Information Technology Journal 3 (2): 211-218, 2004
ISSN 1682-6027

Understanding and Reducing Web Delays

Kevin Curran and Connor Duffy
School of Computing and Mathematical Sciences, University of Ulster,
Magee Campus, Northern Ireland, BT47 3QL, UK

Abstract: This study investigates download times associated with a web request, identifies where delays occur, and provides guidelines which can be followed by web developers to enable a faster and more efficient download and service for their users.

Key words: WWW, web page delay, internet performance, quality of service, internet delay

INTRODUCTION

Internet statistics from NUA[1] imply with an "educated guess" that there are now over 605 million users on the Internet world wide. If only 5% of these users were on at any one time that still means there are 30 million people online. This will clearly lead to great delays which are a major factor affecting the performance of a site and ultimately a sites success[2]. This mostly affects small companies on the net because most of the major players have already established a name for themselves and delays most likely would not put off their customers. This is why it so important for the smaller companies to make every effort to ensure their site is of a high quality and download times are kept to a minimum.

When a user launches a browser and requests an action to be performed the browser interprets the request. It sends information to the appropriate site server where the requested information is stored and this site server sends the information back. The Internet is actually a packet switching network which sends requests via packets of data (datagram's). Each packet contains the IP address of sender and receiver, and the information being requested. On any one request there can be more than one packet. This is because each packet is of a fixed size and some requests may need more than this. This means the request must be broken up into the appropriate number of packets. The route taken to obtain the requested information depends on the sender's geographic location and that of the receivers. If there are a lot of packets along a certain route they will all be queued or find a different route until they reach their destination[3]. The destination cannot send any information until all the associated packets have been received. When all the packets have been received the destination sends the requested information back in packets via routers to the sender. Again the same problem arises in relation to routes and bandwidth[2]. The Transmission Control Protocol (TCP) is a communication protocol which enables two hosts to establish a connection and exchange streams of data[4]. It sends information to a client/server and must receive a response or it will send the information again. The client TCP breaks down the information into smaller packets and numbers it. The server TCP then reconstructs it so the original data can be viewed. When a user types in a URL into the address box and presses "go" a DNS lookup is performed. When the DNS lookup is complete the client connects to the server via a TCP connection. The request is broken down into equal sized packets. Each packet, containing the IP address of the destination server, is then passed through the internet via routers. Each router resolves the next router in line using a routing algorithm. This process is repeated by every router until the destination server is reached with the request. The server then responds to the requesting client in the same way until the request has been fulfilled[5]. Between the server and the client is the most likely place for a congestion to occur. This is why it is so important that the size of the information the user has requested should be as small as possible.

Image compression: When placing images on the internet it is important that they are first optimised. This will do two things, compress it in size and ensure its colours are web safe (can be displayed properly). Compressing an image gets rid of redundant data from the image. Without compression, images remain large and are quite often 'bottlenecks' taking too long to download. For example, if a company is selling products online and they have not compressed their images the extra download time required can alienate a user and cause the company who owns the website great problems. There needs to be a trade off

Corresponding Author: Kevin Curran, School of Computing and Mathematical Sciences, University of Ulster, Magee Campus, Northern Ireland, BT47 3QL, UK Email: kj.curran@ulster.ac.uk

between quality and download time if the company is to succeed. This can also indirectly affect other web users who may not even be visiting this website as large images may cause congestion along the channels their packets are traveling.

