

HIGH QUALITY DESIGN AND METHODOLOGY ASPECTS TO ENHANCE LARGE SCALE WEB SERVICES

Suryakant B Patil¹, Sachin Chavan², Preeti Patil³

¹Department of Information Technology, MPSTME, NMIMS, Shirpur, India.

^{2&3}Department of Computer Engineering, MPSTME, NMIMS, Shirpur, India.

ABSTRACT

Network traffic has increased tremendously due to the rise in the number of Internet users. This has affected several aspects and characteristics of large scale networks such as reduced network bandwidth, increased latency, and higher response time for users who require large scale web services. This paper proposes a novel design and methodology to address these issues. It does this by optimizing network bandwidth and reduces latency and response time for large scale Web applications. The methodology works by analysing content in the proxy cache, identifying content aliasing, duplicate suppression and by the creation of the respective soft links. Here it is necessary to understand the characteristics of the web traffic before one would propose any design to manage it better. For instance, an expanding network results into reduced network bandwidth, increased latency, and higher response time for users. The present solution makes intelligent use of the proxy cache server to overcome these problems. In this study proxies were designed to enable network administrators to control internet access from within intranet. But when proxy cache is used, there develops the problem of Aliasing. Aliasing in proxy server caches occurs when the same content is stored in the cache several times. The present methodology improves performance by avoiding storing the same content in cache multiple times those results in wastage of storage space. The investigation proposed here analysed the cache content of the proxy and checked the replication of content. This work thus proved the way to increase the efficiency of large scale web services. This high quality design approach focused on the content of the access logs and user habits as users would access large scale Web services.

KEYWORDS: Large Scale Web, Cache, Web Proxy, Mirroring, and Duplicate Suppression.

I. INTRODUCTION

Web caching consists of storing most frequently access objects to the local server at our place instead of original server. Due to this reason web server can make better use of network bandwidth, reduced workload on web servers, improves the response time for users. The local servers used to store frequently referred objects are called as web proxy servers. A proxy server acts as a mediator between the original server and the clients. The proxy server setup can be seen in Figure 1. The proxy cache also stores all of the images and sub files for the visited pages, so if the user jumps to a new page within the same site that can be possible. Aliasing in proxy servers caches occurs due to same content is stored in cache multiple times. On the World Wide Web, aliasing commonly occurs when a client makes two requests, and both the requests have the same payload [3]. The major problem associated with using the web cache is storage space requires storing the visited pages with their objects [16].

This Paper is divided into 4 sections; section 1 is an introduction of proxy web cache system which briefs about the role of the Proxy server and setup of the proxy cache server between the client side browser and web servers. Further it list out the advantages and disadvantages of using Proxy cache. Section 2 covers the related work, the concepts of the web traffic, web cache, mirroring, MD5 Hashing Algorithm followed by static caching and dynamic caching are discussed. Section 3 elaborates the survey done for Data collection and reduction with experimental analysis. The data is gathered from various systems based on the user habits and requirement. It reflects the amount of the

cache space saved for respective systems after reduction based on the category of the files. Section 4 consists of results and discussions based on experimental analysis. It shows the enhancement in the large scale web services by using the proxy cache for duplication avoidance with the high quality design and methodology aspects for various categories of the web data like image, HTML, formatted, Style and other files.

