International Journal of Computer Applications in Engineering Sciences
[VOL I, ISSUE IV, DECEMBER 2011] [ISSN: 2231-4946]

Exploring the Web Caching Method to Improve the Web Efficiency

Amit K. Chauhan[1], Vineet Chauhan[2], Ruchi gupta[3]

[1,2]*Department of Computer Science, Phonics Group of Institutions, Roorkee.*
[3]*Department of Computer Science, Omkarananda Institute of Management & Technology, Rishikesh*

go.chauhan@gmail.com

***Abstract*— In this paper, discuss about the Web Caching technique, Web caching is a familiar technique for reducing access latencies and bandwidth consumption. It is reducing server loads, client request latencies, and network traffic. The paper described the basic Cache Models like front-end, Data end and Distributed Cache application design scenario. The paper explain about the advantage of the web Caching and also explain the web caching solution, functions and the benefits of the web caching, for the web applications. The scenario to improve the efficiency of the web browser and also improve the user response times and reduce bandwidth utilization.**

***Keywords*— Web Caching, HTTP Caching, Proxy Cache, Browser Cache.**

I. Introduction

Cache is memory that is stored very close to the CPU, say on the same chip as the CPU, to allow fast access. Similarly, a disk cache is memory that is used to store frequently accessed disk pages for fast access. Web caching is the storage of Web objects near the user to allow fast access, thus improving the user experience of the Web surfer. Examples of some Web objects are Web pages, images in Web pages, etc.

Web caching is the temporary storage of web objects (for example HTML documents) for later retrieval. There are three significant advantages to web caching: reduced bandwidth consumption (fewer requests and responses that need to go over the network), reduced server load (less requests for a server to handle), and reduced latency (since responses for cached requests are accessible instantaneously, and closer to the client being served). Together, they make the web less expensive and better performing. Caching can be performed by the client application, and is built in to most web browsers.

Caching can also be utilized in the middle, between the client and the server as part of a proxy. Proxy caches are often located near network gateways to reduce the bandwidth required over expensive dedicated internet connections. These systems serve many users with cached objects from many servers. In fact, much of the usefulness is in caching objects requested by one client for later retrieval by another client. For even greater performance, many proxy caches are part of cache hierarchies, in which a cache can inquire of neighboring caches for a requested document to reduce the need to fetch the object directly.

Finally, caches can be placed directly in front of a particular server, to reduce the number of requests that the server must handle. Most proxy caches can be used in this fashion, but this form has a different name (reverse cache, inverse cache, or sometimes http accelerator) to reflect the fact that it caches objects for many clients but from only one server.

The rapid development of the World Wide Web has resulted in major network traffic and congestion. Web data transmission has been almost doubling every six months, and despite efforts for capacity increases demands are not always kept up. Improving response times and access latencies for clients became a quite important and challenging issue. Web caching has been proposed as a technique to reduce both the internet traffic and the access times for requested objects.

Many of the Web caching aspects are originated from the caching idea implemented in various computer and network systems and web caching introduces new issues in Web objects management and retrieval across the network. The overall process of accessing data is no longer dependent on the client/server interaction. A client requests objects residing at a server, but instead of accessing the specified server, its local storage media is checked first. If the requested data resides in local cache is withdrawn from there with no extra network access cost, otherwise the original server needs to be contacted. Web Caches are implemented such that information will reside closer to users since clients retain a local cache for Web objects storage. Therefore, both the load of the origin servers and the network traffic reduces, since upon requesting Web objects the clients can access their local cache instead of fetching the data from their original server. In this paper, the problem of supporting effective Web object caching is addressed and certain evolutionary techniques are proposed.

World Wide Web caching differs from the traditional caching in a distributed file system mainly in its access patterns since Web is orders of magnitude larger than any distributed file system [2]. In addition, Web caching differs from traditional caching due to objects sizes no homogeneity [1]. The most significant research issues in Web caching concern cache replacement strategies as well as cache consistency and confirmation.

Web caching is implemented by proxy server applications developed to support many users. Proxy applications act as an interface between Web clients and servers. Clients make their connection to proxy applications running on their hosts. The proxy connects the server and relays data between the client and the server. At each request, the proxy server is contacted first to find whether it has a valid copy of the requested object. If the proxy has the requested object this is considered a *cache hit*, otherwise a *cache miss* occurs and the proxy must forward the request on behalf of the client. Upon receiving a new object, the proxy services a copy to the end-user and keeps another copy to its local storage. When the cache is full, there is a need for a specific technique to remove some of the current copies in order to store more recently requested objects.