Caching and its architectures: Caching is a technique used to store popular documents closer to the user. It uses algorithms to predict user's needs for specific documents and stores documents it deems important. Caching can occur anywhere within a network, on the user's computer, at an ISP or at a server. Many companies also use web proxy caches to display frequently accessed pages to their employees. This can reduce the bandwidth the company requires which in turn can lower costs[6]. Another example of caching can be seen on Google, one of the worlds most popular search engines, which uses caching to provide their users with the documents they need much faster than without caching (sometimes in under half a second). Web-cache performance is directly proportional to the size of the client community[2]. The larger the client community the greater the possibility of cached data being requested and therefore the better the cache's performance. The main drawback with caching is that in most cases cached documents may never be viewed again. Caching a document can also cause other problems. Most documents on the internet change over time as they are updated. Static and Dynamic caching are two different technologies which use caching to combat this problem in order to reduce download time and congestion. Static caching stores the content of a webpage which does not change. There is no need to repeatedly request the same information over and over again as this just wastes time. This is an excellent approach to combat congestion. Dynamic Caching is slightly different. It determines whether the content of a page has changed by checking for updates. If the contents have changed, it will then store the updated version[7]. This unfortunately can lead to congestion and so is possibly not a very good approach as it does require checks to be performed on the source of the data to establish whether or not an update is necessary. If used properly these two technologies can work together to reduce latency and congestion. Prefetching is an intelligent technique used to reduce perceived congestion. It tries to predict the next page or document a user might want to access[8]. For example, if a user is on a page with many links the Prefetching algorithm will predict that the user may want to view associated links within that page. The Prefetcher will then request the pages it predicts the user will try to access next and stores them until the user does actually request them. It will then display the page significantly faster than if the user requested the page without Prefetching. The only real drawback is that if the user does not request the pages the Prefetcher algorithm retrieves, congestion may have been caused needlessly.

Evaluation of page download delay: This study analysed the performance of 35 sites on the Internet; 10 US sites, 10 UK sites, 10 other worldwide sites and 5 Northern Ireland sites using methodical tests. The study investigated the effects of Internet congestion by analysing site download time, site content and other technologies used by 35 websites worldwide. Analysing these sites involved timing the sites homepage download time from initial request until their transfer is complete. The reason for only choosing the homepage was that this should be representative of the whole site. Three types of request are used:

Normal request: No information has been requested from the site before. Therefore the computer will have no files stored from the site. (The cached files and histories were cleared hourly).

Using popup filter: This blocks any popup windows, such as advertisements and other links, which may hinder the performance of the requested webpage and thus decrease the download time of the site. The software used was Filter Gate 4.0.

Cached request: After the normal request the computer will have cached files from the site. This should mean the computer will process this request faster by displaying the sites static content faster. However, any dynamic content will have to be retrieved from the server.

This was done hourly for a week. The next part of the research involved analysing the content of the sites. This meant examining the sites content, from images to text. The images were checked for size and use of compression. All the images are compressed, using Adobe Photoshop 7.01, where possible and compared to the original image to see if appropriate compression was used. A new size was only recorded if the size was decreased and the quality of the image was not reduced. All the results were recorded and analysed to identify why delays were occurring and how delays could be avoided. Additional information was also gathered about the location, using NeoTrace Pro 3.25 of the server where the sites are stored.

Below is a subset of sites this research used in testing. Examples of sites in the USA were ABC (www.abc.com), AOL (www.aol.com), and Google (www.google.com). Sites in the UK included the BBC (www.bbc.co.uk), Dabs (www.dabs.com) and Streets

Online (www.streetsonline.co.uk). Other worldwide sites included Australia (www.csu.edu.au), China (www.edu.cn), Japan (www.nikkei.co.jp). Local sites included Kainos (www.kainos.co.uk), Singularity (www.singularity.co.uk) and Ulsterbus (www.ulsterbus.co.uk). Tests were carried out hourly over a period of 7 days. Each of the 35 websites were tested using a normal request (no cache or popup filter), with cache enabled, and with the popup filter. One of the most important points was to carry out the tests in a consistent manner. The computers cache was checked to make sure it only worked as and when it was required, and the popup filter was turned on and off as appropriate. This means that all the results are accurate. All the information for the 35 sites was recorded. The results have been broken down into categories to help examine the main findings more closely. The categories are: Normal downloads vs web cache downloads, Normal downloads vs popup filter downloads, Download time vs image size, Download time vs total site size, Original image vs compressed image, Total site before compression vs after compression and Download time vs distance. Table 1 is a cross reference for all the graphs used to display the results.

Normal vs web cache downloads: Next we examine the performance of Web cached downloads against normal downloads to see what effects, if any, a web cache has on the download time. This will give an insight into caching and help to identify design techniques for web developers.

