© 2016 IJSRSET | Volume 2 | Issue 2 | Print ISSN : 2395-1990 | Online ISSN : 2394-4099
Themed Section: Engineering and Technology

Improving Caching Technique through Innovative Replacement Algorithm of Page for Web Proxy Caching

Sumit Rajoriya, Varsha Zokarkar

Department of Computer Science and Engineering, Sushila Devi Bansal College of Engineering, Indore, Madhya Pradesh, India

ABSTRACT

With an ever growing importance of human activity on internet, World Wide Web is growing rapidly and number of users is increasing day by day to access web page. By this, the results have heavy network traffic and server load etc. To overcome these problems, proxy server caching is one of the solution. Proxy caching speeds up the service requests by retrieving the saved web pages from an earlier request through the same node or even other nodes. Many researcher have done their work in the field of proxy caching. I have proposed an innovative replacement algorithmof page for proxy server caching i.e. called Mark Page Detect Randomly Proxy Caching (MPDRC) algorithm. The MPDRC algorithm increases the hit rate on proxy server cache. With the help of proposed innovative page replacement algorithm, the performance of proxy server caching has been improved.

Keywords: Proxy Server, Web cache, Server Load, Latency, Network Traffic, Page Replacement Algorithm(LRU, LFU, FIFO).

I. INTRODUCTION

The internet or World Wide Web is a global network system which is interconnected to the different computer nodes. World Wide Web is a network of network which has many public, private government and academic networks. The internet users are increasing day by day. It has more than 3.2 billions of users [1]. Due to the large number of users on internet, it is not possible to satisfy the demands of every user in an proficient manner. Users suffer from the network traffic, low response time and server load etc. Hence to overcome network traffic, low response time and load on server Proxy server is intermediated between the clients and the server. Caching permits organization to extensively reduce their upstream bandwidth. Web proxy cache has to play an important role to reduce the latency, network traffic and server load. As the cache size is limited, it is not possible to keep all the demanded web pages in a cache. When the cache is filled and a request for web page appears, which is not present in the cache then a miss occurs. With the help of page replacement algorithms, one of the pages is to be discarded to create space for new arrival page which is requested by clients. The better utilization of cache is done by the selection of effective page replacement policy. The prominent used page replacement algorithms are LRU, LFU and FIFO etc [2, 3]. The requests of Client's web pages are fulfilled by cache memory at proxy server. Whenever a web page is requested by client, firstly the demand request will arrive to proxy server. Proxy server will check the demand web page in its cache. If requested web page is found Proxy server will check the demand web page in its cache. If requested web page is found then it is served to client otherwise demand request will be forwarded to the main web server and the demand web page is fetched by the proxy server and save the copy of requested demand page and served to clients.

The goal of this paper is to improve caching technique through design and implementation of a web proxy caching algorithm which can be reduced to server load and latency. A cache of web is a caching mechanism for the provisional storage of web documents, to reduce latency and server load. A web cache stores the replica of documents passing throughout it.

The remaining sections of this paper consist of four different parts. In related work section we focus on the work done in the performance improvement of proxy

caching .The methodology section describes the process implementing the system. The result and discussion section focus our finding and significant. In conclusion conclude our work with limitation and future scope.

