

# An Extensive Review of Webs Caching Techniques to Reduce Cache Pollution

Jitendra Singh Kushwah<sup>1\*</sup> Dr. Sitendra Tamrakar<sup>2</sup>

<sup>1</sup>Dept of CSE, ITM University, Gwalior (M.P.) India

<sup>2</sup>Dept of CSE, CCET, Durg (C.G.) India

**Abstract** – Caching has been used for decades as an effective performance enhancing technique in computer systems. The Least Recently Used (LRU) cache replacement algorithm is a simple and widely used scheme. Proxy caching is a common approach to reduce network traffic and delay in many World Wide Web (WWW) applications. However, some characteristics of WWW workloads make LRU less attractive in proxy caching. In the recent years, several more efficient replacement algorithms have been suggested. But, these advanced algorithms require a lot of knowledge about the workloads and are generally difficult to implement. The main attraction of LRU is its simplicity. In this paper we study several techniques about page replacement, cache pollution and web caching technique at the end of this paper we propose a new idea of web page scheduling.

**Keywords:** IP, LAN, HTML, Caching, Proxy.

## 1. INTRODUCTION

Proxy server is placed between a client utility equivalent to a web browser, and a real server the request to the real server. Different key features are improving performance of proxy server like caching the documents and thread polling etc. It provides safety like firewall and security of local Area Network (LAN) from having access to of unauthorized customers. Foremost feature is caching the web documents. Caching refers to store copies of the popular documents in proxy memory and thus used it for future references and reduces the bandwidth requirement. There are different available techniques but this paper is proposing the new and better technique. It blocks all solicitations to the real server to look on the off chance that it might well satisfy the solicitations itself. If not, it forwards unique to improve the performance of Least Recently Used- Distance (LRU-D) of caching the web document in the proxy server. Proxy server is a server (a computer method or a software program) that acts as a middleman for requests from purchasers seeking resources from other servers. A client associates with the proxy server, asking for some supplier, likened to a document, association, net web page or distinctive resource, close by from an alternate server. The proxy server evaluates the request with respect to its filtering rules (Yong Woon 2001). For example, it'll filter traffic with the aid of IP tackle or protocol if the request is validated via the filter, the proxy presents the useful resource via connecting to the vital server and requesting the provider on behalf of the patron. A

proxy server may just alternatively adjust the client's request or the server's reaction, and in numerous examples it might serve the demand without reaching the specific server.

A proxy server has many potential purposes, including:

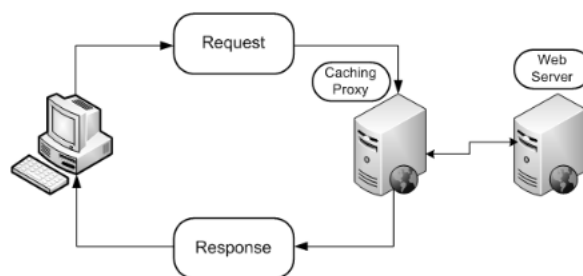


Fig.1 System with caching

- To keep machines in the back of it nameless (most of the time for safety)
- To pace up access to resources (using caching). Net proxies are often used to cache websites from an online server.
- To apply entry policy to network services or content, e.g. to dam undesired websites.

- (d) To log / audit usage, i.e. to provide corporation employee internet usage reporting.
- (e) To avoid safety/ parental controls.
- (f) To examine transmitted content material before movement.
- (g) To examine outbound substance material, e.g., for data leak security.
- (h) To maintain a strategic distance from local limitations (Jochen Hollmann 2007).

