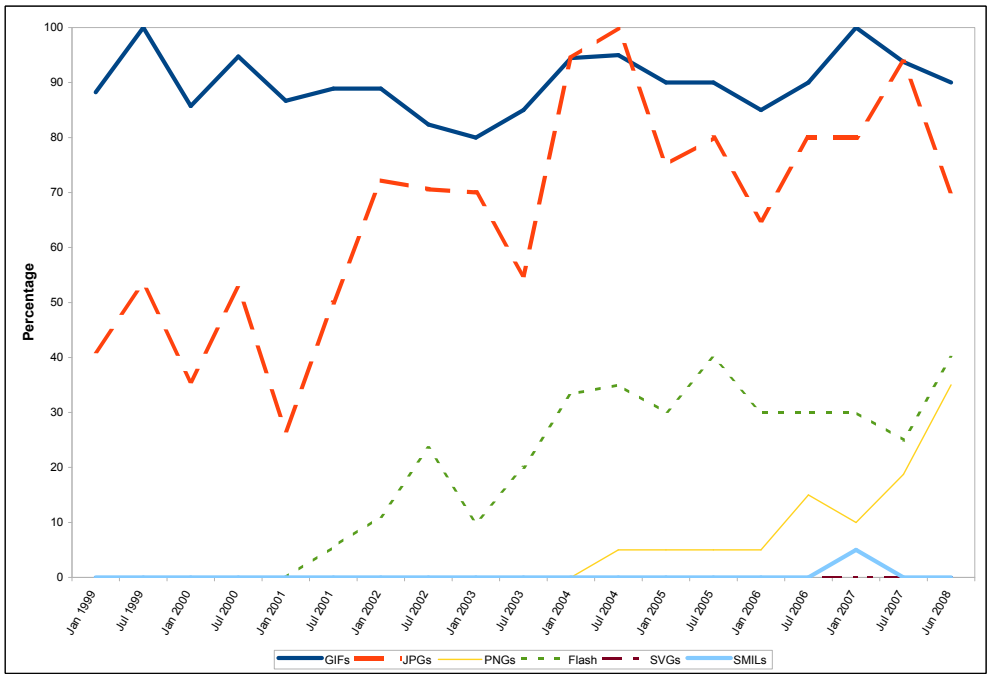

Figure 4.18: Graphical usage percentage for Alexa top 20 websites

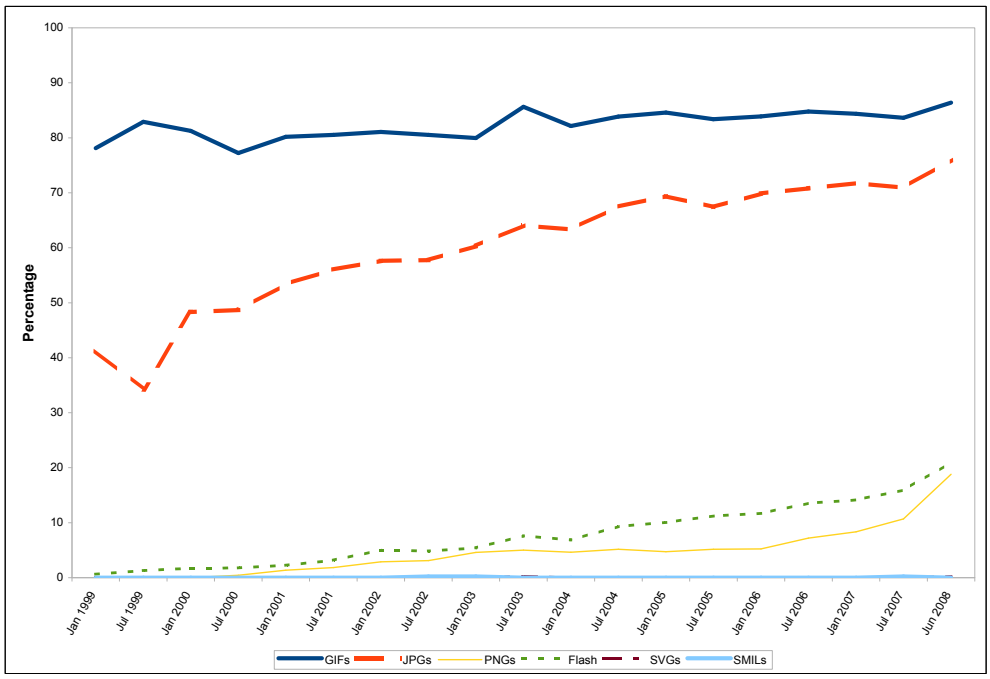

Figure 4.19: Graphical usage percentage for random 500 websites

4.4 Client-side Scripting Results

Client-side scripting plays a vita row in web development. As discussed earlier it provides the means for web developers/authors to control the appearance of the website, and to reduce servers work load that will help to make better use of the network traffic. Two types of client-side scripting were covered in this study; JavaScript and VBScript. Beginning with JavaScript lets looked at our analysis results in figure 4.20.

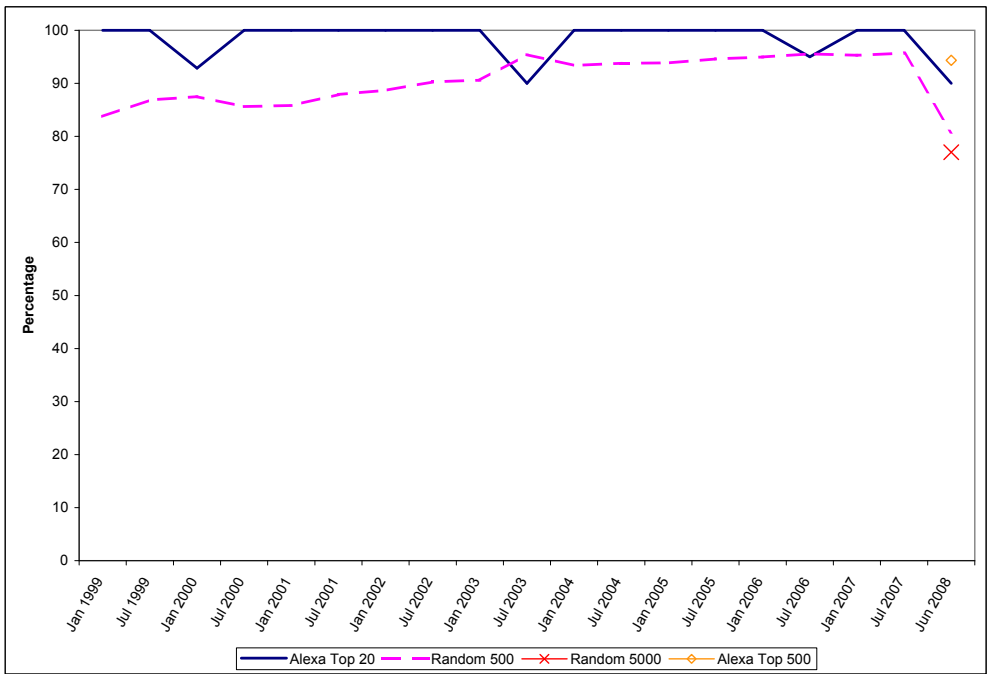

Figure 4.20: JavaScripts usage percentage for Alexa top 20 and 500 websites, and a set of 500 and 5000 random websites

It was observed that most of the Alexa top twenty websites uses some JavaScript in their website design, but a decline in usage was also noticed since July 2007. Our random five hundred websites results also demonstrated similarly trend. By using our Alex top five hundred websites and the random five thousand websites results to validate these claims, a continuous decline in JavaScript usage was predicted. Further analysis is required to understand more about what causes this decline. Some of the possible analysis will be discussed later in section 4.6.

VBScript is the other client-side scripting language covered in this study.

Commonly VBScripts will only be executed when run in Microsoft Internet Explorer. Due to its poor adoption rate by other user-agaents, poor usage by web developers/authors is expected for this type of client-side scripting.