II. METHODS AND MATERIAL

A. Related Work

Web proxy caching and tracing are vastly active research areas. Traffic comprises analysis of Web access hints from the view of browsers proxies. Previous tracing studies were bounded in number of requests, request rate and variety of population. The recent studies of tracing have wide scope and varied area. In addition to static analysis, some studies have also used to trace ambitious cache simulation to characterize the sharing property and localities of very large suggestions and to study the special effects of aborted connections, persistent connections and cookies on the concert of proxy caching. Many researcher have performed useful studies to improve current caching techniques in the field of proxy server. Recent studies on traffic of web are based on the analysis of Web access hints from the approach of browsers proxies. previous tracing studies were restricted in number of requests, request rate, and diversity of population. **John Dilley et al [4]** reported on the implementation and characterization of two newly proposed cache policies, LFU with Dynamic Aging (LFUDA) and Greedy Dual Size – Frequency (GDS-F) in the squid cache.**Ismail Ari et al [5]** proposed an adaptive Caching Scheme using Multiple Experts. It suggested that use of machine learning algorithms to choose the present best mixture of policies or policy by permitting each adaptive node to tune itself based on load of work. **Martin Arlitt et al [6]** introduced virtual caches, an approach for getting better the performance of the cache for multiple metrics simulteneously. **J. Almeida et al [7]** liked to further explore the performance of the new cache replacement policies under a more realistic proxy workload. The best way to do this is by using a benchmark able to generate a workload appropriate for a proxy, such as the wisconsin proxy benchmark. **Richa Gupta et al [8]** have proposed pair of replacement algorithms MFMR and AF-LRU for L1 and L2 cache for proxy server which performs better result than existing pair of replacement algorithms.**Irani et al [9]** have introduced greedy dual size, which have as a feature of locality with size and cost concerns in a simple and non parameterized fashion for elevated performance. **Richa Gupta et al [10]** has described a preeminent pair of replacement algorithms for L1 and L2 cache for proxy server. According with them the access pattern of L1cache and L2cache are diverse. Thus the innovative replacement algorithm of page which is bountiful proficient results for L1 cache may not be suitable for L2. They summarized that the pair of replacement algorithm attempt was more efficient than the same used algorithm. So according to them different algorithm on L1 and L2 cache perform better than same used algorithm. **Yong Zhen Guo et al [11]** proposed web page prefetching technique, they must be able to guess the next set of web pages that will be entréed by users and "Page-Rank Like Algorithm" was proposed for conducting web page prediction. **Golan et al [12]** proposed an optimal offline algorithm for page replacement in multi-level cache based on an algorithm for the rest listing updated problem and the DEMOTE operation. **Shiva Shankar Reddy et al [13]** have proposed a new method of caching HTTP Proxy servers which gets lower bandwidth by maintaining a web cache of internet objects. **Dr. Murali Bhaskaran et al [14]** discussed various data pre-processing techniques that are approved out at proxy server entrée log which generate web entrée pattern. These patterns are used for further application. **Song Jiang et al [15]** have proposed DULO algorithm, which utilizes both spatial and temporal locality in buffer cache management. **Ahcia Trivino-Cabrera et al [16]** performed a trace driven simulation study of the replacement policies has been devloped for the traffic generated by each considered content type. It had studied six replacement algorithms LFU, LFUDA and LRU, other three are specially developed for web documents GDSF, GD and GD Size, finally concluded that no replacement strategy do better than the other for all content type.

B. Issues the Caching of Web Proxy

Due to the continually and explosive growing size of the web, dispersed caching has received huge notice. The major aspire of cache is to shift the frequently accessed information nearer to the end of users. Caching system should get better performance for network operators, content providers and end users, it can be accepted as an effectual way to grow the speed of web access, reduce server load, reducing latency perceived by the end users, improve response time and reduce network traffic to the end of users.

i. Load balancing

The situation occurs at any time for abundance of clients who wants to wish simultaneously access content or access some services from its local cache with particular server. If the position is not provisioned to handle with these entire

clients request simultaneously, service may be lost or degraded. Several approaches to overcome this issue have proposed. The most frequently used method that is called caching. This caching strategy saves the copy of popular web pages or services during the Internet, this spreads the job of serving a web page or service across different servers. Caching accelerates the retrieval of web pages that access by same client or even other clients. It improves the performance of server by reducing lot of server load.

ii. Transparency

Transparency of cache systems allows users to get the advantages of caches without knowing that they exist, or without knowing to their physical location. The benefits of this technique are user friendly, no configuration is required by the end user and users cannot bypass the cache.

iii. **Scalability**

It is essential that the cache system should be scalable as the servers and number of users increases. It can be cooperative or stand-alone caches and clustered. Stand-alone caches are easier to maintain and better suited for individual systems. However, cooperation between caches provided extra information about caching data, which could be communicated between caches without showing to the originating server.

iv. Cache Miss

Cache systems should be proficient of efficiently managing cache misses. When a cache miss occurs, a decision should be taken on where to be forwarded the request. Also a cache system should be decided that which data is to be cached or should all cached data be pleasures equally.

C. Design and Implementation of Proposed Proxy Caching Algorithm

Web caching is a promising technology in Web environment and in web caching, if the client requests a demand page from a server, it forwards to the server and will give response from the server. Which objects are cached form where location, web caching technology can be classified into following three categories i.e., server-end proxy caching and client-end proxy caching.

If the request is authenticated by the filter, the proxy gives the resource by linking to the appropriate server and serves the request service on basis of the client's request. A proxy server may optionally change the server's response or the client's request and sometimes it may serve up the request without calling to specified server. In this case, it caches response from its local cache.