## 2. PROXY SERVER FUNCTIONS

Proxy servers implement one or a greater amount of the accompanying capacities:

- (a) **Caching proxy server** - A caching proxy server quickens benefit asks for through retrieving content saved from a before request made by technique for comparable supporter or even different clients. Caching proxies keep nearby copies of quite often requested resources, allowing enormous organizations to tremendously lessen their upstream bandwidth usage and cost, whilst tremendously increasing efficiency. Most ISPs and big companies have a caching proxy. These machines are worked to convey radiant document framework execution (frequently with RAID and journaling) furthermore contain hot-rod adaptations of TCP. Caching proxies were the first kind of proxy server (Wei-hsiu 2004).
- (b) **Web proxy**- A proxy that focuses on worldwide wave traffic is called a "web proxy". The most well-known utilization of a web proxy is to serve as a web store. Most proxy programs give a way to deny access to URLs particular in a blacklist, hence giving substance material filtering. This is regularly used in a corporate, educational or library environment, and wherever else where content filtering is desired. Some internet proxies reformat web sites for a targeted reason or audience, akin to for mobile phone telephones and PDAs.
- (c) **Content-filtering web proxy**-A content filtering web proxy server presents regulatory control over the content which might be handed-off through the proxy. It is mostly utilized in each business and non-industrial companies (peculiarly schools) to ensure that internet usage conforms to suitable use coverage. In some circumstances users can circumvent the proxy, considering that there are offerings designed to proxy (Geetha 2009).
- (d) **Anonymous proxy server**-An anonymous proxy server (at times called a web proxy) for the most part endeavors to anonyms web surfing. There are diverse assortments of anonymizers. One of the more common variations is the open proxy. Because they are regularly problematic to track, open proxies are especially useful to these looking for on-line anonymity, from political dissidents to computer criminals.
- (e) **Hostile proxy**-Proxies can be mounted with the intention to eavesdrop upon the dataflow between client machines and the net. All accessed pages, as good as all varieties submitted, will also be captured and analyzed via the proxy operator.
- (f) **Intercepting proxy server**-A intercepting proxy consolidates an proxy server with an entryway or router (consistently with NAT capacities) Connections made by way of purchaser browsers by means of the gateway are diverted to the proxy without consumer-aspect configuration (or often knowledge) (W. Ma 2002).
- (g) **Transparent and non-transparent proxy server**-The time period "transparent proxy" is most frequently utilized erroneously to signify "intercepting proxy" (for the reason that the customer does now not have to configure a proxy and can't instantly detect that its requests are being proxies). Obvious proxies can be carried out utilizing Cisco's WCCP (web Cache control Protocol). A 'non-obvious proxy' is a proxy that modifies the request or response as a way to furnish some introduced service to the person agent, such as workforce annotation offerings, media sort transformation, protocol discount, or anonymity filtering".
- (h) **Forced proxy**-The time period "forced proxy" is questionable. It way both "intercepting proxy" (on record that it channels all visitors on the main close by passage to the Internet) and its detailed inverse, "non- intercepting proxy" for the reason that the consumer is compelled to configure a proxy with the intention to entry the internet) (S. Sen 1999).
- (i) **Open proxy server**-Because of the fact proxies possibly used to manhandle, strategy administrators have developed an amount of strategies to refuse service to open proxies. Numerous IRC networks mechanically test customer frameworks for perceived types of open proxy. In like manner, an email server might be arranged to consequently test email senders for open proxies.



- (j) **Reverse proxy server**-A reverse proxy is a proxy server that's established in the regional of a number of internet servers. All traffic coming from the web and with a destination of probably the most net servers goes by way of the proxy server (R. Tewari 1998).

### 3. FEATURES OF PROXY SERVER

Some of most important aspects are as given below:-

- (a) **Caching**- The objective of all caching is the same – to block reacquiring data that has as of now been recovered with the goal that reaction time is enhanced and transfer speed usage minimized (R. Ayani 2003).

- (i) **These are advantages of Caching like**- Web caching reduces bandwidth consumption, thereby decreases network site visitors and lessens network congestion.

- (ii) **There is some Disadvantages of Caching like**- The predominant drawback is that a purchaser possibly watching at stale information as a result of the lack of right proxy updating.

- The entry latency may increase within the case of a cache pass over due to the extra proxy processing. Subsequently, cache hit rate ought to be expanded and the cost of a cache miss ought to be minimized when planning a caching system.
- A solitary proxy cache is constantly a bottleneck. A confine must be set for the amount of customers a proxy can serve. An effectivity therapist certain will need to try and be upheld.

- A solitary proxy is a solitary purpose of failure.

- Using a proxy cache will diminish the hits on the first remote server which may frustrate a considerable measure of learning suppliers, in light of the fact that they can't keep up a real log of the hits to their pages. Hence, they might decide not to allow their documents to be cacheable (Jun Pyo Lee 2008).