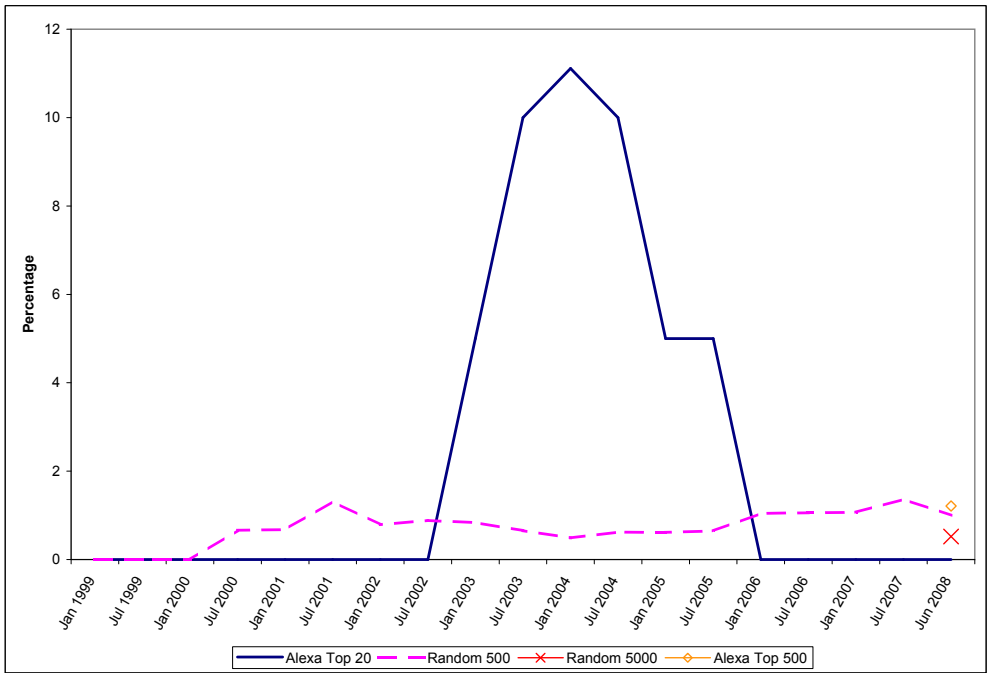

Figure 4.21: VBScripts usage percentage for Alexa top 20 and 500 websites, and a set of 500 and 5000 random websites

The graphs shown in figure 4.21 reflected our view based on both the Alexa top twenty websites and the random five hundred websites results. This type of client-side scripting language never seem to get adopted by the Alexa top twenty websites, even though a pick up in usage was notice between July 2002 and January 2006. Initially a gain in its popularity was noticed in year 2000 from the random five hundred websites results, but its usage percentage remained almost the same after that. Based on the results from the Alexa top five hundred websites and random five thousand websites, a similar percentage of usage for VBScript was expected for the near future, but no significant increase for foreseeable.

AJAX is a model created to take advantage of the popularity and capability of JavaScript, the asynchronous technology, and XML. Using the methodology discussed earlier in sub-section 3.3.5, data were extracted to identify the usage of AJAX. In figure 4.22 it shows that a growing usage trend for the AJAX model

using the combined result of the iFrame element and the XMLHttpRequest object. The Alexa top twenty websites led the way in the usage of AJAX, while the random five hundred websites grew in popularity gradually. Pearson correlation was applied to these results for a correlation test, and a significant relationship between them was noticed, $r = .75$, p (two-tailed) $< .01$.

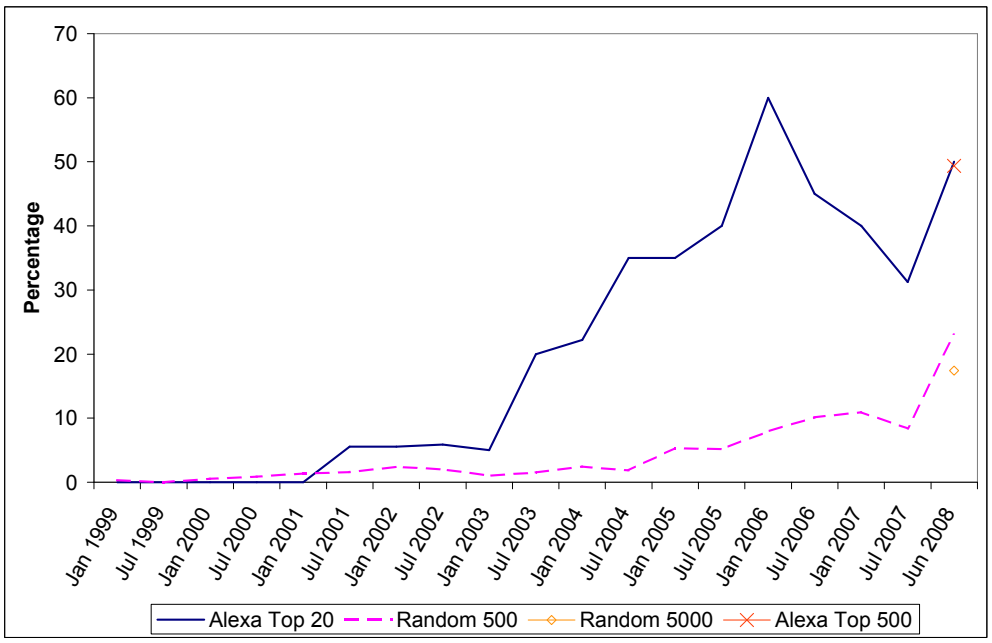

Figure 4.22: AJAX detection based on the combination of iFrames and XMLHttpRequest usage percentage for Alexa top 20 and 500 websites, and a set of 500 and 5000 random websites.

The analysis break down for the two methods used to detect the usage of AJAX were presented in figure 4.23 and 4.24. The first analysis was done by searching for the usage of the XMLHttpRequest object within the JavaScript source code, and the second analysis was done by searching the use of iFrame elements within the HTML code.

The usage results for AJAX detection using the XMLHttpRequest object within JavaScript was presented in figure 4.23. It shows that the Alexa top twenty websites led the trend while the random five hundred websites exhibited a similar trend. Pearson correlation was used to check if the both sets of results have any correlation. There was a significant relationship between the both sets of results, $r = .64$, p (two-tailed) $< .01$. These trends were verified using the

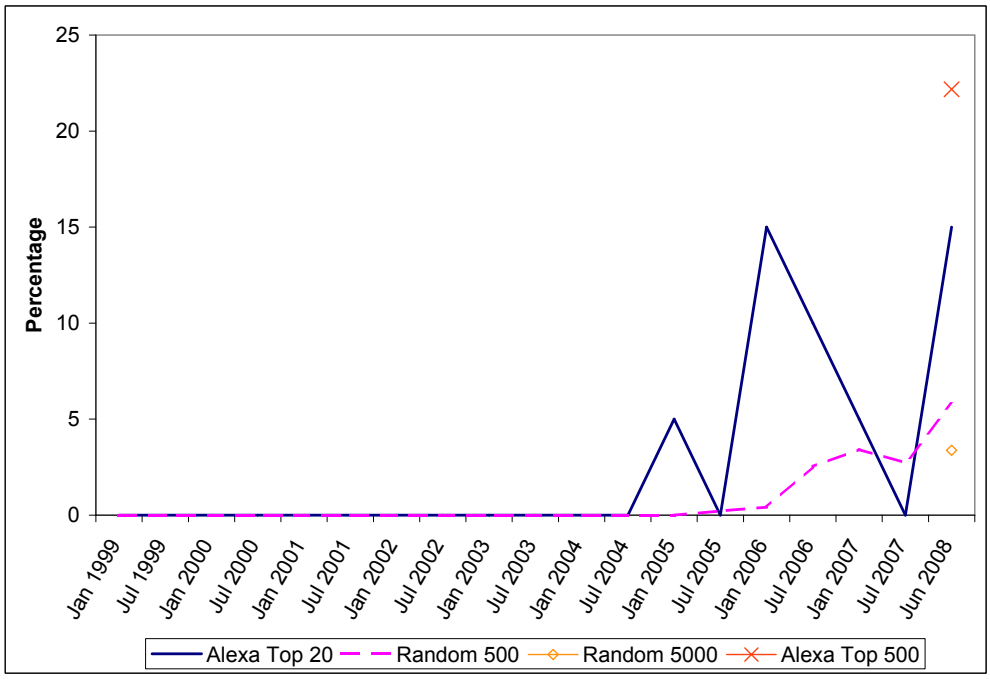

Figure 4.23: AJAX detection based on the XMLHttpRequest usage percentage for Alexa top 20 and 500 websites, and a set of 500 and 5000 random websites.