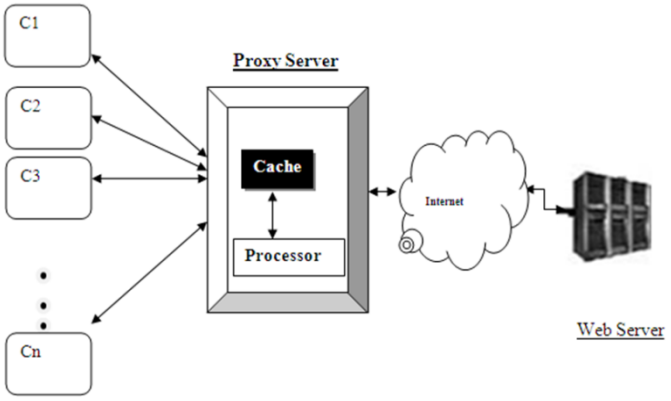

Figure 1 : Proxy Server with Cache

Figure 1 show a proxy server with cache memory which runs with a lot of characteristics such as reduces Latency time, load on web server, network traffic. This architecture also innately helps speedy browsing of pages. In this system when According to the proxy cache filled and new demand page request disembarks at proxy a page replacement algorithm settle on which page has to be evicting from the cache. The page replacement algorithm is decided the efficiency of system.

Mark Page Discard Randomly Caching Algorithm

There has been embracing theoretical and pragmatic work done on exploring policies of web caching that achieve best result under different recital metrics. Many algorithms have proposed and initiate efficient for caching of web proxy. These algorithms collection from simple habitual schemes such as least-recently used (LRU), least- frequently used (LFU), first-in first-out (FIFO) and various size-based algorithms, to compound hybrid algorithms such as LRU-Threshold, which bear a resemblance to LRU with limit of size on single cache elements, lowest relative value (LRV), which is used to size , cost and last position time to evaluate its utility, and Greedy Dual, which combines locality, cost and size deliberation into a single online algorithm.

The table 1 shows the proposed Mark Page Discard Randomly Caching algorithm. This algorithm is focused on the characteristic i.e. algorithm of swapping documents. With the study of web caching features is going to further;

algorithms of exchange documents based on the statistics of serenaded web data are proposed. Following deem factors into its scheme:

- Freshness of document
- Document reference frequency
- Consistence of documents
- Document size

Efficient caching schemes combine the factors more than one in their implementation of caching. Some algorithms follow different cache architecture to get better caching performance.

Table I: Mark Page Discard Randomly Caching Algorithm (MPDRC)

1. WHILE there is a page p_i € Cache in the existing window W.
2. Serve up first such p_i and does not mark the page
3. IF again p_i find than marks the page
4. IF all the pages in the web cache are marked
5. Unmark all the pages
6. Discard randomly an unmarked page in the cache.

III. RESULTS AND DISCUSSION

A proposed innovative page replacement algorithm (MPDRC) is developed in windows XP by using C# .Net (Microsoft visual studio 2010) and the unique identification number is assigned to unique URL's to log of proxy server. These numbers is taking as a reference string that suits input to the innovative algorithms. The results of this algorithm on the basis of hit ratio are shown in Table II.

Cache Size(MB)	Number of Hit Analysis on Different Cache Size and Number of Request is 2500			
	Replacement Algorithms			
	FIFO	LFU	LRU	MPDRC
2	337	413	337	460
3	398	483	405	537
4	466	535	456	625
5	578	640	536	709
6	565	676	579	731
7	604	730	629	717
8	657	750	689	788
9	717	756	746	862
10	805	786	861	870

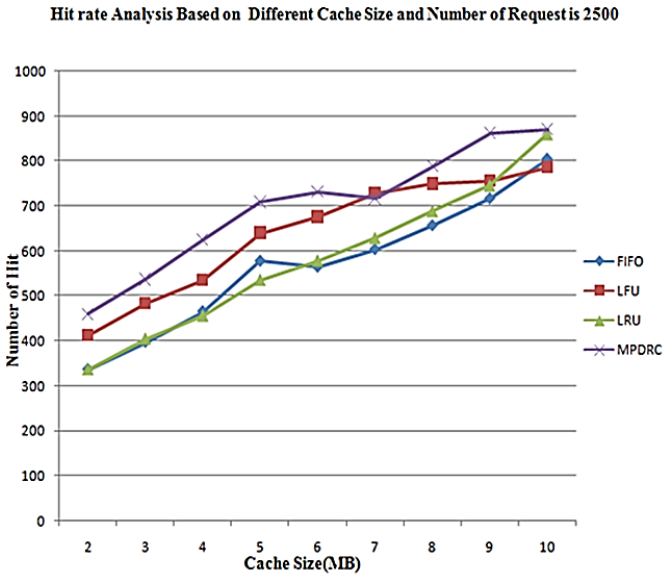

Figure 2 : Hit Ratio Analysis

IV. CONCLUSION

This paper is basically focused to explore the web proxy caching algorithm which is best suitable for proxy server. A real trace web reference is got with the help of log details of proxy server. For the simulation numeric reference string was acquired by giving numeric identity to each of the URLs. During simulation we have compared FIFO, LRU and LFU replacement algorithms with my proposed innovative page replacement algorithm (MPDRC). We find that MPDRC algorithm 24.93% better than FIFO, 9.58% better than LFU and 23.98% better than LRU algorithm. Overall average of my innovative page replacement examined 19.26% better than existing algorithm. After comprehensive simulation experiments is summarized that for proxy caching the MPDRC hit ratio performance better than others existing algorithms.