- (b) **Web Mining**- Web caching is the temporary storage of web articles (tantamount to HTML documents) for later retrieval. Proponents of web caching announce three huge focal points to web caching: lessened data transfer capacity utilization (less demands and reactions that need to go over the network), reduced server load (fewer requests for a server to control), and diminished latency (for

the reason that cached responses are available instantly, and towards the client being served). A fourth is regularly added: extra reliability, as some objects may be retrievable through cache even when the usual servers aren't reachable. Together, these features can make the World Wide Web less expensive and better performing. One predicament of caching is the talents of making use of an out-of-date object stored in a cache as a substitute of fetching the present object from the origin server.

- (c) **Web Caching Techniques**-A cache is a capacity territory that is nearer to the element requiring it than the authentic source. Accessing this cache is commonly rapid than being in a position to access the understanding from its original source. A cache is most likely saved in memory or on disk. A memory cache is ordinarily more rapidly to read from than a disk cache, but a memory cache typically does not survive system restarts.

- (d) **Proxy Caching**-In proxy caching, the cache server receives the request for an object from a client. If the object is present in its cache, it responds with the object. Else, it requests the source of object and ensures the client has the requested item. In the event that required, the server may likewise store the article in its cache so to bring down the network congestion consequent time the object is asked. Caching server is placed virtually customer (at network gateway) to cut back the latency and hops. The advantages of Proxy caching contain lowered latency and network traffic that makes expertise of the web higher and greater availability of the web sites. Nevertheless, the disadvantages that lie with this approach are that cache is single factor of failure, the browsers have got to be configured and no such method exists that can dynamically add more caches when required.

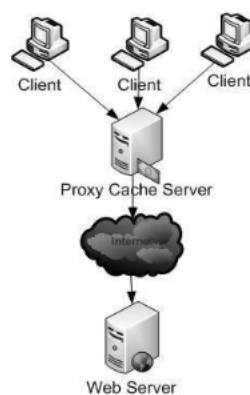


Fig 2. Proxy caching

- (e) **Reverse Proxy Caching** - Alternatively of inserting cache server virtually client, the cache server is placed close to the servers in this type of variant of Proxy Caching. When a server is predicted to obtain sumptuous requests concurrently, in any such scenario, reverse proxy caching is an amazing answer because it pretends as the origin server for the requests being generated. This is advantageous because it maintains and continues uptime of the server noticeably excessive and assures high quality of service (QoS). It is a useful solution in scenarios where virtual domains have been mapped to a single physical site. Close by forward proxy caching, traffic Server offers with web data solicitations to starting servers because of the guests asking for the data. Reverse proxy caching (often referred to as server acceleration) is exclusive easily for the reason that visitors Server acts as a proxy reminiscence cache on there are two methods to set up transparent caching: switch level and router level. Part of the most vital expectation servers that keep up the data. Traffic Server is designed to behave outwardly as supply server which the patron is trying for connecting to. In a average scenario the promoted hostname of the beginning server eliminates to site visitors Server, which serves client requests instantly, taking information from the genuine beginning server when crucial.

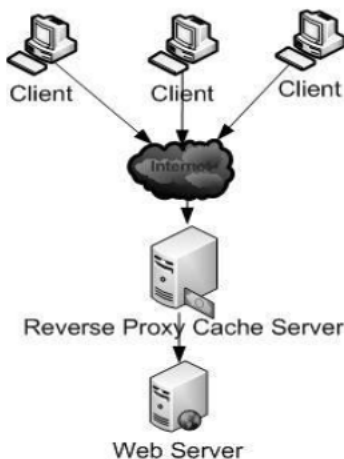


Fig 3. Reverse Proxy Caching

- (e) **Transparent Caching**- The main issue with proxy server strategy is that it requires configuration of the web browser. Transparent web caching alternatively intercepts HTTP requests on the gateway without being seen and redirects them to web cache or clusters. There are two techniques to set up transparent caching: switch level and router level. Router level transparent caching makes use of coverage centered routing to direct requests to correct cache or server. Switch level transparent caching presents change to act as

a committed load balancer. It is more preferable approach because switches generally cost cheaper than routers. As video streaming and rich media downloads preserve up to flood operator networks, with no result in perceive, network operators are analyzing as good as deploying transparent web caching inside of their networks to address a so much wider choice of internet content. The intent is two-fold. The first is to alleviate the network infrastructure and data transfer usage costs associated with over the top (OTT) content and the second is to distinguish their consumer broadband servicing as well as deliver better user performance. By getting rid of any skills slows down related to the web and even the content beginning, caching enables the operator to emphasize their funding being made in the access network and deliver more content material at top speeds.