Alexa top five hundred websites and the random five thousand websites results. Using these results a forecasted increase in the XMLHttpRequest object usage was concluded.

The next analysis for AJAX detection was searching for the use of iFrame element(s) within the HTML code. These results showed that initially the random five hundred websites led the trend, but the Alexa top twenty websites were quick to pick up, and eventually surpassing the random five hundred websites to lead the usage trend. Applying Pearson correlation to the both sets of results gave a significant relationship between them, $r = .70$, p (two-tailed) $< .01$. Based on the Alexa top five hundred websites and the random five thousand websites results, a gradual increase in usage of iFrame element(s) can be expected.

To conclude this section, figure 4.25 and figure 4.26 demonstrated a huge difference between the usage of JavaScript and VBScript for both the Alexa top twenty websites and the random five hundred websites results. When comparing VBScript with JavaScript, VBScript seems to never get adopted by web developers/authors. However it was notice that the usage of JavaScript for both sets of data had begin to roll off since July 2007, hence further analysis such as plotting

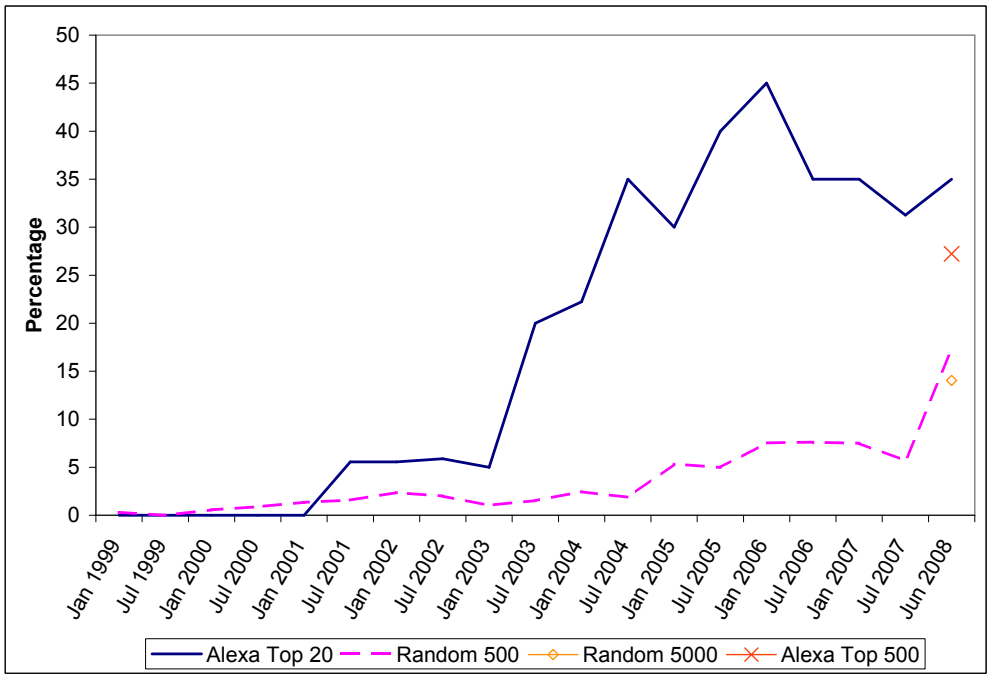

Figure 4.24: AJAX based on the iFrames usage percentage for Alexa top 20 and 500 websites, and a set of 500 and 5000 random websites.

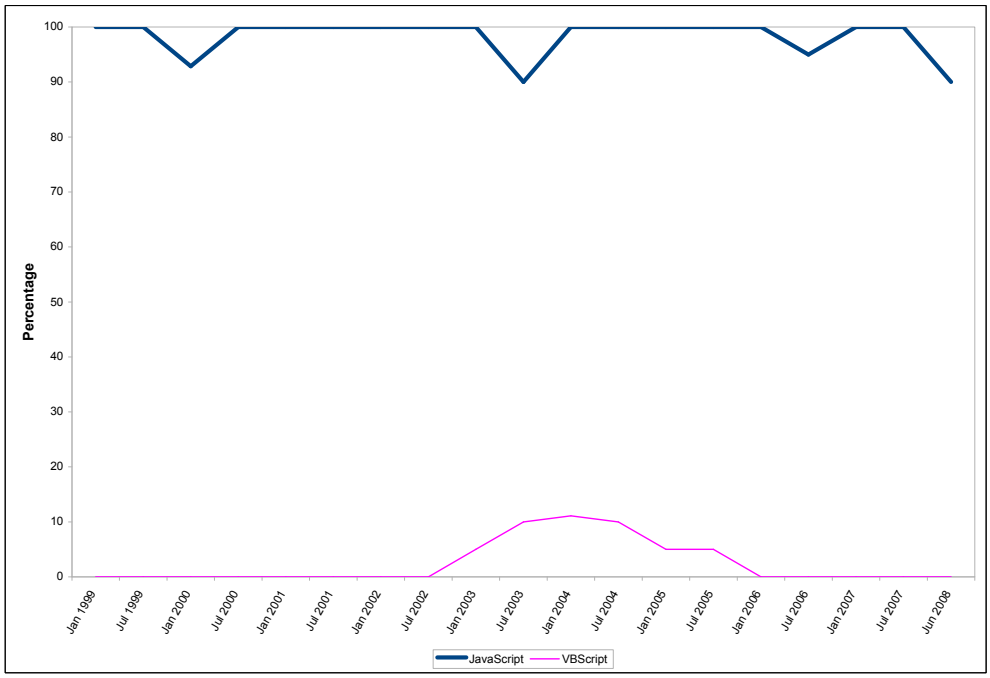

Figure 4.25: Scripting usage percentage for Alexa top 20 websites

JavaScript with AJAX, is required to understand more about the reason for this trend. On the average for all the analysis done on AJAX usage saw the Alexa top websites led the usage trend, and a significant relationship between the Alexa top twenty websites and the random five hundred websites results.