A Web cache is an application residing between Web servers and clients such that it watches requests for information objects identified as html pages, images, documents and files. Web cache servers reply to the users request by sending the requested Web object and by (at the same time) saving a copy for the cache itself. If another request refers to the same object, cache will use the copy it has, instead of asking the original server for it again.

Caching on the web, nowadays, a variety of cache servers are available for the World-Wide Web caching and most of the recent Web servers include caching modules (e.g. Apache, Spinner, Jigsaw, and Purveyor).

Web caching is one of the most misunderstood technologies on the Internet. Webmasters in particular fear losing control of their site, because a cache can 'hide' their users from them, making it difficult to see who's using the site. On the other hand, if you plan your site well, caches can help your Web site load faster, and save load on your server and Internet link. The difference can be dramatic; a site that is difficult to cache may take several seconds to load, while one that takes advantage of caching can seem instantaneous in comparison. Users will appreciate a fast-loading site, and will visit more often.

Web caches can be deployed at different levels:

A. Local cache

The cached copies of Web objects are stored on a local computer. Most popular Web browsers keep a cache of previously accessed objects by default. For example, Internet Explorer calls them "Temporary Internet files." Copies cached locally are only helpful when the user accesses the pages frequently from the same machine.

B. Proxy cache

Proxy servers are separate machines that cache Web objects for multiple users/client computers in the organization. They are computers that sit between the client and the hosting Web server, and they are more efficient than local caches because when a Web object is accessed by any user or computer on your local network, the cached copy is then available to any other user/computer that wants to access that object, without going out to the Internet server to retrieve it. A proxy cache can be integrated with a firewall at the network's edge.

II. WHAT'S A WEB CACHE?

A Web cache sits between Web servers (or origin servers) and a client or many clients, and watches requests for HTML pages, images and files (collectively known as objects) come by, saving a copy for itself. Then, if there is another request for the same object, it will use the copy that it has, instead of asking the origin server for it again. Web Caching is the technique of locally storing the frequently requested documents so that repeated requests to the same document can be serviced from the cache itself. Instead of fetching the document from the remote server. This will considerably decrease the response time involved. Several studies have presented the Web caching as the most beneficial solution for Web performance improvement. Web caching systems can lead to significant bandwidth savings, higher content availability, reducing client latency and increasing server's scalability and availability.

There are two main reasons that Web caches are used:

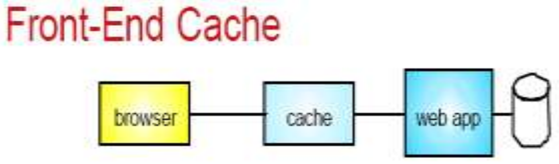
A. To Reduce Latency

Because the request is satisfied from the cache (which is closer to the client) instead of the origin server, it takes less time for the client to get the object and display it. This makes Web sites seem more responsive.

B. To Reduce Traffic

Because each object is only gotten from the server once, it reduces the amount of bandwidth used by a client. This saves money if the client is paying by traffic, and keeps their bandwidth requirements lower and more manageable.

A Web cache is a temporary storage place for files requested from the Internet. After an original request for data has been successfully fulfilled, and that data has been stored in the cache, further requests for those files (a Web page complete with images, for example) results in the information being returned from the cache rather than the original location.



Web caching enhances Web browsing in much the same way. Web server and store a copy of that page on your Local Area Network (LAN). The next time a user requests the Web cache delivers the locally cached copy of the page. The user will experience a very fast download because the request did not have to traverse the entire Internet, the files all came from a local source. Also, the bandwidth that would normally be used to download the Web site is not required and is free for other information retrieval or delivery.

Web caching is a widely deployed technique in the Web architecture that takes advantage of the web object's temporal locality to reduce the user perceived latency and bandwidth consumption. Web caching stores the most popular web objects already requested by users into a pool close to the client-side to avoid requesting again the objects to the original web servers.

Caching is useful for any library. Faster response to user's requests and saved bandwidth are never a bad thing. Caching really makes sense for libraries that feel they must purchase more bandwidth to keep up with increased usage. In such cases a cache server or cache appliance could very likely lower the demand on the existing bandwidth, thus make a costly bandwidth upgrade unnecessary. Suppose an upgrade from a 256K data circuit to a full T1 will increase your monthly Internet bill by $500. A $3,000 investment in caching (and caching solutions often can be implemented for much less than that) will start to show a return after six months.