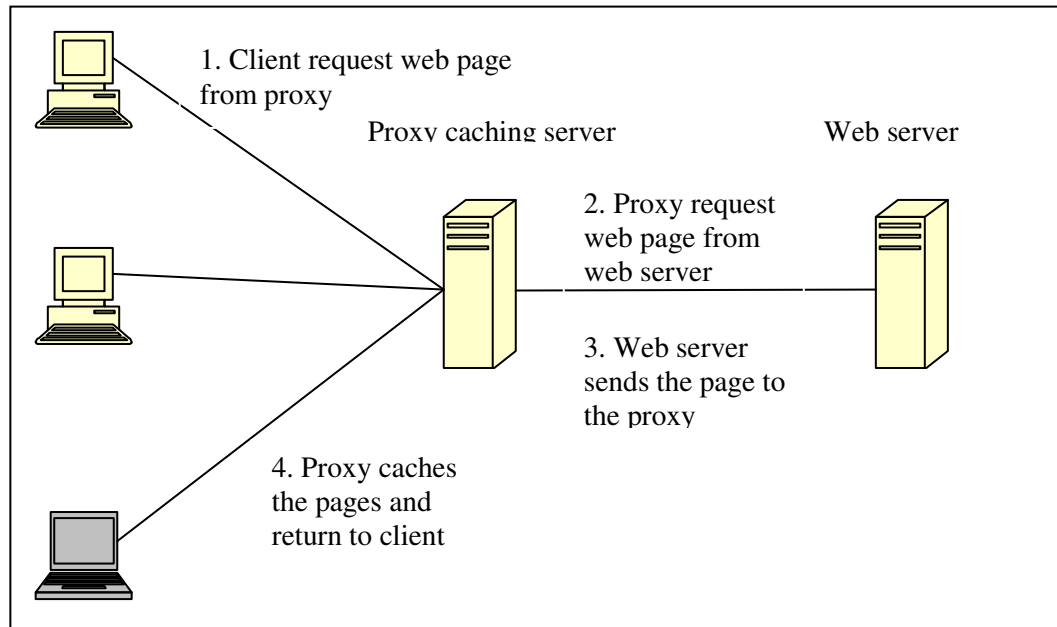


Figure 1. Proxy server setup

1.1 Advantages of Caching

1. Web caching reduces the workload of the remote Web server
2. If the remote server is not available due to the remote server's crash or network partitioning, the client can obtain a cached copy at the proxy.
3. It provides us a chance to analyze an organization usage patterns.

1.2 Disadvantages of using a caching:

1. The main disadvantage is that a client might be looking at stale data due to the lack of proper proxy updating.
2. The access latency may increase in the case of a cache miss due to the extra proxy processing.
3. A single proxy cache is always a bottleneck.
4. A single proxy is a single point of failure.

II. RELATED WORK

2.1 Mirroring

Shivkumar and Garcia-Molina investigated mirroring in a large crawler data set and reported that in the WebTV client trace far more aliasing happens than expected [11]. Similarly, Bharat et al. surveyed techniques for identifying mirrors on the Internet [17]. Bharat and Broder investigated mirroring in a large crawler data set and reported that roughly 10% of popular hosts are mirrored to some extent [17]. Broder et al. considered approximate mirroring or “syntactic similarity” [17]. Mirroring is defined as keeping multiple copies of the content of a Web site or Web pages on different servers using different domain names. A mirrored site is nothing but an exact replica of the original site and is updated frequently to ensure that it reflects the updates made to the content of the original site. The main purpose of mirroring is to build in redundancy and ensure high availability of web

documents or objects. Mirrored sites also help make access faster when the original site is geographically distant.

2.2 Web Traffic

The amount of data sent and received by visitors to a website is web traffic. It is analysis to see the popularity of web sites and individual pages or sections within a site. Web traffic can be analyzed by viewing the traffic statistics found in the web server log file, an automatically generated list of all the pages served.

Traffic analysis is conducted using access logs from web proxy server. Each entry in access logs records the URL of document being requested, date and time of the request, the name of the client host making the request, number of bytes returned to requesting client, and information that describe how the client's request was treated as proxy [1].

Processing these log entries can produce useful summary statistics about workload volume, document type and sizes, popularity of document and proxy cache performance [7].

2.3 Web cache

Web caching can play a valuable role in improving service quality for a large range of Internet users. A web cache is a mechanism for the temporary storage (caching) of web documents, such as HTML pages and images, to reduce bandwidth usage, server load, and perceived lag. A web cache stores copies of documents passing through it; subsequent requests may be satisfied from the cache if certain conditions are met. There are two types of Web caches a browser cache and a proxy cache [9].

A proxy cache is a shared network device that can undertake Web transactions on behalf of a client, and, like the browser, the proxy cache stores the content. Subsequent requests for this content, by this or any other client of the cache will trigger the cache to deliver the locally stored copy of the content, avoiding a repeat of the download from the original content source [6].

A client, such as a web browser, can store web content for reuse. For example, if the back button is pressed, the local cached version of a page may be displayed instead of a new request being sent to the web server. A web proxy sitting between the client and the server can evaluate HTTP headers and choose to store web content. A content delivery network can retain copies of web content at various points throughout a network. Performance of web cache system is measured in terms of utilization, bandwidth, efficiency, hits, throughput latency, availability and response time [14].

2.4 Static Caching

It is a new approach of web caching which uses yesterday's log to predict the today's user request. Static caching is performed only once in a day. It is simpler approach to use and requires very low CPU overhead. It improves cache performance by using compression techniques. Static web pages can be compressed to reduce their size and free up the cache space. Cache server performance typically relies on two major factors first is byte hit ratio and second one is hit ratio. Hit ratio is nothing but the percentage of all accesses that are fulfilled by data in cache; while, byte hit ratio is the hit rate with respect to the total number of bytes in the cache [8].

The static caching algorithm defines a fixed set of URLs by analyzing the logs of previous periods. It then calculates the value of the unique URL. Depending on the value, URLs are arranged in the descending order, and the URL with the highest value is selected. This set of URLs is known as the working set. When a user requests a document and the document is present in the working set, the request is fulfilled from the cache. Otherwise, the user request is fulfilled from the origin server [8].