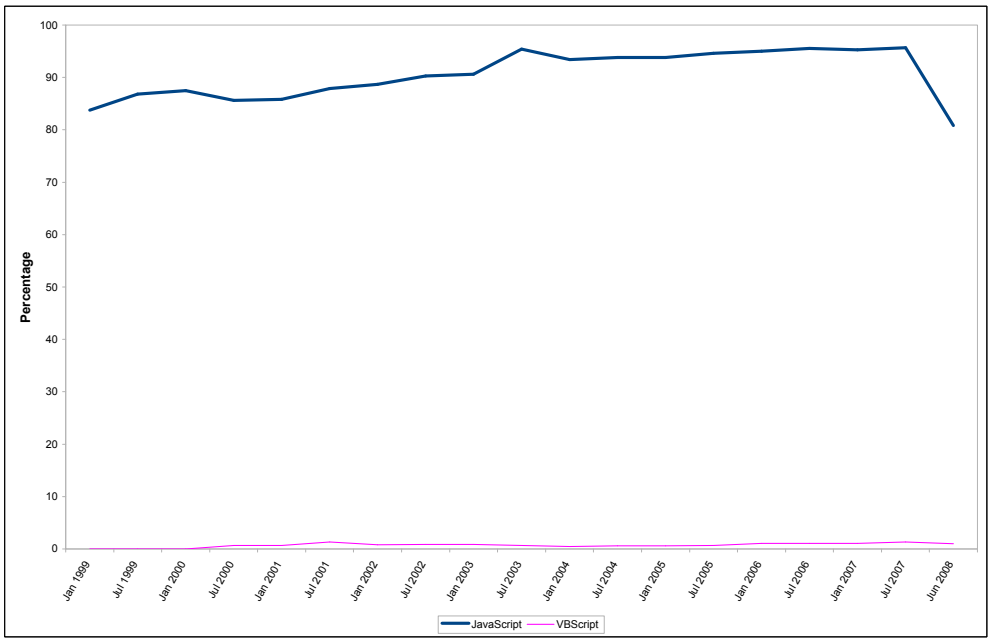

Figure 4.26: Scripting usage percentage for random 500 websites

4.5 Guidelines Conformance

As discussed previously, the WCAG will be the only guideline analysed in this study for web content accessibility conformance. Also discussed earlier, we will look for only the display of conformance to these guidelines on a website to detect if it is conformed to it. This can be in the form of displaying a logo or in plain text. The level of conformance was also search during this experiment, such as A, double A, or triple A.

Table 4.2 showed a poor conformance results collected using our method (see sub-section 3.3.2) for the pessimistic view. From the results presentted by Watanabe and Umegaki [46], we suspect that more of these websites may be conform to the WCAG 1.0 guidelines, but either not all of them display their conformance on

Year	Alexa Top 20		Alexa Top 500		Random 500		Random 5000	
	Pessimistic	Optimistic	Pessimistic	Optimistic	Pessimistic	Optimistic	Pessimistic	Optimistic
Jan 1999	0	0	-	-	0	0.94	-	-
Jul 1999	0	0	-	-	0	1.32	-	-
Jan 2000	0	7.14	-	-	0	0	-	-
Jul 2000	0	5.26	-	-	0	1.55	-	-
Jan 2001	0	13.33	-	-	0	1.13	-	-
Jul 2001	0	11.11	-	-	0	1.84	-	-
Jan 2002	0	11.11	-	-	0	2.89	-	-
Jul 2002	0	11.76	-	-	0	2.43	-	-
Jan 2003	0	10	-	-	0.21	2.51	-	-
Jul 2003	0	10	-	-	0.22	3.05	-	-
Jan 2004	0	5.56	-	-	0	2.93	-	-
Jul 2004	0	5	-	-	0	2.69	-	-
Jan 2005	0	0	-	-	0	3.29	-	-
Jul 2005	0	10	-	-	0	3.89	-	-
Jan 2006	0	10	-	-	0	5.02	-	-
Jul 2006	0	10	-	-	0	5.29	-	-
Jan 2007	0	10	-	-	0	6	-	-
Jul 2007	0	12.5	-	-	0	7.27	-	-
Jun 2008	0	5	0	3.43	0	7.8	0.06	2.68

Table 4.2: WCAG 1.0 conformance results in percentage for both pessimistic view and optimistic view. Notice that for both Alexa Top 500 and Random 5000 data, only June 2008 was presented, this was because these sets of data only looks at the current web as discussed earlier.

their websites or they may be in the form of plain that were displayed before the last one hundred characters. Thus our optimistic view experiment showed some more promising results that were closer to those mentioned by Watanabe and Umegaki. Pearson correlation was applied to check if there was any correlation between the Alexa top twenty websites and the random five hundred websites results, but no significant relationship was found as one would expect from figure 4.27. With these results, as seen in figure 4.27 demonstrated that the Alexa top twenty websites were quicker to be adopt by these guidelines then our random five hundred websites. It also shows that the Alexa top twenty websites led the trend for conformance, while our random five hundred websites were gradually catching up. Hence it can be concluded that more websites prefer to display their conformance via plain text, and this can be found in any part of a webpage. After validating the historical data results with the current web results, no increase in conformance was forecasted.

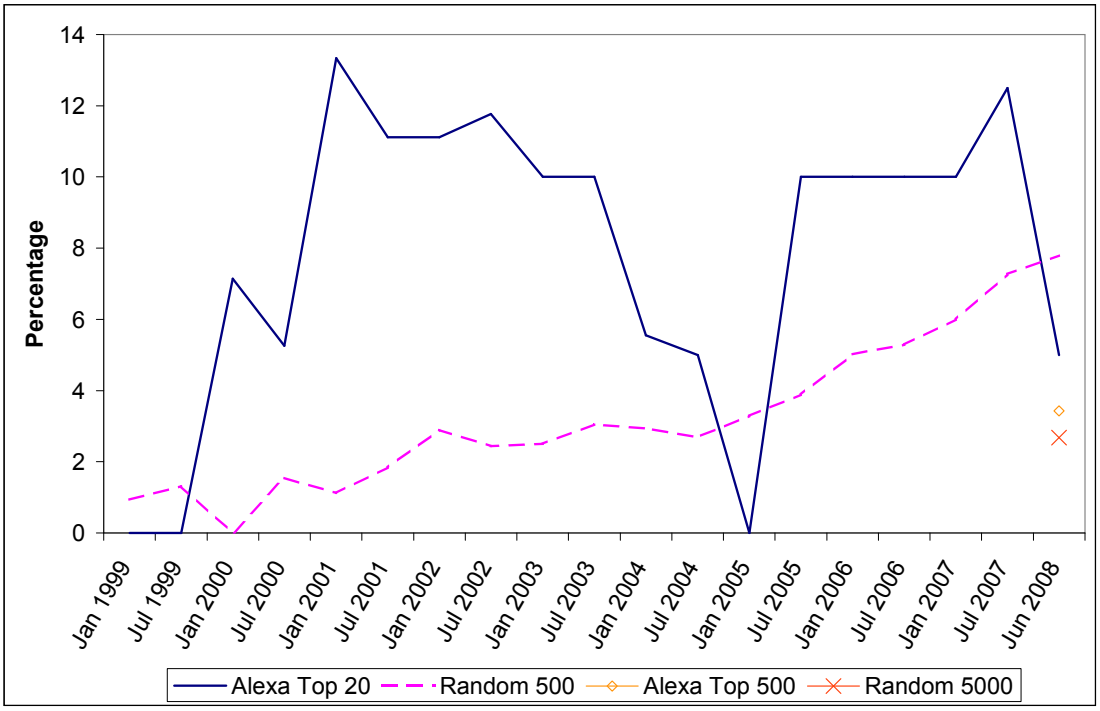

Figure 4.27: WCAG 1.0 conformance percentage (Optimistic view) for Alexa top 20 and 500 websites

These results demonstrated that little increase in the conformance to the

WCAG 1.0 guidelines had be achieved for the last ten years, since May 1999 when it became a W3C recommendation. The adoption rates for WCAG 1.0 never seems to improve when comparing with the other W3C recommendations discussed earlier. A lot more research is required to understand the reasons for these trends, and more aiding tools are suggested to make these guidelines seems easier to be taken up or conformed by more websites. One of the reasons for the low conformance rate is due to the small user population that it will benefit, thus the economical benefits return is not huge. Another reason for this low conformance rate could be due to the many different types of web content accessibility guidelines available, thus it may seem confusing, difficult, and not beneficiary for web developers/authors to conform to them when their economical return are low. However in a recent report by Yesilada et. al., suggests that both people with or without disabilities experience similar limitations, and barriers when interacting with websites on mobile devices [47]. This claim may put forward a better case for web developers/authors to conform to these guidelines as it will benefit a larger user population.