Table 1: Reference key for results

Web sites	Abbreviation
8over8.com=	8ov
abc.com=	abc
amd.com=	amd
americanmail.com=	ame
aol.com=	aol
bahn.de=	bah
bbc.co.uk=	bbc
bet365.com=	bet
buyagift.co.uk=	buy
clarendonmanor.co.uk=	cla
csu.eudu.au=	csu
dabs.com=	dab
danskirsk.dk=	dan
edu.cn=	edu
google.com=	goo
intel.com=	int
kainos.co.uk=	kai
kelkoo.co.uk=	kel
louvre.fr=	lou
martins-seafresh.co.uk=	mar
mlbapparel.com=	mlb
navyseals.com=	nav
nic.it=	nic
nikkei.co.jp=	nik
questionmarket.com=	que
sevengatesdesigns.com=	sev
singularity.co.uk=	sin
streetsonline.co.uk=	str
ulsterbus.co.uk=	uls
unam.mx=	una
yahoo.com=	yah
wtr.ru=	wtr

Figure 1 displays the average of the normal downloads against the average of the web cache for the UK. The majority of the sites have significantly reduced their download times; however, two of the sites have not. These are www.8over8.com (local) and www.dabs.com (UK). On analysis of the content of these sites it is not immediately obvious why they have not performed well. They both have a smaller number of images, a smaller total size of images and a smaller HTML file size than some other sites which have performed better. 8over8 actually has a low normal download time and the web cache time is also low. Dabs, however, was slightly higher than most other sites. This may be because it has 56 images on the site which take longer to process.

The results of Fig. 2 for the rest of the world are much the same as those found for the UK. The web cache appears to improve the download time. Only three of these twenty sites have not actually performed as well as the others. These were www.edu.cn, www.intel.com and www.nikkei.co.jp and this is only because their original download times were quite high. Interestingly, the sites which performed the best from both the UK and rest of world when using the web cache were sites which had mostly images and very small HTML files with respect to the other sites. These sites were www.clarendonmanor.co.uk, www.google.com, www.nic.it, www.sevengatesdesigns.com and www.wtr.ru.

It was also interesting to see that the fastest download for the web cache was www.clarendonmanor.co.uk which had the largest size of images. Also this site has no text to display. The HTML for this site has only tags to display the images. The other three sites have some text to display but not a lot when compared to the other websites which were used in this paper. Of these five sites, www.google.com is the slowest to display when cached (Table 2). This could be because the HTML file is slightly larger than all but www.sevengatesdesign.com, and also the code required to carry out the search for the search engine is a bit more complex. These findings do seem to indicated that images play a major role in download times.

Normal download vs popup filter: The results here were very interesting. The popup filter had no visible effect on decreasing the download time. It did however noticeably increase the download time for www.buyagift.co.uk. This is perhaps because this site was displayed in frames and the popup filter may have viewed the HTML documents