2.5 Dynamic Caching

In contrast to dynamic caching where cached documents are updated more than once a day. As mentioned before, the benefit of current Web caching schemes is limited by the fact that only a fraction of web data is cacheable. Non-cacheable data is of a significant percentage of the total data. For example, measurement results show that 30% of user requests contain cookies.

Dynamic caching is more complex than static caching and requires detailed knowledge of the application. One must consider the candidates for dynamic caching carefully since, by its very nature,

dynamically generated content can be different based on the state of the application. Therefore, it is important to consider under what conditions dynamically generated content can be cached returning the correct response. This requires knowledge of the application, its possible states, and other data, such as parameters that ensure the dynamic data is generated in a deterministic manner [5].

III. DATA COLLECTION, REDUCTION AND EXPERIMENTAL ANALYSIS

3.1 Document Types

While downloading files and viewing websites, you'll meet up with many file formats. Most are common, and encountered frequently, others are rarer and require specialist programs to open or use [2]. Here, the categorized list of the documents and files are given below

- HTML: “.html”, “.shtml”, “.htm”, and “.map”.
- Image: “.gif”, “.jpeg”, “.gif89”, “.jpg”, “.png”, “.bmp”, “.jpg”, “.pcx”, “.rgb”, “.tif”, and “.ras”.
- Audio: “.au”, “.aiff”, “.aif”, “.aifc”, “.mid”, “.snd”, “.wav”, and “.lha”.
- Video: “.mov”, “.qt”, “.avi”, “.mpe”, “.movie”, “.mpeg”, “.mpg”, “.mp2”, and “.mp3”.
- Compressed: “.z”, “.gz”, “.zip”, and “.zoo”.
- Formatted: “.ps”, “.pdf”, “.dvi”, “.ppt”, “.tex”, “.rtf”, “.src”, “.doc” and “.wsrc”.
- Dynamic: “.cgi”, “.pl”, and “.perl”.

When cache is use to store the object then the URL of web page is going to be cache in the cache memory, as the URL is stored the objects associated with the Page such as images, Audio, Video files are also stored at the same place for quick retrieval of that page for next time [10].

3.2 Changing of proxy server

In most of the organization's or institution server does not support the proxy cache, so it is difficult to use main server as cache server so we have to change the proxy server from main server to other server [4].

Following are the steps to switch machine to other proxy:

1. Open the browser for ex. Internet Explorer
2. In internet explorer pull down the **Tools** menu and click **Internet Options...**
3. Click the **Connections tab**:
4. click the **LAN Settings...** button:
5. In the Address: box change "proxy1 Address" to "proxy2 Address" or vice versa and click **OK**.
6. Click **OK** on the Internet Options dialogue box to get back to the browser screen and you will now be able to get external sites.

3.3 Duplication of Data

Duplication of data means storing the multiple copies of same data object. In case of cache when we cache the object or the webpage that web page is stored at cache memory but when the different users request the same page then the multiple copies of that object or web page is stored at cache memory which results in the wastage of storage space as we all know the maintenance of cache is an expensive task so such wastage is not affordable. To avoid the problem of duplication of the data objects or web page duplicate suppression mechanism is to be used [9]. If the duplicate copy of data is saved at proxy cache then it acquires more space of storage in the analysis part given in work shows that the effect of duplication in the cache space [6].

3.4 Duplicate Suppression

You can reduce storage space requirements by avoiding duplicating copies of the same data. Content Engine provides the option to suppress storage of duplicate content elements. Duplicate suppression applies to any kind of content. Incoming content is not added to the storage area if identical content exists in the storage area; only unique content is added [14].

Many web pages having different URL's are duplicates of each other. If cache finds the duplicate copy of requested page, it will reduce the network cost when the page is served from cache.

Technique like data compression, caching and content simplification can be used to avoid unnecessary use of network bandwidth [6].

Due to large network size there are many pages on web, most of those pages will not be referenced multiple times by any one cache, means the probability with which the K^{th} page will be referenced is $1/K$. re-referenced follow a distribution similar to Zipf's law [12].

Utility of web pages can be extended using several methods such as delta encoding, perfecting and partial transfer. Simple hit ratio is always higher than weighted hit ratio because the no of references to the smaller resource is more than those larger resources.

3.5 Experimental Analysis

The experimentation carried out at the lab of our institute. Five systems from the lab are under the observation for 3 months. Those systems are use to analyse for access of internet to check the user interest and habits. File types such as .XML, .TXT, .PL, .GIF, .HTML, .CH, .CSS, .PNG, .PHP, and .BMP considered for analysis [2]. Size of the file given in table is in kb. The result of this analysis for five systems are summarised in table 1.

Table 1. Size of content According to type.