4.6 Further Analysis

Now that we had discuss all the results from the individual standards and recommendations analysis, further analysis were also conducted to understand more about the reasons behind some of these standards and recommendations usage trends. A few analysis were suggested to be done so that better understanding between the relationships and reason behind these standards and recommendations trends. The first analysis was done between CSS and JPEG because through visual observation the CSS (figure 4.2) usage seems to possess similar growth trend patterns with JPEG (figure 4.13). This analysis will explain the growth in usage for some graphical formats and what effected it.

Both CSS and JPEG demonstrated similar trends in figure 4.28. Pearson correlation was used to determine if a correlation exist between these recommendations for the Alexa top twenty websites, and the random five hundred websites results. A significant relationship was noticed for the Alexa top twenty websites between the CSS results and the JPEG results, $r = .75$, p (two-tailed) $< .01$. There was also a significant relationship for the random five hundred websites between the CSS results and the JPEG results, $r = .95$, p (two-tailed) $< .01$.

Figure 4.28: CSS VS. JPEG percentage for Alexa top 20 and 500 websites

Hence there was clearly a significant correlation between CSS and JPEG results, and they possess similar usage trends. This suggest that the take up of JPEG may have some relationship with the increase in usage for CSS. The reason for this claim was because CSS allows the web developers/authors the flexibility, and more control over the webpage's presentation, thus this led to an increase in the usage for different types of graphical formats such as JPEG.

Two other analysis were conducted to understand reasons for the decline of JavaScript usage trend. From these analysis, we hope to understand why introducing a model that uses existing standards, and recommendations may not necessary improves the technology's popularity. The first one was between AJAX and JavaScript as shown in figure 4.29. The results from this analysis gave an interesting view of JavaScript and AJAX usage trends. It can be noticed that even when the usage of AJAX was increasing, a decline in JavaScript usage was still observed. Therefore another analysis was carried out between Flash and JavaScript as seen in figure 4.30. Again it was noticed that around July 2007, an increase in Flash usage was noticed around the same time when JavaScript began to roll off. This analysis supplies a reason for the trend of the increase

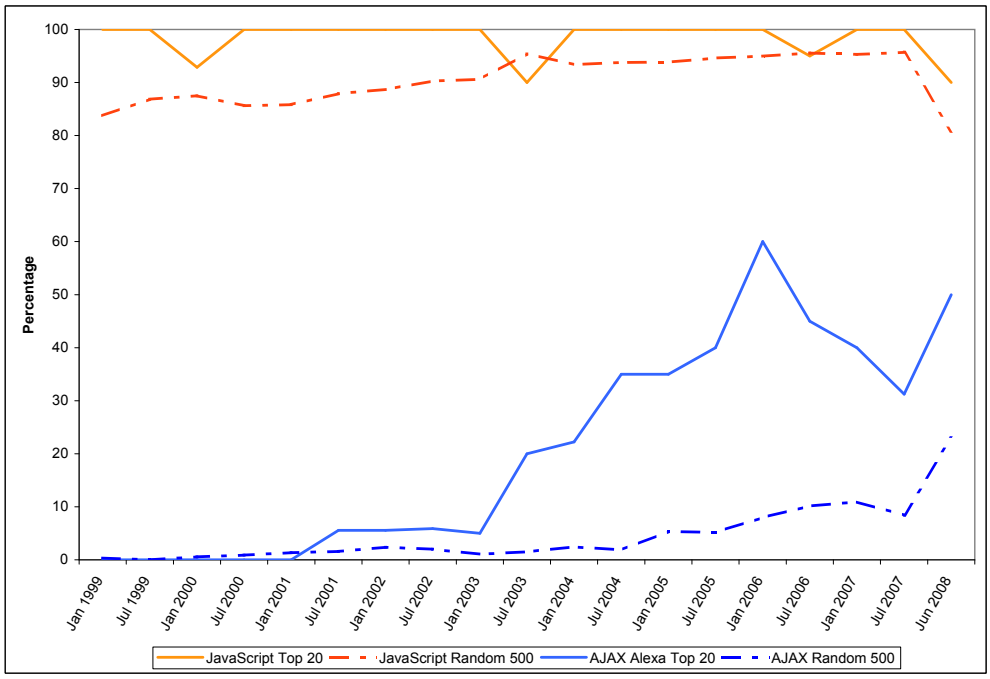

Figure 4.29: AJAX VS. JavaScript percentage for Alexa top 20 and 500 websites

in AJAX popularity, and the contrary results for the usage of JavaScript trend. This showed that the AJAX model may be gaining popularity, but it may not be enjoying the full increase, but it was sharing it with another technologies such as Flash that is capable of providing the asynchronous model.

To conclude the section on further analysis, the increase in usage for graphical formats such as JPEG may be the benefiting from the fruits of the CSS usage increase. This is because the CSS allows the web developers/authors more flexibility and control over the webpage presentation, thus this allows better use of graphics. The other analysis conducted surrounding AJAX conclude that it is not enjoying the full popularity of the asynchronous technology revolution, but it is sharing it with other web technologies such as Flash. Since the roll off of JavaScript has just began in 2007 further research will be required to determine if this is a true decline or was it just a dip in usage.

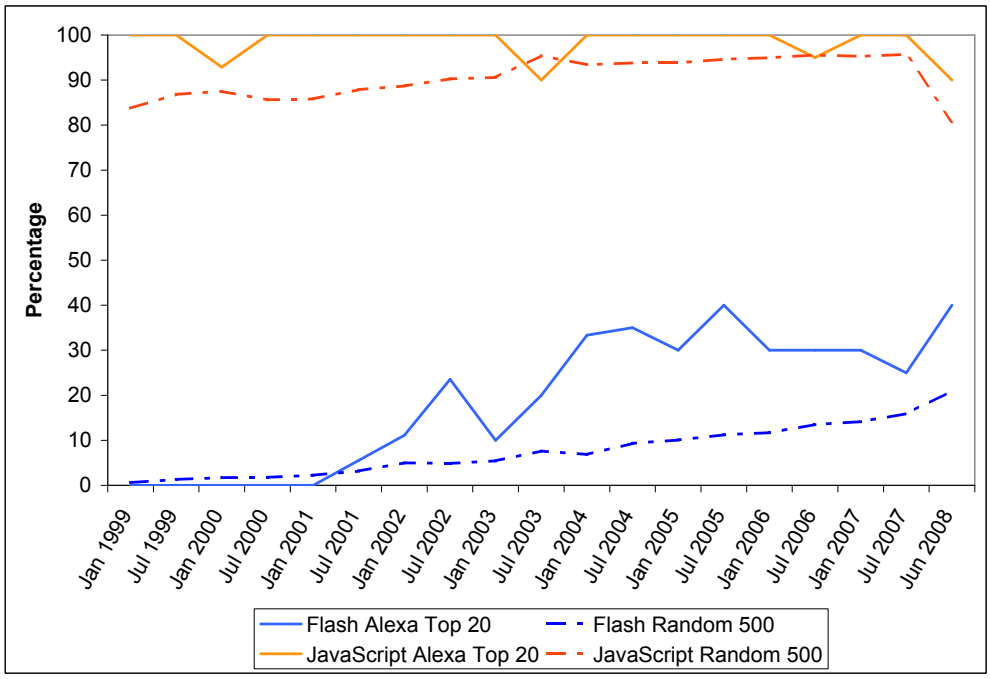

Figure 4.30: Flash VS. JavaScript percentage for Alexa top 20 and 500 websites

4.7 Analysis Overview

The discussions on the analysis presented above highlights and predicts possible trends for individual standards, and recommendations when done separately. However when plotted against each other, further understanding for the reasons behind these trends were better explained. As discussed earlier, the Alexa top websites does give a good representation of the random web when analysing the W3C standards and graphical formats. From the analysis results done on the web content accessibility conformance, no increase in conformance for the WCAG 1.0 guidelines was forecasted. Further analysis were conducted to understand more about the reasons behind some of these trends such as AJAX, CSS, Flash and JPEG. It was noticed that more websites were taking up the asynchronous model, but this trend was shared between AJAX and Flash. CSS has given the web the flexibility and the control over the presentation of web contents. Due to this a healthy usage growth for this standard was predicted, and graphical formats such as JPEG usage also benefiting from it.