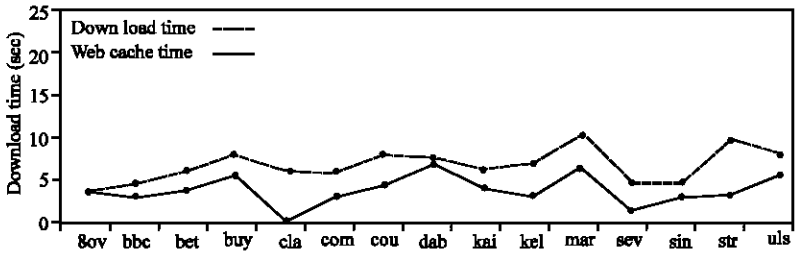

Fig. 1: UK-normal download vs web cache

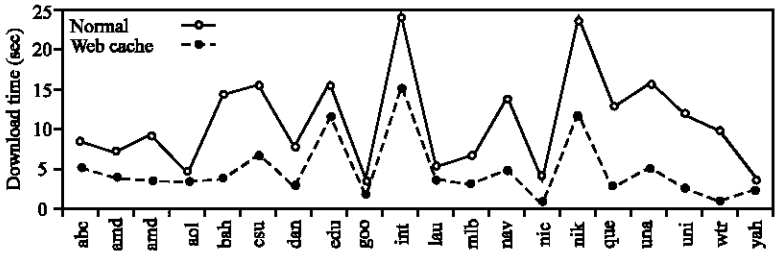

Fig. 2: Rest of world-normal download vs web cache

Table 2: Best performance using cache

Web sites	No. of images	Size of images	HTML File	Cached time
Clarendonmanor.co.uk	6	82.50	2.93	0.17
google.com	1	8.35	3.67	1.65
nic.it	6	21.30	2.40	0.55
sevengatesdesigns.com	5	38.90	4.10	1.35
wtr.ru	3	20.50	1.80	0.88

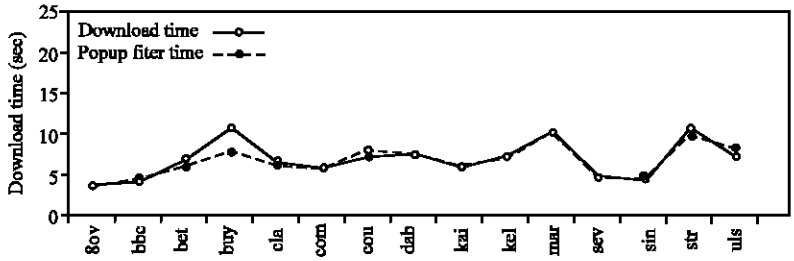

Fig. 3: UK-normal download vs popup filter

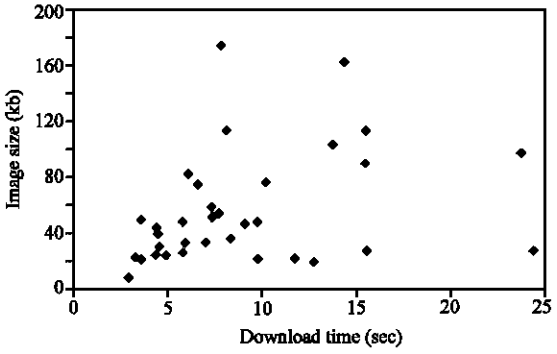

Fig. 4: Download time versus image size

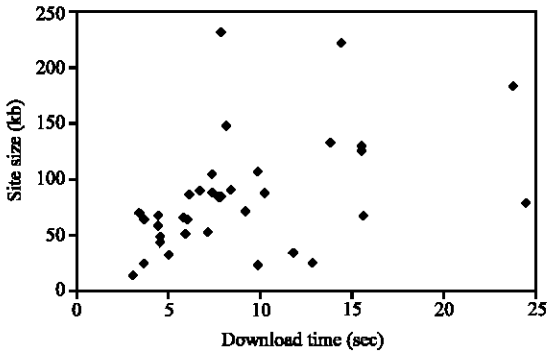

Fig. 5: Total site size vs download time

being displayed in these frames as popup windows. The results from the worldwide sites were the same - the popup filter does not have much effect on download time (Fig. 3). It even had an effect of increasing download time for one of the sites.

Download time vs image size: This is important as it will indicate the importance of image use on sites. All images were of the JPEG format.

Figure 4 displays all the results for download time against image size. It is apparent from these results that download time is effected by image size. There is a positive gradient curve which represents this. Figure 5 displays the total site size against download time. Again there is a positive gradient curve present which indicates that as site content increases so too does download time. There are also a few anomalies present but again this was expected. These anomalies are exactly the same anomalies that were identified (Fig. 4) and have already been proved insignificant.

Original image vs compressed image: This is a very important section as it aims to highlight the importance of image compression. It should provide details about the use of image compression.

Figure 6 displays the results for the UK sites of original image sizes for each site against compressed image sizes. There is an obvious pattern emerging. The larger the image content of each site the greater the compression achieved. Five sites here have revealed the use of good image compression. These are www.bbc.co.uk, www.bet365.com, www.dabs.com, www.kainos.co.uk and www.singularity.co.uk. The trend was also similar for the worldwide sites. Out of those - twenty sites four performed well. These were www.google.com, www.louvre.fr, www.unisa.ac.za and www.yahoo.com.