There are two broad categories of Web caching:

- Forward caching, where copies of Web objects from Internet servers that are frequently accessed by your users are stored on your local network.
- Reverse caching, where copies of the Web objects on your own internal Web servers are stored on a proxy server at the edge of your network to increase performance for outsiders who visit your sites.

III. Caches Models

There are mainly following kinds of models for web caching shown in fig.1 , fig.2 and fig.3 :

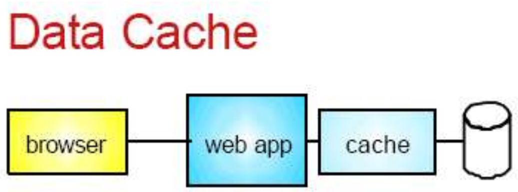


Fig.1 Data Cache Model**.**

Fig.2 Front –End Cache Model

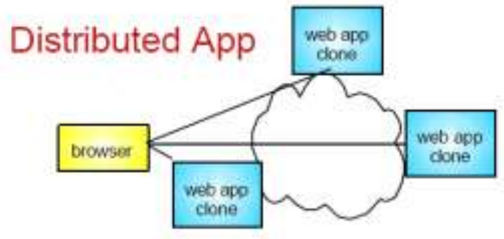


Fig.3 Distributed Cache Model

IV. Variety Of Web Caches

A. *Client Side Caching*

Client-side caching refers to caches that are built into most web browsers, which cache Internet objects for a single user, but from a variety of servers. Caches are found in browsers and in any of the web intermediate between the user agent and the original server. Typically, a cache is located in the browser and the proxy. If we examine the preferences dialog of any modern web browser (like Internet Explorer, Safari or Mozilla), we will probably notice a cache setting. Since most users visit the same web site often, it is beneficial for a browser to cache the most recent set of pages downloaded.

B. *Browser Caches*

The Caching function of a Web browser application. The two most popular Web browsers, Netscape Communicator and Microsoft Internet Explorer, have this function built in. Cached pages are stored on the local hard drive and are only used by that computer. This caching is automatically enabled. Users can control some features such as cache size. If you observe the preferences dialog of any modern browser (like Internet Explorer, Opera or Netscape), you will maybe notice a *cache* setting. This lets you set aside a section of your computer's hard disk to store objects that you've seen, just for you. The browser cache works according to fairly simple rules. It will check to make sure that the objects are fresh, usually once a session (that is, the once in the current invocation of the browser). This cache is useful when a client hits the 'back' button to go to a page they've already seen. Also, if you use the same navigation images throughout your site, they'll be served from the browser cache almost instantaneously.

C. *Proxy Caches*

Web proxy caches work on the same principle, but a much larger scale. Proxies serve hundreds or thousands of users in the same way; large corporations and ISP's often set them up on their firewalls. Because proxy caches usually have a large number of users behind them, they are very good at reducing latency and traffic. That's because popular objects are requested only once, and served to a large number of clients.

Most proxy caches are deployed by large companies or ISPs that wasn't to reduce the amount of Internet bandwidth that they use. Because the cache is shared by a large number of users, there are a large number of shared hits (objects that are requested by a number of clients). Hit rates of 50% efficiency or greater are not uncommon. Proxy caches are a type of shared cache.

D. *Cache Appliance/ Cache Server*

A cache appliance is a hardware and software caching solution all in one unit. A cache server is a software-only solution. The software is installed on an existing server. Unlike a browser cache that only benefits one user, cache servers or appliances are shared and benefit every user in the network. The cache server/appliance sits on the Local Area Network.

V. The Advantages Of Web Caching

Web caching has the following advantages:

- Faster delivery of Web objects to the end user.
- Reduces bandwidth needs and cost. It benefits the user, the service provider and the website vendor.
- Reduces load on the website servers.

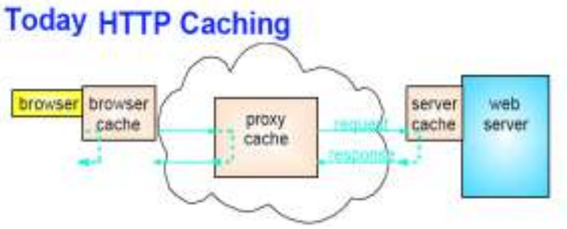


Fig.4 HTTP Caching

VI. Web Cashing Solution

A key performance measure for the World Wide Web is the speed with which content is served to users. As traffic on the Web increases, users are faced with increasing delays and failures in data delivery.