- (f) **Adaptive Web Caching** -Adaptive Web caching includes replacement algorithms that examine demands produced by the customer and stores the most got to objects within the cache deposit. Adaptive caching includes a couple of, disbursed caches which dynamically become a member of and leave cache companies based on content demand. The general inspiration of Adaptive Caching imagines a decent mesh of self-organizing, overlapping multicast groups of servers that adapt when quintessential to altering conditions. This mesh forms an implicit and scalable hierarchy that's used to effectively diffuse standard web content toward the demand. The two fundamental segments are basic correspondence ways between the neighboring caches and flow of solicitations for data along ways.

- (g) **Push Caching**-Servers choose when and where the articles are reserved; this thought was accepted by push storing. Servers comprehend which purchasers require that data most likely and location it close to them. Knowledge is dynamically mirrored. An assumption about push caching is that the capacity to launch caches traverses administrative boundaries. Push caching is a strong answer for content material vendors. Push-caching has subsequently diminished the actual quantity of network traffic without considerably impacting almost every primary server's load. If handiest primary servers would store duplicated documents then push-caching could be of questionable fee. The abilities of push-caching, however, are that it's vitally easy so as to add extra servers. Proxy-servers, for instance, are perfect contender for tolerating copied things just on the grounds that they're unquestionably



running web caching software, ceaselessly are appended to large disks, and are usually not hidden behind firewalls. Push-caching can flow into the stress from overloaded major servers onto proxy-servers and other servers without exotic an unacceptable load given that all servers caching replicated objects could refuse replacement objects at any time.

- (h) **Active Caching**-The plan makes it feasible for servers to provide cache applets to be connected with records, and obliges proxies to summon cache applets upon cache hits to outfit the imperative processing without reaching the server. Cache applets allow servers to receive the benefit of proxy caching without losing the capability to track user accesses and tailor the content presentation dynamically (Brian D. 1999).

#### 4. LITERATURE SURVEY

V. Sathiyamoorthi et al and Dr. V. Murali Bhaskaran al (2012) define the approach that Web caching is a mechanism used to improve network performance by reducing network traffic, load on the Web Server and delay in accessing the Web page. This is achieved by storing frequently accessed Web pages on proxy cache placed within the network. Caching can be taken location either at purchaser aspect or in the proxy Server. Web proxy cache can potentially support network performance via lowering the number of requests that reaches the server, the quantity of data transferred by means of the network and the delay in getting to Web page. At the point when a requested page is not present in the cache and cache is full then removal of one or more cached documents take pace. The execution of proxy relies on upon page replacement algorithm. The decision by which document is evicted from the cache is depends on different kinds of replacement policies used. Various leaving cache replacement algorithms, which endeavor to diminish different cost measurements, for example, hit rate, byte hit rate, normal latency and aggregate cost.

Negrão et al. think about that the web extends its mind-boggling nearness in our day by day lives, the weight to strengthen the efficiency of web servers raises. A most important optimization process that allows internet-scale web servers to provider purchasers extra efficiently and with lower resource demands consists in caching requested web objects on middle cache servers. On the core of the cache server operation is the replacement algorithm, which is accountable for identifying, in step with a cache substitute policy, the cached pages. that should be removed in order to make space for new pages. Traditional replacement strategies utilized as a part of apply exploit worldly reference territory by utilizing

expelling the minimum of late/as a general rule asked for pages from the cache. On this paper we endorse a brand new resolution that adds a spatial dimension to the cache substitute approach. Our answer is motivated by means of the remark that users most likely browse the net via successively following the hyperlinks on the web pages they seek advice from. This process, called SACS, measures the distance between objects in phrases of the number of links essential to navigate from one object to a different. Then, when alternative takes location, objects which can be distant from probably the most lately accessed pages are candidates for elimination; the closest an object is to an as of late gotten to page, the a great deal more outlandish it is to be evicted. We've carried out a cache server utilizing SACS and evaluated resolution in opposition to different cache substitute systems.