Chapter 5

Conclusion And Future Work

The study of the evolution of the web were conducted for different purposes. From this study, it will provide the proofs and recommendations to our arguments proposed earlier, and to understand the relationship and trends between the underlying web standards, recommendations, guidelines, and its adoption time. A lag was noticed between the time these standards and recommendations were introduced till the time when they were adopted. This causes a disconnection between the actual user experience, and what was expected by the technology stake-holders. Thus for these issues, understanding the evolution of the web will helps us to understand the relationship and trends between the underlying standards, recommendations, guidelines and its adoption time. In this study, we focused on the human factors issues surrounding the evolution of the web user interface. Since the web is constantly evolving, tracking the adoption for these standards, recommendations, and guidelines will not be easy. Web robots were employed to capture and carry out the web mining processes for further analysis in this study. From the results of these analysis, recommendations and answers to the arguments presented earlier were achieved. Thus these results will act as recommendations for future work surrounding the human interaction between the web user interface.

The process for selecting the websites began as early as 13 June 2008, but due to missing data from Internet Archives, and no longer existing website chosen, our capturing and data integrity validation process lasted for about a month, between 24 June 2008 and 24 July 2008. Thus the analysis part could only begun in early August 2008, which was behind our scheduled by a few days. However this was because stages two and three (see figure 3.4) were done thoroughly to ensure data

integrity, thus most of our results analysed was of satisfactory expectation.

From our analysis, we can conclude that the evolution for both the W3C standards and graphical formats can reliability be represented by the Alexa top websites to give a good idea of how the web was evolving. On average, the Alexa top websites adopts to a W3C standard one year earlier than the random websites, and websites are replacing their HTML 4 webpages with XHTML 1.0. This trend was predicted to continue for the near future. For the graphical formats, it was observed that a new graphical format if adopted will more likely be used side by side with a older format than to replace it. Besides this both the Alexa top websites and the random websites do learn from each other's trend when it comes to taking up a new graphical format. An initial roll off for JavaScript usage was noticed even though an increase in AJAX usage was found. However from our further analysis this may be due to websites converting to Flash technology. However further work must be conducted to analyse if the roll off of JavaScript's usage was a true decline in usage, or was it just a dip. The competitor of JavaScript, the VBScript, never seem to get adopted by web developers/authors over the last ten years. This is mainly due to the poor adoption of this client-side scripting language by popular user-agents. Our web content accessibility conformance analysis had presented very poor results for our pessimistic view, however a more promising results from our optimistic view was reported. these results were closer to those mentioned by Watanabe and Umegaki [46]. From our analysis, no increase in conformance was forecasted for the WCAG 1.0 guidelines, however from a recent report by Yesilada et. al. [47], this may put forward a better case that will encourage more web developers/authors to conform to these guidelines since the user population is larger.

The results discussed earlier in this study (see chapter 4) showed that not all technologies get adopted by the web such as VBScript, however new technologies were created to provide either better web experience, or to overcome existing problems. The adoption of technologies depends on the needs of the web users, and the web developers/authors to convey their work across the web. As discussed previously when comparing the relationships between the WCAG conformance results and the W3C standards usage, these results demonstrated that the adoption of guidelines do not necessary affect the usage trend for a technology. Thus the technological adoption by the web users, developers, authors, and user-agents

do not get affect by the trends of the adoption of guidelines. Most technical intervention are likely to get adopted by the popular user-agents so that their existing users, and the new users will be attracted to use their products. However as seen from the discussions on our VBScript's analysis, the poor adoption of this technology by the popular user-agents do affect its usage popularity. From this case, a relationship between the technology's popularity and the adoption of the technology by the user-agents can be noticed. The history of the web can help us to understand and predict the trends of existing technologies and guidelines, as seen from most of our discussions, but since the web is constantly evolving, relying on these information is not sufficient to lead the web. The usage of a technology over the web will represent the economical returns for a technology or guidelines, hence this will lead to new interventions by engineers. Finally we suggest that using history together with the take up of technologies by the users, and the interventions from engineers are the best approach to lead the future of the web.

5.1 Future Work

This study gave an introduction of how the web was evolving over the past ten years (1999-2008), and its existing problems. From these findings, future work relating to this study were highlighted, and justified. This can be a continuation of the existing work, a rectification of the method used to analyse a recommendation or standard in this study, or the issues that were highlighted from this study. Below are some suggestions for the possible future work surrounding this study.

(1) Web content accessibility analysis A different method or approach to analyse the conformance of web content accessibility was suggested. Verifying each websites for the conformance to the guidelines is the next approach suggested. May be analysing the conformance with a different type of guidelines may give a better conformance rate, for example instead of WCAG 1.0, Web Aim guidelines can be used. From here another future work such as analysing the conformance to different accessibility guidelines for the same webpage is presented in our next point.

(2) The conformance of the different accessibility guidelines There are a number of web content accessibility guidelines such as WCAG, section 508 and Stanca Act. Analysing the popularity of the different guidelines can help to understand the evolution of the accessibility guidelines. Research can be done to understand if there was a common protocol for these guidelines, and these may give rise to one standard guideline that will make it seems easier for web developers/authors to conform. This type of research will act as recommendations for future work surrounding web content accessibility and the web community.

(3) Continuation of this study A number of questions had risen from this study such as 'Why are SVG and SMIL usage so low?', and the approach to check for web content accessibility conformance with a different method is suggested. A larger number of graphical formats is also suggested to provide a more comprehensive study.

(4) Analysing the model/standards that uses the asynchronous technology When conducting this study for AJAX, different web technologies that uses the asynchronous technology, and issues relating to it were visited. Some of the issues include 'How does asynchronous technology improves or worsen the accessibility of web content for a webpage using it?'. This type of technology will require the use of some special techniques, or methodology to analyse the "deeper web" as discussed by O'Neill et. al. [36]. Hence studying the evolution of asynchronous technology for the different models, standards and technologies such as AJAX, iFrames, and Active X will help to understand the relating issues better.

Bibliography

[1] A snapshot of cyberspace. *Library of Congress Bulletin*, 57(11), November 1998.

[2] R. Albert, H. Jeong, and A.-L. Barabasi. The diameter of the world wide web. *Nature*, 401:130–131, Sep 1999.

[3] E. Amitay, D. Carmel, A. Darlow, R. Lempel, and A. Soffer. The connectivity sonar: detecting site functionality by structural patterns. In *HYPERTEXT '03: Proceedings of the fourteenth ACM conference on Hypertext and hypermedia*, pages 38–47, New York, NY, USA, 2003. ACM.

[4] J. Axelsson, M. Birbeck, M. Dubinko, B. Epperson, M. Ishikawa, S. McCarron, A. Navarro, and S. Pemberton. `http://www.w3.org/TR/xhtml2/`, July 2006.

[5] P. Ballard and M. Moncur. *Sams Teach Yourself Ajax, JavaScript and PHP All in One*. Sams Publishing, 1st edition, 2009.

[6] BBC News. Was Y2K bug a boost? `http://news.bbc.co.uk/1/hi/sci/tech/590932.stm`, January 2000.

[7] BBC News. Fifteen years of the web. `http://news.bbc.co.uk/go/pr/fr/-/2/hi/technology/5243862.stm`, August 2006.

[8] T. Berners-Lee. WWW: past, present, and future. *Computer*, 29(10):69–77, 1996.

[9] K. Bharat and A. Broder. A technique for measuring the relative size and overlap of public web search engines. *Comput. Netw. ISDN Syst.*, 30(1-7):379–388, 1998.

[10] B. Bos. W3C Cascading Style Sheets (CSS). `http://www.w3.org/Style/CSS/`, February 2008.

[11] B. E. Brewington and G. Cybenko. How dynamic is the web? *Computer Networks*, 33(1-6):257–276, June 2000.