V. REFERENCES

[1] http://www.internetworldstatus.com/status.htm Accessed 15 Nov 2015).

[2] David A. Malts and Pravin Bhagwat “Improving HTTP caching proxy performance with TCP tap”. Technical report, IBM. March 1998.
[3] Abraham Silberschatz And Peter Baer Galvin, “operating system concepts”5th Addison-Wesley Longman Publishing Co., Inc. Boston, MA, USA , 1997.
[4] John Dilley, Martine Arlitt and Stephane Perret “Enhancement and Validation of Squid’s Cache Replacement Policy” Internet Systems and Applications Laboratory HP Laboratories Palo Alto HPL- 1999-69, May 2009.
[5] Ismail Ari, Melani Gottwals Dick Henze ,” SANBOOST: Automate SAN Level Caching in Storage Area Netowkrs “ International Conference on Automatic Computing pp-164-171,2004.
[6] Martin Arlitt, Ludmila Cherkasova, John Dilley, Richard Friedrich, Tai J in "Evaluating Content Management Techniques for Web Proxy Caches", published in ACM SIGMETRICS Performance Evaluation Review Volume 27 Issue 4, March 2000.
[7] J. Almeida and P. Cao "Measuring Proxy Performance with the Wisconsin Proxy Benchmark", Technical Report, University of Wisconsin-Madison, April 1998.
[8] R. Gupta and Sanjiv Tokekar, ”Pair of replacement algorithms MFMR and AF-LRU on L1 and L2 cache for proxy server”. INDICON 2009 publication by IEEE.
[9] P. Cao and Irani, “Cost aware WWW Proxy Caching Algorithms” , In roc USENIX Symp. Internet Technology and System Monterey 1997.
[10] R. Gupta and Sanjiv Tokekar , ”Preeminent pair of replacement algorithms for L1 and L2 cache for proxy server”. First Asian Himalayas International Conference AH-ICI 2009.
[11] Yong Zhen Guo, Kotagiri Ramamohanarao and Laurence A. F. Park “Personalized PageRank for Web Page Prediction Based on AccessTime-Length and Frequency” This paper published in 2007 IEEE/WIC/ACM International Conference on Web Intelligence.
[12] Gala Golan “Multilevel cache management based on application Hints” computer science department, Technion Haifa 32000, ISRAEL. November 24, 2003.
[13] Shiva Shankar Reddy P,Swetha L “Analysis and Design of Enhanced HTTP Proxy Cashing Server“ paper published in International Journal of computer Technology, Volume 2 (3), 537-541.
[14] V. Sathiyamoorthi and Dr.Murali Bhaskaran “Data Preprocessing Techniques for Pre-Fetching and Caching of Web Data through Proxy Server” International Journal of Computer Science and Network Security, VOL.11 No.2011.
[15] Song Jiang, X. Ding E. Tan and X. Zhang,”DULO: An Effective Buffer Cache Management Scheme to Exploit Both Temporal and Spatial Locality.” Proc. Of the USENIX Symp. File Storage and Technologies 2005.
[16] F.J. Gonzalez-Canete, E. Casilari, Ahcia Trivino-Cabrera "Characterizing Document Types to Evaluate Web Cache Replacement Policies," ecumn, pp.3-11, Fourth European Conference on Universal Multiservice Networks (ECUMN'07). 2007 IEEE.
[17] J. L. Hennessy and D. A. Patterson, Computer Architecture: A Quantitative Approach, 3rd Edition, Chapter 5 – Memory Hierarchy Design. Morgan Kaufmann Publishing, 2002.
[18] Paul H J Kelly “Advanced Computer Architecture” Chapter 2. Caches and Memory Systems, 2009.
[19] [Online] https://developers.google.com/speed/articles/web-metrics (Accessed 30 June 2015).
[20] Y.Zhou and K.Li, “Second Level Buffer cache Management”, IEEE Transactions on Parallel and Distributed Systems July2004.
[21] L. Bresalu, Cao, S. Shenker , “Web Caching Zipf like Distrubution”1999 IEEE.
[22] Vladmir V. Prischepa, “AN Efficient Web Caching Algorithm based on LFU-k replacement policy”, Spring Young Researcher’s Colloquium on Database and Information System,2004 .
[23] M. Bach “The Design of UNIX Operating System”, Prentice-Hall 1986.
[24] X. Li, Aboulnanga, K. Salem,” Second Tier Cache Management Using Hints”, USENIX Conference on file and storage Tecnologies,2005.