Shamsher Singh et al and G.L. Saini al (2015) Aims to review few caching architectures. These models join intermediary proxy caching, cooperative caching, adaptive caching, caching and active caching. Furthermore, as it has been repeatedly observed, same data is transmitted over same network links time and again to thousands of users. Such redundancies desire the need for caching algorithms that optimally utilize the finite cache space.

Rassul Ayani, Yong Meng Teo and Yean Seen Ng et al. (2003) define that Caching has been used for many years as an potent performance enhancing technique in computer methods. The Least Recently Used (LRU) cache replacement algorithm is a straightforward and by and large utilized plan. Proxy caching is a unique procedure to lower network traffic and delay in numerous WWW applications. By and by, a few attributes of WWW workloads make LRU less engaging in proxy caching. In the recent years, several more efficient replacement algorithms have been suggested. Yet, these advanced algorithms require a great deal of learning about the workloads and are by and large hard to execute. The principle attraction of LRU is its effortlessness. In his work, he present two modified LRU algorithms and compare their performance with the LRU. Their results indicate that the performance of the LRU algorithm can be improved substantially with very simple modifications.

Harshal N. Datir et al. (2011) define the major challenges associated with the Internet is the problem of increased response time caused due to the ever – increasing traffic on the Internet. Numerous arrangements, both hardware and software, have been proposed to defeat this test. The famous hardware choices are to expand the data transmission of the Connection and recreate the net archives at numerous areas. Increasing the bandwidth will increase the data transfer rate, and hence decrease the response time. The replication of

records will facilitate the nearest record to be fetched, minimizing the response time. Just lately, much research has concerned about making improvements to web performance through lowering the bandwidth consumption and WWW traffic. It means that fewer requests and responses need to go over the network and fewer requests for a server to handle.

Williams et al (1996) et al. discussed that measurement outperforms than LFU, LRU and a number of LRU variants in terms of distinct performance measures cache hit ratio and byte hit ratio. In their experiments, they fail to do not forget object frequency in decision making method.

Rachid et al (1988) proposed a procedure referred to as class-based LRU. C-LRU works as recency-founded as good as size-based, aiming to obtain a well-balanced combo between enormous and small documents in the cache, and for this reason, just right performance for each small and massive objects requests. The caching procedure classification-headed LRU is a change of normal LRU.

Cao Pei et al proposed that caching algorithms should address the network cost.

They proposed another calculation known as Greedy Dual-Size that joins region, size and cost together. The cost can be defined as the downloading latency, the network cost or other variables depending on the goal of the algorithm. Authors introduced two versions of the Greedy Dual-Size algorithm, GD-Size (1) and GD-Size (packets). GD Size (1) sets the cost for every report to 1, and GD-Size (bundles) sets the cost for every record to 2+ (size/536). GD Size (1) tries to minimize miss ratio, and GD-Size (packets) tries to curtail the network traffic as a consequence of the misses. The results show that clearly, GD-Size (1) achieves the best hit ratio among all algorithms across traces and cache sizes.

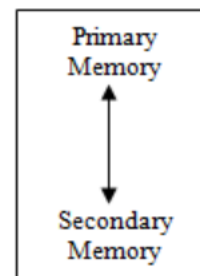
Martin et al (1997) used trace-driven simulations to assess the performance of different cache replacement policies for proxy caches and utilized a trace of client requests to Web proxy in an ISP environment to determine the performance of several present alternative replacement policies. The outcome on this paper are based on probably the most large web proxy workload characterization yet suggested within the literature.