[12] A. Budd. *CSS Mastery : Advanced Web Standards Solutions*. Apress, 2006.

[13] D. Bulterman and et al. Synchronized Multimedia Integration Language (SMIL 3.0). `http://www.w3.org/TR/SMIL3/`, 2008.

[14] B. Caldwell, M. Cooper, L. Reid, G. Vanderheiden, W. Chisholm, J. Slatin, and J. White. WCAG 2.0. `http://www.w3.org/TR/WCAG20/`, December 2007.

[15] K. Chandra, S. S. Chandra, and S. S. Chandra. A comparison of VBscript, Javascript, and Jscript. *J. Comput. Small Coll.*, 19(1):323–335, October 2003.

[16] L. Cherkasova and M. Karlsson. Dynamics and evolution of web sites: Analysis, metrics and design issues. *Sixth IEEE Symposium on Computers and Communications (ISCC'01)*, 2001.

[17] W. Chisholm, G. Vanderheiden, and I. Jacobs. Web Content Accessibility Guidelines 1.0. `http://www.w3.org/TR/WAI-WEBCONTENT/`, 1999.

[18] J. Cho and H. Garcia-Molina. The evolution of the web and implications for an incremental crawler. In *VLDB '00: Proceedings of the 26th International Conference on Very Large Data Bases*, pages 200–209, San Francisco, CA, USA, 2000. Morgan Kaufmann Publishers Inc.

[19] J. Clark. *Building Accessible Websites*. New Riders Publishing, United States of America, 2003.

[20] Connolly. HTML 3.0 draft. `http://www.w3.org/MarkUp/html3/`, December 1995.

[21] D. Connolly. `http://www.w3.org/MarkUp/html-spec/`, September 1999.

[22] F. Douglis, A. Feldmann, B. Krishnamurthy, and J. Mogul. Rate of change and other metrics: a live study of the world wide web. In *USITS'97: Proceedings of the USENIX Symposium on Internet Technologies and Systems on USENIX Symposium on Internet Technologies and Systems*, pages 14–14, Berkeley, CA, USA, 1997. USENIX Association.

[23] D. Fetterly, M. Manasse, M. Najork, and J. Wiener. A large-scale study of the evolution of web pages. In *WWW '03: Proceedings of the 12th international conference on World Wide Web*, pages 669–678, New York, NY, USA, 2003. ACM Press.

[24] I. S. Graham. *XHTML 1.0 language and design sourcebook: the next generation HTML*. Wiley, 2000.

[25] A. Gulli and A. Signorini. The indexable web is more than 11.5 billion pages. In *WWW '05: Special interest tracks and posters of the 14th international conference on World Wide Web*, pages 902–903, New York, NY, USA, 2005. ACM Press.

[26] S. Hackett, B. Parmanto, and X. Zeng. A retrospective look at website accessibility over time. *Behaviour and Information Technology*, 24(6):407–417, December 2005.

[27] S. Harper. Web evolution and its importance for supporting research arguments in web accessibility. In *Web Science Workshop (WSW2008)*. International World Wide Web Conference, 2008.

[28] S. Lawrence and L. C. Giles. Accessibility of information on the web. *Nature*, 400(6740):107–107, July 1999.

[29] C. Lilley and D. Schepers. Scalable Vector Graphics (SVG). `http://www.w3.org/Graphics/SVG/`, 2008.

[30] C. J. Lyons. *Essential Design for Web Professionals*. Pentice Hall PTR, 1st edition, 2001.

[31] S. Mckeever. Understanding web content management systems: evolution, lifecycle and market. *Industrial Management & Data Systems*, 103(9), 2003.

[32] J. Meiert. HTML elements index. `http://meiert.com/en/indices/html-elements`, June 2008.

[33] S. Murugesan. Understanding web 2.0. *IT Professional*, 9(4):34–41, 2007.

[34] J. Niederst. *Web Design In A Nutshell: A Desktop Quick Reference*. O'Reilly, 2nd edition, 2001.

[35] A. Ntoulas, J. Cho, and C. Olston. What's new on the web?: the evolution of the web from a search engine perspective. In *WWW '04: Proceedings of the 13th international conference on World Wide Web*, pages 1–12, New York, NY, USA, 2004. ACM Press.

[36] E. T. O'Neill, B. F. Lavoie, and R. Bennett. Trends in the evolution of the public web (1998 - 2002). *D-Lib Magazine*, 9(4), April 2003.

[37] L. Page, S. Brin, R. Motwani, and T. Winograd. The pagerank citation ranking: Bringing order to the web. Technical report, October 2001.

[38] Patrick and S. Horton. *Web Style Guide: Basic Design Principles for Creating Web Sites*. Yale University Press, New Haven and London, 1st edition, 1999.

[39] S. Pemberton and et al. XHTML 1.0 (second edition). `http://www.w3.org/TR/xhtml1/`, August 2006.

[40] D. Raggett. HTML 3.2 reference specification. `http://www.w3.org/TR/REC-html32`, Jan 1997.

[41] F. Ricca and P. Tonella. Web site analysis: structure and evolution. In *Software Maintenance, 2000. Proceedings. International Conference on*, pages 76–86, San Jose, CA, USA, October 2000.

[42] J. T. Richards and V. L. Hanson. Web accessibility: a broader view. In *WWW '04: Proceedings of the 13th international conference on World Wide Web*, pages 72–79, New York, NY, USA, 2004. ACM.

[43] L. Seeman, M. Cooper, R. Schwerdtfeger, and L. Pappas. WAI-ARIA Version 1.0. `http://www.w3.org/TR/wai-aria/`, February 2008.

[44] Shawn. WAI. `http://www.w3.org/WAI/`, March 2008.

[45] M. Toyoda and M. Kitsuregawa. Extracting evolution of web communities from a series of web archives. In *HYPERTEXT '03: Proceedings of the fourteenth ACM conference on Hypertext and hypermedia*, pages 28–37, New York, NY, USA, 2003. ACM Press.

[46] T. Watanabe and M. Umegaki. Capability survey of japanese user agents and its impact on web accessibility. In *W4A: Proceedings of the 2006 international cross-disciplinary workshop on Web accessibility (W4A)*, pages 38–48, New York, NY, USA, 2006. ACM.

[47] Y. Yesilada, A. Chuter, and S. L. Henry. Shared web experiences: Barriers common to mobile device users and people with disabilities (table format), July 2008.

Appendix A

HyperText Markup Language (HTML)

The HTML is a Standard Generalized Markup Language (SGML) format. SGML is an ISO standard created for defining markup languages. It is a metalanguage for defining markup language rules where one can define the grammar and syntax rules for a markup language. With SGML, every document consists of two parts. (1) The document itself (tag and content) and (2) a resource called the document type declaration (DTD). The DTD defines the languages grammar; that is, it defines the names of the elements, the attributes that each type of elements supports, and the grammar for the element. Web parsers (HTML parsers used by web browsers) should ignore tags which they do not understand, and similarly it will ignore attributes belonging to tags which they do understand [24].

More currently, the eXtensible HyperText Markup Language (XHTML) was introduced as the successor of HTML. It avoids many of the problems laid down by HTML while ensuring its design to be backward compatible (to HTML) for Web browsers which do not understand XHTML but HTML.

A.1 eXtensible HyperText Markup Language (XHTML)

XHTML is an revised version of HTML 4 as an XML 1.0 application, and the corresponding DTDs to the ones defined by HTML 4. This revision will overcome

the previous issues where HTML documents are not valid XML, thus XML processors cannot interpret them, and therefore previous HTML documents cannot be mixed with XML documents [39, 24].

The World Wide Web Consortium (W3C) is currently in its working draft of the second edition of XHTML (XHTML 2.0). This revision attempts to make XHTML 2.0 as generic to XML as possible; this means that existing facilities in XML will be use, rather than re-inventing it. Lesser presentation resources (this is left to CSS), better internationalization, and better usability are some of the objectives included when developing XHTML 2.0. However to ensure that XHTML 2.0 will still be usable by older web browser that do not support it, XHTML 2.0 is developed to be backward compatible to it predecessor; HTML and XHTML 1.0 [4].