Site content before compression vs after compression: This section looks again at the effects of image compression. This time the whole content of each site has been graphed to see how the compressed images would change the site size. This is significant as it has already been shown that download time is effected by size of both image size and overall site content (Fig. 7).

Figure 7 shows the total site content , including images and html files, for the UK before and after the compression of the images. These results further prove that the greater the site content the higher the compression ratio achieved. This means out of this sample of fifteen sites from the UK, seven have performed well for the use of compression on images. The trend was the same for the worldwide sites.

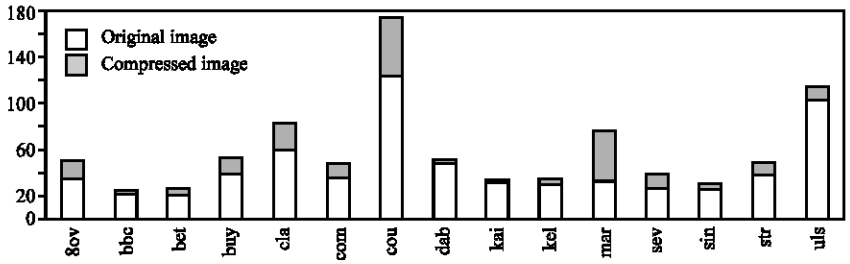

Fig. 6: UK-origional image versus compressed image

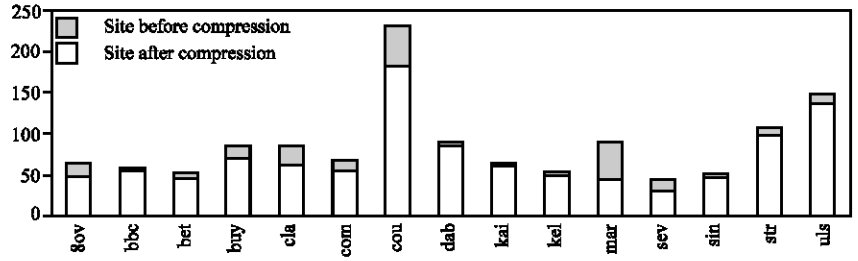

Fig. 7: UK-site before vs after compression

RESULTS

The overall performance of the present study for each site in the categories mentioned must be taken into consideration. All the results indicated the importance of image size and compression over download time. Appropriate categories have been chosen to evaluate the results and identify the best and worst sites.

Best performing sites: Three categories will be used to identify the best sites. These categories are download times (Normal), cacheability and image compression. The best sites will be those which have performed well in at least two of these categories. This is because sites may achieve good results from one category and not in others. The download times for all the samples varies from 3-25 sec. In order to decide which sites were the best for download time a criteria must be set. The criteria was 0-5 sec = Excellent, 5- 10 sec = Good, 10-12 sec= Average and 12-25 sec = Poor. For a site to be classed as the best it must fit into the excellent range (0-5 sec). Table 3 displays the download times for all the sites.

Table 3: Download times for all sample sites

Web sites	Download time (Sec)
8over8.com	3.621142857
kainos.co.uk	5.961571429
abc.com	8.403285714
kelkoo.co.uk	7.060571429
amd.com	7.301
louvre.fr	4.932714286
americanmail.com	9.160285714
martins-seafresh.co.uk	10.175
aol.com	4.442285714
mlbapparel.com	6.767714286
bahn.de	14.41757143
navyseals.com	13.77457143
bbc.co.uk	4.354571429
nic.it	3.604857143
bet365.com	5.897571429
nikkei.co.jp	23.79514286
buyagift.co.uk	7.729571429
Questionmarket.com	12.77242857
clarendonmanor.co.uk	6.113857143
sevengatesdesigns.com	4.487
com3.org	5.822857143
singularity.co.uk	4.539857143
countrybookshop.co.uk	7.867428571
streetsonline.co.uk	9.838142857
csu.edu.au	15.48971429
ulsterbus.co.uk	8.131142857
dabs.com	7.349428571
unam.mx	15.49042857
danskirsk.dk	7.648571429
unisa.ac.za	11.75942857
intel.com	24.45642857
wtr.ru	9.843142857
google.com	2.970428571
yahoo.com	3.363571429