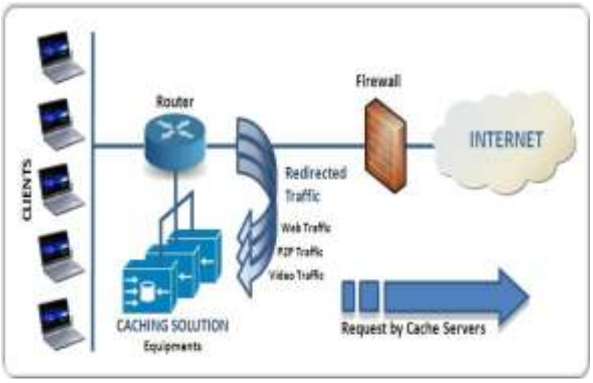


Fig. 5 Web Caching Solution

Web caching is one of the key strategies that have been explored to improve performance. An important issue in many caching systems is how to decide what is cached where at any given time. Solutions have included multicast queries and directory schemes.

A. *Benefits*

- Huge savings of international bandwidth for Service Providers.
- Faster delivery of contents to subscribers and maintain customer satisfaction.
- Ensure customer loyalty through high customer satisfaction.

Caching has been employed to improve the efficiency and reliability of data delivery over the Internet. A close to cache can serve a page quickly even if the make server is busy or the network path to it is congested.

While this argument provides the self-interested user with the motivation to develop caches, it is significance noting that using widespread use of caches also produce a general good, if requests are stop by nearby caches, then fewer go to the source server, reducing load on the server and network traffic to the benefit of all users.

B. *Functions*

- Storage of popular web, video and P2P contents.
- Reduce the download time for popular web, video and P2P contents.
- Reduce the need of always accessing the Internet and able to deliver locally.

VII. Applications

Web cache system can be successfully implemented in the following environments to provide fast and efficient page loading and optimum utilization of network bandwidth.

- E- Business
- Online Reservation (flight booking etc.)
- Educational Institutions.
- Mobile.

VIII. Conclusion

The Web Caching Methods are used to improve the Web efficiency. The users used the web applications, call a web page, the contents of that page can be stored in the browser's cache so it does not require to be re-requested and re-transmitted it. Efficiently using the browser and the web application cache and updated contents of the cache can improve the user reply times and reduce bandwidth exploitation. There are many factors that impact whether or not content can or will be retrieved from the browser cache on repeat visits, including the browser location, web page , the web site.

There are three significant advantages to web caching: reduced bandwidth consumption (less requests and responses that require to go over the network), reduced server load (less requests for a server to handle), and reduced latency. They are compose the web less expensive and better performance , to improve the web efficiency.

References

[1] C. Aggarwal, J. Wolf, and P.S. Yu, "Caching on the World Wide Web," IEEE Transactions on Knowledge and Data Engineering, vol. 11, no.1, pp. 94–107, 1999.

[2] M.A. Blaze, Caching in Large-Scale Distributed File Systems, Princeton University, Ph.D. Thesis, Jan. 1993.

[3] Web caching: Optimizing for internet and Web Traffic (White paper).

[4] Geetika Tewari and Kim Hazelwood. "Adaptive web caching algorithms."

[5] G.N.K. Suresh Babu , S.K .Srivatsav. "Evaluation of Efficient Web Caching and Prefetching Technique for Improving the Proxy Server Performance"

[6] Martin Arlitt, Rich Friedrich. "Performance Evaluation of Web Proxy Cache Replacement Policies ".

[7] Evolution Techniques for web caching January 2002., Distributed and parallel Database Volume 11 issue 1.

[8] C. Aggarwal, J.Wolf, and P.S.Yu, "Caching on WorldWideWeb," IEEE Transactions on Knowledge and Data Engineering, vol. 11, no.1, pp. 94–107, 1999.

[9] M. Arlitt, R. Friedrich, and T. Jin, "Performance evaluation of web proxy cache replacement policies," Performance Evaluation Journal, vol. 39, 2000.

[10] J. Gwertzman and M. Seltzer, "World Wide Web cache consistency," in Proceedings of the USENIX 1996.