	Type of files										
System	XML	TXT	PI	JPEG	GIF	HTML	CH	.CSS	.PNG	PHP	BMP
1	586	35	202	1492	464	423	259	1169	495	131	0
2	0	26	0	981	312	772	0	776	163	0	361
3	604	142	0	151	126	608	248	387	252	0	49
4	0	17	0	77	44	174	0	315	487	0	0
5	0	0	107	516	196	326	0	423	180	50	0
AVG	1190	220	309	3217	1142	2303	507	3070	1577	181	410

Table 1 show that Style and image documents account for close to 60 - 65 percent. Unlike the web server workloads, however Table 1 show that image files consistently more requested document type (40-43 percent), followed by Style documents (about 20-22 percent). It has been observed that some of the file systems are not present on all the systems. Most of the cache space is acquired by the JPEG files.

Table 2. Reduced size of content on Disk.

	Type of files										
System	XML	TXT	PI	JPEG	GIF	HTML	CH	.CSS	.PNG	PHP	BMP
1	566	1	192	757	215	300	216	1065	385	69	0
2	0	6	0	626	177	558	0	744	131	0	361
3	594	3	0	118	18	539	248	205	135	0	13
4	0	1	0	20	32	150	0	300	418	0	0
5	0	0	97	281	45	93	0	311	161	44	0
AVG	1160	11	289	1802	487	1640	464	2625	1230	113	374

Table 2 show effect of reduction on the content stored at cache. We get much reduction effect in case of image files. If we discard the duplicate copies of the image files then we can save 50 percent of space required to store that image files, where the reduction in XML, PL and CSS files is very less. Reduction in number of copies of same files decrease the access latency time for the proxy cache [15].

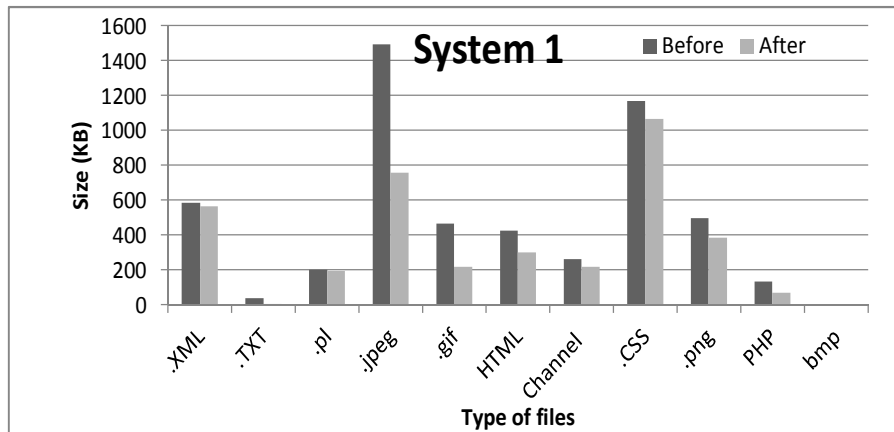


Figure.2 Type of files present on system 1

Figure 2 shows the type of data present at the cache of first system, from Figure 2 we can say that the maximum size occupied of first system is by image files. And the files like TXT and BMP is completely absent. That shows the user interest and habits. Here first bar shows the actual size of files with replication where second bar shows the size if we discard replicated data. “Aging daemon” which generates dummy references to all documents to age their reference rate estimates [18].

From Figure 2 it is clear that in image file of type jpeg we get 50 percent of space is saved if we discard the replicated data. Wherein the space saved in gif file is also same i.e. near about 50 percent. In case of HTML and png it is less about 10-20 percent. Wherein case of XML, CSS, and Channel it is negligible. So after reduction of image files we can save the 50 percent storage space occupied by image files. Kai Cheng, Kambayashi, Y., Mohania M., shows that more than 60% of total cache contents are never used and the low utilization rate is largely due to users' unawareness of the cache contents [19].