Nine sites were identified as excellent. These were www.8over8.com, www.aol.com, www.bbc.co.uk, www.google.com, www.louvre.fr, www.nic.it, www.yahoo.com, www.sevengatesdesigns.com and www.singularity.co.uk. Another important measure was caching. Many users will revisit the same sites time and time again. Computers can store files from various sits on their internal cache for reuse when a user requests information from the same location. This paper discovered that a web cache can notably decrease the download time for many sites. Five sites were identified that performed very well when a cache was used. These sites were www.clarendonmanor.co.uk, www.wtr.ru, www.google.com, www.nic.it and www.sevengatesdesigns.com .

An important point is that sites cannot be judged by their size as the content of the site. However, they can be judged on their use of image compression. The use of image compression and its effects on overall site content have been analysed. The sites which performed the best are those which achieved a low overall difference in site size when comparing size before compression with after compression. Twelve sites were identified that fit into this category. They are: www.8over8.com, www.bbc.co.uk, www.bet365.com, www.com3.org, www.dabs.com, www.edu.cn, www.google.com, www.kainos.co.uk, www.louvre.fr, www.singularity.co.uk, www.unisa.ac.za and www.yahoo.com.

The overall best sites are those which appear in two or more of the above categories. Therefore the best sites from this sample are www.8over8.com, www.bbc.co.uk, www.google.com, www.louvre.fr, www.nic.it, www.singularity.co.uk, www.sevengatesdesigns.com and www.yahoo.com. This means eight sites out of thirty-five sites, from this sample, are well developed, with download times and users in mind. This is only 23%. The overall best site was www.google.com. It obvious that the key here is simplicity. Google has 1 small image and a small HTML file to display the site. The techniques used by these sites are apparent. There was a good use of compression. This is important as it depends on the actual image being used. A large image with lots of different colours will not display as well as a smaller image with a small range of colours. There were also no bulky special effects and small HTML files.

Worst performing sites: The sites that caching has the poorest effect on were those which have not significantly improved from their original download times and also had a normal download time outside the excellent criteria. These were www.dabs.com , www.edu.cn, www.intel.com and www.nikkei.co.jp. Again these sites will not be judged on size but their use of image compression. There were eight sites which fell into this category for bad use of compression. These were www.bahn.de, www.countrybookshop.co.uk, www.martins-seafresh.co.uk, www.navyseals.com, www.nic.it,

www.nikkei.co.jp and www.wtr.ru. The overall worst sites are those which fall into two or more of the above categories were www.bahn.de, www.edu.cn, www.intel.com, www.navyseals.com, www.nikkei.co.jp and www.unam.mx. This means six sites out of thirty-five (17%), from this sample, are badly developed. The main characteristics of the worst sites were large images, bad code. Some included special effects which increases size and slows performance. Another important point to realise at this stage is that although a site may be strong in one area it may be weak in others. For example, www.edu.cn appears in the best category for Image compression but it appears in two of the worst categories for Download Time and Cacheability.

The effect of HTML editors on page size: It is known that inefficient coding can result in a delay at the client side reconstruction of a site. This is true of HTML. Many developers now-a-days use timesaving software which can write HTML code for them. Some of these include Dreamweaver and FrontPage. Alternatively they can use Notepad where they will have to write the code themselves. A question which must be asked is how efficient these automatic web development software products are in comparison to manual coding. It has been discovered that there are differences in the amount and size of code each of these produce. For example, we produced various web pages using these three pieces of software and discovered that there was a difference in the file sizes of each. Notepad produced the smallest file followed by Dreamweaver and then FrontPage (Table 4).