## 5. PROPOSE WORK

The goal of all caching is the same – to avoid reacquiring data that has already been retrieved so that response time is improved and bandwidth utilization minimized. The caching technologies detailed here are designed to achieve that goal by caching data locally and serving it immediately when a request for the data is received. Literature survey shows that main cause of cold cache pollution is due the objects present in the cache that is once accessed

and not accessed again. For that replacement algorithm used in previous work. In literature survey, there are many algorithms like LRU, SLRU, LRU-Distance etc. are used to remove cold cache pollution but they were not able to fully remove cold cache pollution. Our proposed work is to compare these replacement algorithms on the proxy server. We proposed a modified LRU-Distance algorithm to remove cold cache pollution better than previous algorithms.

In our proposed algorithm beginning we divide cache into two parts primary cache and secondary cache in primary cache data store using least recently use manner, data which used least, shift into the secondary memory.



When a user accesses data from the server it put it into primary memory when data is least used it shifted to secondary memory using LRU-D, data take position using LRU-D

Total size allocated for caching of document is CS.

$$PM = CS/2$$

$$SM = CS/2$$

Size of total information stored at any time is IS in primary memory.

Total number of web objects stored at any time is TNO.

Size of object to be inserted is OS.

Top of stack is represented by TOSP for primary and TOSS for secondary memory.

Jumping of web document is represented by JB.

Position of web document returned using index area in secondary memory and Position of web document returned using sequential algorithm in primary memory.

Significance of D is described as SM/2. At start web page contain first location and then second take D position of web page but in this situation there are wastage of memory location so that we do memory compaction after find out all D location we compact

the memory and create new segmentation so that we create another D locations.

Searching for primary memory

Input: Requested objects from Proxy

Output: Total caching

Steps:

- (1) Receive request for object I
- (2) IF I is already present in primary THEN
- (3) Serve request internally from cache
- (4) Go to 13
- (5) ELSEIF I is not present in primary search for secondary memory
- (6) Serve request internally from cache
- (7) Go to 13
- (8) ELSE Serve request from origin server
- (9) IF I can be accommodated in primary cache THEN
- (10) Bring I into cache
- (11) ELSE I cannot be accommodated in cache
- (12) Evict least recently requested object from primary cache and put it into secondary memory Dth location or replace it with I
- (13) END IF
- (14) END IF
- (15) Repeat steps 1 to 12 for next object request

(a) Searching for secondary cache

**Step 1.** If (OS > (CS-IS)) then do the following

- (a). "Delete one or more web documents from bottom of stack, sum of whose size is just greater than or equal to object to be inserted.
- (b). Go to step 2.

**Step 2.** Do the following

- (a). "Insert the document at the distance D from bottom of stack".

(b). IS = IS + OS.

**Step 3.** If (D! = TOS)

JB=1;

**Step 4.** Exit.

**Step5.** If (JB == 0) then do the following

"Put the accessed object at the top of Stack", and go to step 3.

Else

Pos = Index / 2.

**Step6.** "Put the object at the position pos in the stack".

**Step7.** If (pos == TOS) then do

JB = 0.

**Step 8.** Exit.

## 6. COMPACTION ALGORITHM

- (i) Compute the forwarding location for live objects.
- (ii) Keep track of a *free* and *live* pointer and initialize both to the start of heap.
- (iii) If the *live* pointer points to a live object, update that object's forwarding pointer to the current *free* pointer and increment the *free* pointer according to the object's size.
- (iv) Move the *live* pointer to the next object
- (v) End when the *live* pointer reaches the end of heap.
- (vi) Update all pointers
- (vii) For each live object, update its pointers according to the forwarding pointers of the objects they point to.
- (viii) Move objects
- (ix) For each live object, move its data to its forwarding location.

For all designing we consider memory as a doubly linked list so that access of memory and compaction become easy.



Linked list algorithm

p->info = x

current = head

while current is not null and current->info < x

previous = current

current = current->link

if current is head

p->link = head

head = p

else

previous->link = p

p->link = current

## 7. CONCLUSION

The widening gap between processor and memory speeds makes data locality optimization a very important issue in modern cache systems. Computer architects focus on optimizing data cache locality using intelligent cache management mechanisms. In this paper we study several techniques to remove cache pollution at end we propose a new technique to handle cache pollution.