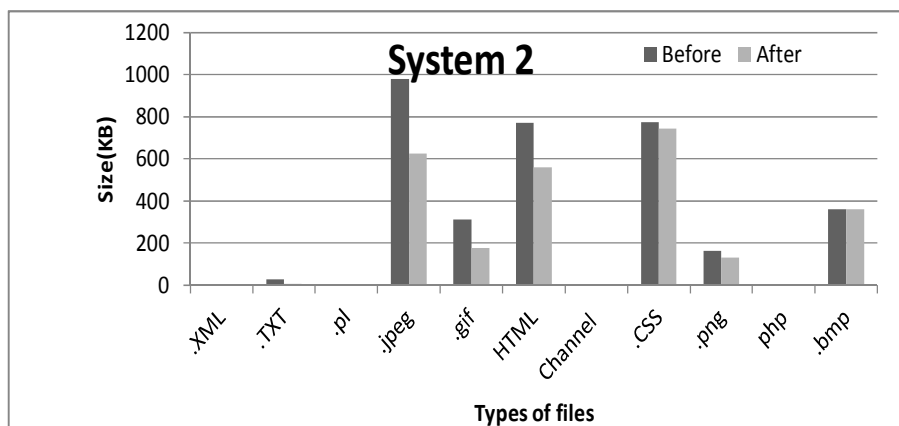


Figure 3. Type of files present on system 2

Figure 3 shows the type of data present at the cache of second system, from Figure 3 we can say that the cache occupies less space as compare to system 1. And the files like TXT, XML, PL, CHANNEL and PHP is completely absent. That shows the user interest and habits. Here first bar shows the actual size of files with replication where second bar shows the size if we discard replicated data. Storing of dynamic content in cache is not useful because that content does not used again for second time [5].

From Figure 3 it is clear that in image file of type jpeg and gif we get 40 percent of space is saved if we discard the replicated data. In case of HTML file we get 20 percent. Wherein case of PNG, CSS,

and BMP it is negligible. So after reduction of files we can save 20-30 percent storage space occupied by files. At the second system most of the files are not present so the actual size occupied by cache is less but we can observe that there is replication of data. Dynamic content cache stores new copy from web server each time [5].

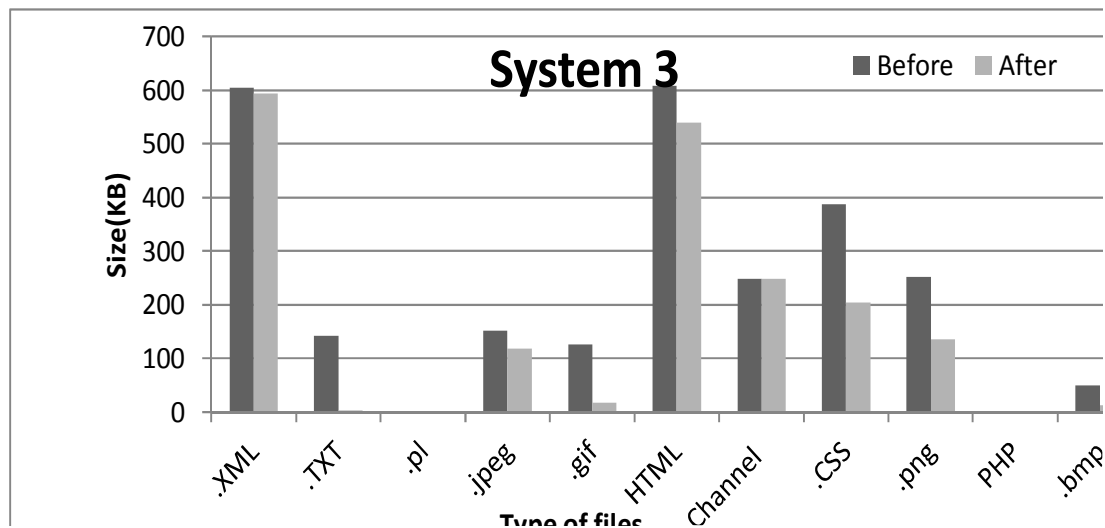


Figure 4. Type of files present on system 3

Figure 4 shows the type of data present at the cache of third system, from Figure 4 we can say that the maximum size occupied is by XML and HTML files. And the files like PL and PHP is completely absent. That shows the user interest and habits. Here first bar shows the actual size of files with replication where second bar shows the size if we discard replicated data [13].

From Figure 4 it is clear that the space is saved by discarding the replicated data is negligible in case of XML and HTML files. We save maximum size by discarding duplicates of GIF and TXT files about 80-90 percent. Wherein the space saved in JPEG, HTML and BMP file is also near about 10-20 percent. Wherein case of PNG, CSS, and BMP it is 30-40 percent. So after reduction of files there is no much effect in storage space requirement.