The web is evolving rapidly and consistently. This new addition to the web will enable web developers to relish the existing benefits of XML while ensuring their content to be backward and future compatibility. Hence understanding the evolution of such technologies will help identify the trends between the technologies. Styles can be added to the presented text over the web. The main standard that allow this to happen over the web is CSS. This will be discussed in further depth in the next section.

A.2 Cascading Style Sheets (CSS)

One of the ways which a web developer can add style to their web document is to use CSS. In fact, this technology has become a standard by W3C for web page design, which gives a reason why this form of technology needs to be addressed in our web evolution analysis. In order to use CSS efficiently, having a well-structured and meaningful code (e.g. XHTML) is important. CSS makes it possible to control how a page is presented, and it separates the presentational part of the web document from its content and the computation required by the client-side (e.g. via JavaScript). Instead of overusing complex table structures to control the layout of a web document, CSS allows a better way to do this in a more controlled environment. These advancement allow web developers to develop the designs of the web pages without affecting the under laying content and Markup languages. However, CSS may behave slightly different when the web document is viewed on different web browsers [12, 10].

Appendix B

Graphical Formats

The use of graphics not only enhances the design of the web page, but also brings out contents that are important and difficult to explain when dealt with correctly. The WAI guidelines (see section 2.4) attempts to cover the associated accessibility issues relating to graphical presentation, but a lot are still left to the web content authors to conform to them. To begin with, lets cover the characteristics of web graphics for the common types of formats.

- GIF (Graphic Interchange Format)
  Lossless compression is employed for this type of file format. It is possible to make a colour transparent in GIF, and it allows the combination of multiple GIF images into a single file to create an animation. However, colours are limited to 8-bit (256 or fewer colours) and the transparent property is not selective [38].

- JPEG (Joint Photographic Experts Group)
  This type of file format was developed specifically for photographs and other images with lots of colours. Unlike GIF graphics, JEPG images are "true colours" or have a depth of 24 bits; this is far more than the human eye can see. On the contrary, it uses a lossy compression algorithm (loss in image quality) and colours cannot be transparent [30].

- PNG (Portable Network Graphics)
  The PNG graphical format was created to overcome the limitations of GIF format. It is a well-specified standard for lossless bitmapped image files and inherits the good attributes of GIF format. Additionally, it includes "true colours" images and it is designed to be portable, simple, robust,

supports full file integrity-checking, and quick, simple detection of common transmission errors [30].

- Flash
  Initially Flash was commonly also referred to as Shockwave Flash/Macromedia Flash/Adobe Flash which was developed to add animation and interactivity to web pages. It can manipulate with vector and raster graphics for animation, and support streaming of audio, and video. However, the web audience is required to have a Flash player plug-in installed on their computer before this type of format can be played back. The version of the Flash player plays an important role as it will determines how advance the Flash application can be played on a particular computer [34].

- Scalable Vector Graphics (SVG)
  SVG is a two-dimensional structured, vector-based graphics described in XML. Due to this it provides a means to make self-describing libraries. Nevertheless, it has the ability to be scalable (clean zoom), and supports animation. It uses CSS to provide the control over colours and highlighting/outlining of the structure. Described in XML means that additional information (Metadata) is available and content can be portable. Hence, it is possible to mix SMIL with SVG. Stylesheets in XML gives the user control over rendering equivalents (e.g. text), layout and styling [29].

- Synchronized Multimedia Integration Language (SMIL)
  SMIL is a XML-based language and it is recommended by the W3C. It works for a media player in a similar way as HTML does for a web browser. More recently, the current release of SMIL 3.0 objectives include the reuse of its syntax and semantics in other similar types (particularly those who requires synchronization and timing representation) of XML-based languages. This will allow the integration of SMIL timing into XHTML and SVG [13]. Since this is another form of web technology on the user's end (Client-side) that conforms to the WAI guidelines, it will be interesting to analyze how well it has been accepted by web content authors and developers.

Appendix C

Client-side Scripting

Client-side scripting is commonly employed to reduce the work load from the server, and enables dynamic control on client-side. Due to the popularity of these technologies, they are important to be included in our web evolution analysis. Although there are a few client-sides scripting language available, only JavaScript and VBScripts will be discussed.

JavaScript is a client-side web development scripting language that is dynamic, weakly typed, prototyped-based language (an object-oriented programming which classes are not present) with first-class functions. Its plugin comes along with the most popular web browsers, but due to this its performance may vary slightly across the different types of web browsers.

Visual Basic Scripting (VBScript) is a client-side Active Scripting language developed by Microsoft. Commonly it can only be executed when run by Microsoft Internet Explorer. Other popular web browsers such as Firefox and Opera do not have built-in support for VBScripts hence its competitor (JavaScript) is normally a more popular choice.

Appendix D

Guidelines

Some recommended guidelines have been created by the W3C to explain how to make web content accessible to people with disabilities. These guidelines are intended for developers of web content and authoring tools, and web content authors. The W3C has designed [17] as a useful source of reference for accessibility principles and design ideas.

The Web Accessibility Initiative (WAI) is a good point to begin with. It is one of the four Domains within the W3C that develops web accessibility solutions for all disabilities that affect access to the Web, these includes visual, auditory, physical, speech, cognitive, and neurological disabilities [44]. Due to the recent technological advancement, the web has been transforming our society, and it is increasing becoming an important source for many aspects of life, companies and organisations [41, 2]. A series of accessibility standards and guidelines were developed by WAI, some of these include WCAG, UAAG, WAI-ARIA, ATAG and accessibility for specific technologies.

In order to allow true accessibility to the web for people with disabilities, having an accessible web content is not enough. It is also important to allow disabled web content authors to have an accessible user interface on these authoring tools. However we will only be covering guidelines relating to web content accessibility.

D.0.1 Web Content Accessibility Guidelines (WCAG)

The Web Content Accessibility Guidelines (WCAG) [14] covers the guidelines for website designers and web content authors so that their websites are accessible to people with disabilities. In order to realise this, components such as user agents,

contents, assistive technologies, users knowledge, developers and evaluation tools are equally important. The guidelines follow the four principles of accessibility; perceivable, operable, understandable, and robust. When all these layers are in place, it will make web content more accessible. Currently, the stable version of WCAG is in version 1.0, however, a newer version (2.0) is presently being developed. This new version of WCAG can be applied to more web technologies and the technologies of the future [14]. Although the set of guidelines (WCAG 2.0) is still in its working draft version, web pages that show signs of efforts to conform will definitely benefit for having a wider web audience.

A set of level conformance has been set in WCAG to meet the needs of different groups and different situations. Basically, there are three levels of conformance a web page can be achieved: A (lowest), AA, AAA (highest). Each guideline will go through a number testable requirements and conformance testing, and the level of conformance will be awarded according to the results attained [14].

From the research point of view, since WCAG is the core guidelines for web content accessibility, identifying or analysing its adoption trends can help to provide as recommendations for future web related HCI work.

D.0.2 Accessible Rich Internet Applications Suite (WAI-ARIA)

Recent advancements (e.g. AJAX) saw content on the web becoming more dynamic. This was partly influenced by the vision of Web 2.0 to make a participative, and read/writeable web [33]. Web applications (especially when they are complex) often become inaccessible to people with disabilities when assistive technologies fail to determine the semantics behind the web content. In order to cope with this, in 2008 the W3C has included the current draft version of WAI-ARIA as part of WAI guidelines (see section 2.4). It describes how to make Web content, and Web applications with dynamic content (i.e. advanced user interface controls developed with AJAX, HTML, JavaScript and related technologies) more accessible, as well as better usability of web resources for people with disabilities without extensive modification to existing libraries of web resources [43].