## REFERENCES

- André Pessoa Negrão, Carlos Roque, Paulo Ferreira and Luís Veiga, "An adaptive semantics-aware replacement algorithm for web caching" *Journal of Internet Services and Applications*, a Springer Open Journal, pp. 1-14
- Brian D. Davison (1999). "A Survey of Proxy Cache Evaluation Techniques" April.
- Geetha, K, Ammasai Gounden, N. ; Monikandan, S. (2009). "SEMALRU: An Implementation of modified web cache replacement algorithm" *IEEE Trans.*, 9-11 pp. 1406-1410
- Harshal N. Datir, Yogesh H. Gulhane and P.R. Deshmukh, (2011). "Analysis and Performance Evaluation of Web Caching Algorithms" *International Journal of Engineering Science and Technology (IJEST)*, NCICT Special Issue, pp. 106.
- Jochen Hollmann, Anders Ardo and Per Stenstrom (2007). "Effectiveness of Caching in a Distributed Digital Library System" *Journal of System Architecture*, vol.53, No.7, pp. 403-416
- Jun Pyo Lee and Sung Han Park, (2008) " Video Proxy Server Management Policy using Virtual Caching Technique" *The 23<sup>rd</sup> International Technical Conference on Circuits/Systems, Computers and Communications (ITC-CSCC)*, pp. 393-396
- MAJ Richard Howard and MAJ Bernard J. Jansen, (1998). "A Proxy Server Experiment: an Indication of the Changing Nature of the Web" *IEEE Trans.*,
- Martin F. Arlitt and Carey L. (1997). *Williamson Trace-Driven Simulation of Document Caching Strategies for Internet Web Servers Simulation* vol.68, pp. 23-33.
- Pei Cao, Sandy Irani. *Cost-Aware WWW Proxy Caching Algorithms*. Research Report, Department of Computer Science, University of Wisconsin-Madison.
- Pranay Nanda, Shamsheer Singh and G.L. Saini (2015). "A Review of Web Caching Techniques and Caching Algorithms for Effective and Improved Caching" " Volume 128 – No.10.
- R. Ayani, Yong Meng, (2003). "Cache Pollution in Web Proxy Servers," *IEEE proceedings of the IPDPS*.
- R. Tewari, H. Vin, A. Dan, and D. Sitaram, (1998). "Resource-based caching for web servers" *Proc. of SPIE/ACM Conference on Multimedia Computing and Networking*, San Jose,
- Rassul Ayani, Yong Meng Teo and Yean Seen Ng (2003). "Cache Pollution in Web Proxy Servers" *IEEE proceedings of the IPDPS*,
- S. Sen, J. Rexford, and D. Towsley, (1999). "Proxy prefix caching for multimedia streams," in *Proc. Of IEEE INFOCOM*, New York City, NY
- S.Williams, M.Abrams, C.R. Standbridge, G.Abdulla and E.A.Fox. (1996). *Removal Policies in Network Caches for World-Wide Web Documents*. *Proceedings of the ACM Sigcomm96*, Stanford University.
- Shamsheer Singh, G.L. Saini, (2015). "A Review of Web Caching Techniques and Caching Algorithms for Effective and Improved Caching" *International Journal of Computer Applications (0975 –8887)*, Volume 128 – No.10, pp. 41-45
- V. Sathiyamoorthi, Dr. V. Murali Bhaskaran, "Web Caching Through Modified Cache



Replacement Algorithm” ICRTIT-2012 IEEE,  
pp 483-487

W. Ma and D. H.C. Du, (2002). “Reducing bandwidth requirement for delivering video over wide area networks with proxy server,” IEEE Trans. Multimedia, vol. 4, pp. 539–550

Wei-hsiu Ma and David H. C. Du, (2004,). “Design a Progressive Video Caching Policy for Video Proxy Servers” IEEE Trans. On Multimedia, VOL. 6, NO. 4, pp. 599-610.

Yong Woon Park and Ki Dong Chung, (2001). "A Caching Policy for Continuous Media Objects Based on Logical Caches and Object partitioning" IEEE Tran on Multimedia,

---

**Corresponding Author**

**Jitendra Singh Kushwah\***

Dept of CSE, ITM University, Gwalior (M.P.) India

E-Mail – [jsk2474@gmail.com](mailto:jsk2474@gmail.com)