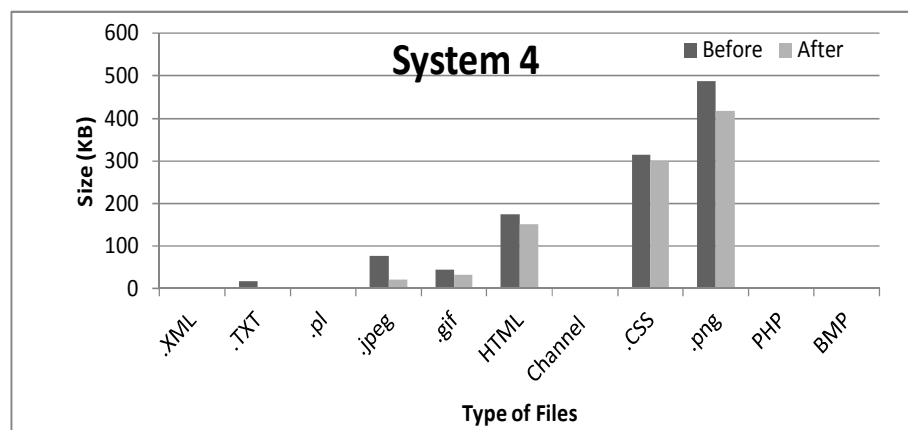


Figure 5. Type of files present on system 4

System 4 also has the less overhead on cache size because some of the files are not present in system 4 cache such as XML, PL, CHANNEL, PHP and BMP. And the files which are present on system 4 have no much effect of reduction. In case of jpeg there is maximum effect of reduction i.e. 70 percent. Wherein case of GIF, HTML, CSS and PNG it is only 10-20 percent. From the access log of system 4

we predict that there is variety of content is surf that's why there is no much effect of reduction on cache storage [6].

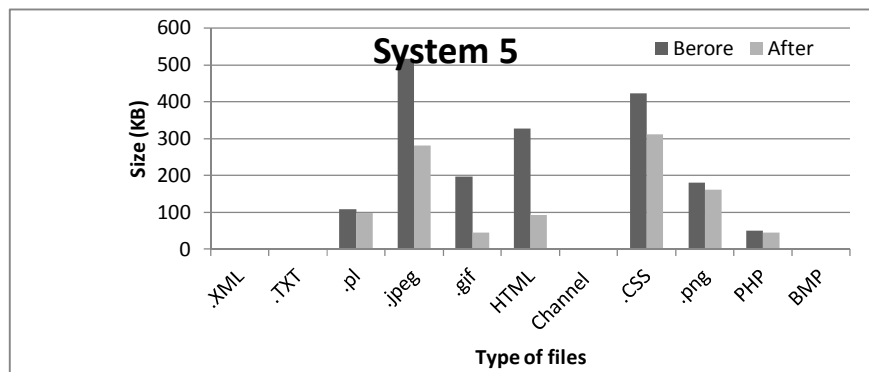


Figure 6. Type of files present on system 5

At system 5 we observe that XML, TXT, CHANNEL and BMP files are not present. Maximum size of cache is occupied by JPEG and CSS files. If we avoid the duplication of JPEG object then we can save 40 percent of storage space occupied by JPEG files [13]. In case of GIF and HTML the effect of reduction is better i.e. 70-80 percent. In case of PNG, PHP and PL files the effect of reduction is very less nearly 10 percent. If avoid duplication of CSS content then we save 20 percent of storage space required to store CSS files.

IV. RESULTS AND DISCUSSION

Figure 7 shows the overall cache space occupied by all the files present on all the five systems as well as the effect of reduction on cache space. From Figure 7 we can say that by reducing the XML content we can save 40 percent of cache size. In case of TXT files it is 50 percent. As we observe the graph for PL, PHP and BMP files then there is no effect of reduction. JPEG and CSS files requires maximum of cache space and the effect of reduction is 50 percent in case of JPEG files and 40 percent in case of CSS files. Maximum effect of reduction is seen in HTML and PNG files i.e. 60 percent. Cache space occupied by GIF files is less but the effect of reduction is much i.e. 50 percent.

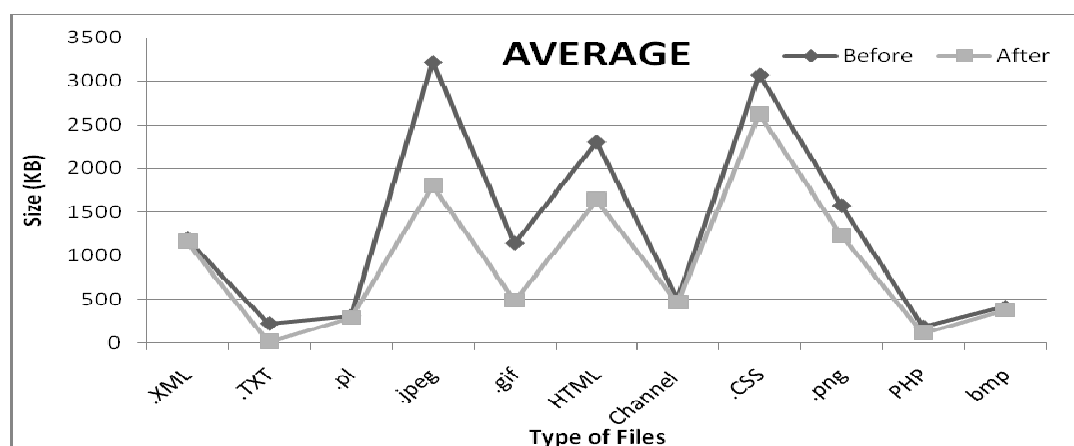


Figure.7 Average size of files on all Systems

Figure 7 shows the effect of reduction on the files present in cache of all systems from the statistics shown in figure 7 we get the effect of duplicate suppression. Duplicate suppression saves the cache space as well as the access latency is decreases for cache described in [15].