Table 4: HTML editors

	Site 1	Site 2	Site 3
Notepad	268 bytes	669 bytes	1.48 kb
Dreamweaver	548 bytes	998 bytes	1.67 kb
FrontPage	640 bytes	1.9 kb	1.73 kb

This means that both Dreamweaver and FrontPage produced redundant code and resulted in a larger file than was needed. Despite the fact that this research was only carried out on a small scale the results are notable. If more research was carried out on larger sites developed from these types of software products it may be found that the problem is of a greater order of magnitude (i.e. larger sites means larger discrepancies and more redundant code). This area is significantly important now as people and technology are moving away from Desktop interfaces and towards mobile devices such as PDA's.

Page design guidelines: This section details guidelines, identified from the findings of this study, for web developers on how to optimise websites for transfer over the internet. Following these guidelines will result in smaller, more efficient file content which will both reduce bandwidth required to transfer a site from a server to a client and also reduce server side reconstruction time. This will therefore reduce the download time observed by the user. The guidelines are as follows:

- All images, irrespective of original size, should be compressed. Image simplicity is also important. Avoid using too many colours. Using a small number of colours will result in a much higher compression than an image with a large range of colours. There are tradeoffs between quality and speed. This is up to the developers own judgement.
- Keep content static where possible as dynamic content will cause a delay while a cache retrieves the dynamic data from the sites server
- Use technologies which will decrease the download time and the effort required for client side reconstruction. We tested how DHTML compared in size to a Flash file for a websites Introduction. It was discovered that using DHTML code instead of inserting a Flash file into a website decreased the size of the data required to produce exactly the same effect.
- The DHTML reduces effort required in developing a complicated Flash file (Table 5) and also lowers the size and therefore reduces the bandwidth required for transferring the file, thus decreasing download time. Also because it has been written in HTML it will start immediately and not require the user's browser to load a Flash file.

Table 5: DHTML vs Flash

DHTML	Flash and HTML
	Flash file size= 4.31 kb
Size= 2.98 kb	HTML file size= 0.94 kb
	Total of two files= 5.25 kb

- Avoid the use of background images where possible and opt for Hex colour codes instead. Background images increase site size and take longer to reconstruct than hex codes.
- Locate server close to target market. A host server may offer inexpensive domain hosting but this may mean it takes longer for a site to reach its targeted users.
- If any code has been written using automatic coding software it should be checked for redundant tags and other excess code, which increase site size and therefore download time.

CONCLUSIONS

It was apparent from the results obtained from this research that delays in Internet traffic and congestion can often times be attributed to poorly developed sites. There

is a direct correlation between site size and download time at times thus web developers must be careful when choosing images to place on websites as the findings of this paper identified that image size has a considerable effect on a sites download time. Therefore image size is an important contributing factor to Internet traffic congestion. This study has shown that 66% of the sites sampled either did not use compression tools at all, or, did not use them properly. This was only a random sample of a small number of sites. If further research was carried out it may be discovered that this problem is worse. An interesting and important finding from the research was the fact web caches store images more readily than text. Another interesting finding from the research was that popup filters have no effect on download time. This means internet browsers can display multiple windows which have no effect on one another.

REFERENCES

1. NUA, Internet Trends and Statistics, "How Many Online?" http://www.nua.ie/surveys/ how_many_ online/
2. Saiedian, M. and M. Naeem, 2001. Understanding and Reducing Web Delays. IEEE Computer Journal, Vol. 34, No. 12, December, pp: 30-37.
3. Idaho State University, Certified Internet Webmaster, History of the Internet- Brief Timeline. http://www.homelandweb.com/ciw/history.htm
4. Xilinx, Transmission Control Protocol/Internet Protocol (TCP/IP), Taken from: http://www.xilinx. com/esp/optical/net_tech/tcp.htm
5. Franklin, Curt, How Routers Work, HowStuffWorks.com, http://www.howstuffworks. com/router.htm
6. Caching Tutorial for Web Authors http://www.web-caching.com/mnot_tutorial/intro.html
7. Dynamic Caching, foedero.com, http://www. foedero.com/dynamicCaching.html
8. Fan, L., Q. Jacobson, P. Cao and W. Lin, 1999. Web Prefetching Between Low-bandwidth Clients and Proxies: Potential and Performance, ACM SIGMETRICS Conference, Georgia, USA, pp: 178-187.