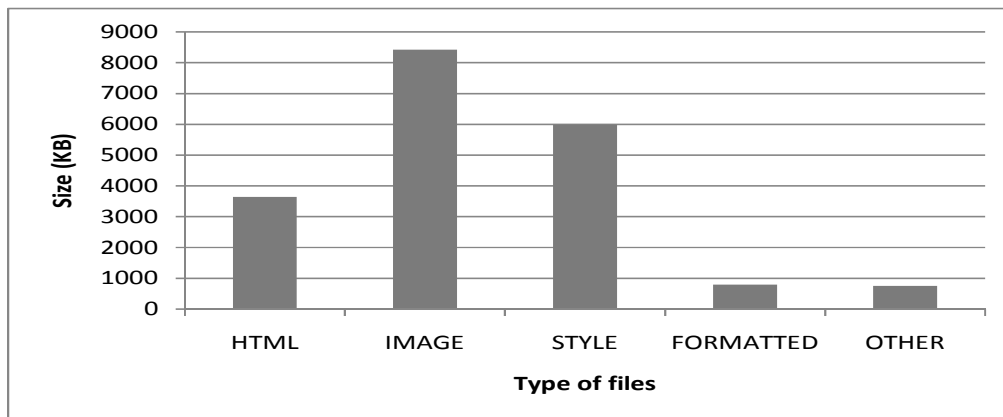


Figure.8 Average size of files on all Systems

Figure 8 shows the comparative graph of type of files when it is grouped category wise. Files such as GIF, JPEG, BMP and PNG is categorised into image files. CSS and PL files categorised as style file. And the remaining files in the category of other. From Figure 8 we can predict that the image files are most requested content in all. Image files occupy 43 percent of total storage space of cache. And also the effect of reduction has greater significance in saving of storage space. Major portion of storage space is occupied by the image, HTML and Style files 80-90 percent.

It is clear that maximum size of cache is occupied by image file and also the effect of duplicate suppression is higher in case of image files. So the reduction in image files helps us to save the cache space as well as the access latency. By using the concept of duplicate suppression we save the space required by the files along with that the access latency at proxy is also decrease. If we store multiple copies of same object that increase the search time for lookup of specific files that increase the response time for user.

V. CONCLUSION

The analysis based experimental results proves the need for methodology that improve the web access performance to enhance bandwidth utilization and greater connectivity speed. Here the suggested Design aspects improves the web performance in terms of reduced traffic, reduced latency, improved user response time, and optimal use of the existing bandwidth by using web caching. Content aliasing successfully detected using a web-based application, database queries and files system calls even in the large scale web applications. A considerable amount of duplicate storage can be avoided through the suggested methodology. This methodology is useful to avoid content aliasing at proxy cache that saves the unnecessary cache space required to store multiple copies of same content. It is also helpful in reducing the access latency and increase the response time of proxy cache. This methodology helps to web proxy caches to improve the performance during large web services. This work can be further optimize by the Daemon Process, which can be design and run periodically to check the consistency of the data cached and the data at the web server. This can be scheduled during the slack time with the less traffic which will not add any additional toll on the bandwidth as well as it updates the TTL – Time to Live Period of the cached data; which results in the more cache hits with fresh data.

ACKNOWLEDGEMENTS

This paper would not be possible without the continuous guidance and valuable inputs of Dr. Tapan Bagchi, Director – SVKM's NMIMS Shirpur Campus. The additional support taken from the following corporate professionals, for required data and design aspects based on their experience which creates innovative methodologies and data management solutions to help accelerate business breakthroughs and deliver outstanding cost efficiency: Mr. Sachin Punadikar, Senior Staff Software Engineer, NAS Team, India Software Lab, IBM India. Ozone2, Pune; Mr. Pramod Ghorpade, Dev Manager at NetApp, Bengaluru; Mr. Vinod Ghorpade, Senior Practice Manager at Wipro Technologies, Reading, United Kingdom (UK), Mr. Sujay Ghorpade, SAP PS/c Projects Consultant, Robert Bosch Engineering & Business Solutions, Bangalore.

REFERENCES

- [1] Kartik Bommepally, Glisa T. K., Jeena J. Prakash, Sanasam Ranbir Singh and Hema A Murthy "Internet Activity Analysis through Proxy Log" IEEE, 2010.
- [2] Jun Wu; Ravindran, K., "Optimization algorithms for proxy server placement in content distribution networks," Integrated Network Management-Workshops, 2009.
- [3] Ngamsuriyaroj, S. ; Rattidham, P. ; Rassameeroj, I. ; Wongbuchasin, P. ; Aramkul, N. ; Rungmano, S. "Performance Evaluation of Load Balanced Web Proxies" IEEE, 2011.
- [4] E-Services Team, "Changing Proxy Server" by the Robert Gordon University, School hill, Aberdeen, Scotland-2006.
- [5] Chen, W.; Martin, P.; Hassanein, H.S., "Caching dynamic content on the Web," Canadian Conference on Electrical and Computer Engineering, 2003, vol.2, no., pp. 947- 950 vol.2, 4-7 May 2003.
- [6] Sadhna Ahuja, Tao Wu and Sudhir Dixit "On the Effects of Content Compression on Web Cache Performance," Proceedings of the International Conference on Information Technology: Computers and Communications, 2003.
- [7] Mark S. Squillante, David D. Yaot and Li Zhang "Web Traffic Modeling and Web Server Performance Analysis" Proceedings of the 38' Conference on Decision & Control Phoenix, Arizona USA December 1999.
- [8] C. E. Wills and M. Mikhailov, "Studying the Impact of More Complete Server Information on Web Caching," Computer Communications, vol. 24, no. 2, pp. 184.190, May 2000.
- [9] J Wang "A Survey of Web Caching Schemes for the Internet" - Cornell Network Research Group (C/NRG), Department of Computer Science, Cornell University 1999.
- [10] A. Mahanti, C. Williamson, and D. Eager, "Traffic Analysis of a Web Proxy Caching Hierarchy," IEEE Network Magazine, May 2000.
- [11] N. Shivakumar and H. Garcia-Molina, "Finding near Replicas of Documents on the Web" Proc. Workshop on Web Databases, Mar. 1998.
- [12] L. Breslau, P. Cao, L. Fan, G. Phillips, and S. Shenker. Web caching and Zipf like Distributions: Evidence and Implications. In Proc. Infocom '99. New York, NY, March, 1999.
- [13] Jeffrey C. mogul "A trace-based analysis of duplicate suppression in HTTP," Compaq Computer Corporation Western Research Laboratory, Nov. 1999.
- [14] Guerrero, C.; Juiz, C.; Puigjaner, R.; "Web Performance and Behavior Ontology," Complex, Intelligent and Software Intensive Systems, 2008. CISIS 2008. International Conference on, vol., no., pp.219-225, 4-7 March 2008.
- [15] Wei-Kuo Liao, Chung-Ta King "Proxy Pre-fetch and Prefix Caching" Parallel Processing, International Conference on, 2001. , vol., no., pp. 95- 102, 3-7 Sept. 2001.
- [16] Triantafillou, P.; Aekaterinidis, L.; "Web proxy cache replacement: do's, don'ts, and expectations," Network Computing and Applications, 2003. NCA 2003. Second IEEE International Symposium on, vol., no., pp. 59-66, 16-18 April 2003.
- [17] K. Bharat, A. Broder, J. Dean, and M. R. Henzinger, "A Comparison of Techniques to Find Mirrored Hosts on the WWW," IEEE Data Eng. Bull., vol. 23, no. 4, 2000.
- [18] Shim, J.; Scheuermann, P.; Vingralek, R.; "Proxy cache algorithms: design, implementation, and performance," Knowledge and Data Engineering, IEEE Transactions on , vol.11, no.4, pp.549-562, Jul 1999.
- [19] Kai Cheng; Kambayashi, Y.; Mohania, M.; , "Efficient management of data in proxy cache," Database and Expert Systems Applications, 2001. Proceedings. 12th International Workshop on , vol., no., pp.479-483, 2001.

Authors

Suryakant B Patil was born in Maharashtra, India, in 1975. He received the Bachelor and the Masters both in Computer Engineering, both from the Walchand College of Engineering, Sangli. He is currently pursuing the Ph.D. degree with the Department of Computer Engineering, NMIMS Mumbai and he has submitted his final thesis of the same. His research interests include Software Engineering, Project Management, MIS, Web Engineering, Mobile Communication and information Security.



Sachin D Chavan was born in Maharashtra, India, in 1987. He received the Bachelor in Computer Engineering from North Maharashtra University, Maharashtra and pursuing his Masters in Computer Engineering from the NMIMS Mumbai. His research interests include Web Engineering, Design and Analysis of Algorithms, Computer Networking, Operating



Systems and Software Engineering.

Preeti S Patil was born in Karnataka, India, in 1977. She received the Bachelor in Computer Engineering from VTU University and the Masters in Computer Engineering from Mumbai University. She is currently pursuing the Ph.D. degree with the Department of Computer Engineering, NMIMS Mumbai and she has submitted her final thesis of the same. Her research interests include Web Engineering, Mobile Communication, information Security, Databases, Data Warehousing and Mining.

