Volume 3, Issue 4, April 2013 ISSN: 2277 128X
International Journal of Advanced Research in Computer Science and Software Engineering
Research Paper
Available online at: www.ijarcsse.com

Survey on Integrating Web Caching and Pre-Fetching

T.V. Sree ramya
Department of CSE, Sona College of Technology
Salem-5, Tamil Nadu, India

V. Sathiyamoorthi
Department of CSE, Sona College of Technology
Salem-5, Tamil Nadu,India

Abstract— Due to the phenomenal growth of the World Wide Web network loads and response time for accessing the documents online are increased. Many researchers use Web caching and pre-fetching techniques to overcome these difficulties. With the increased commercialization of web, exceeding the "eight second rule" for downloading web page content is quite annoying to the user, this may result in a significant loss of revenue as many user might switch over to the other sites, if they are not satisfied with the performance of current web page. Web caching and pre-fetching is one of the technique to increase the scalability of web. Thus by integrating the web caching and pre-fetching technique the performance of the web can be efficiently increased.

Keywords- web caching, pre-fetching,response time,eight second rule,world wide web.

I. INTRODUCTION

The World Wide Web is the Internet's most widely used tool for information access and dissemination, but today's users often experience long access latency due to network congestion—particularly during peak hours and big events, such as the Olympic Games. Caching frequently used data at proxies close to clients is an effective way to alleviate these problems. Specifically, caching can reduce load on both the network and servers (by localizing the traffic) and improve access latency (by satisfying user requests from local storage rather than remote servers). Caching proxies have become vital components in most Web systems, and managing proxy cache is critical. Researchers have studied this management task extensively in other systems, such as memory hierarchies and distributed file-sharing systems. However, the Web and the Internet offer several unique challenges in this area, not the least of which are network size and the ever-evolving diversity of technologies and user behavior. Given this, we need novel solutions for deploying Web caching proxies on the Internet. Here, we offer an overview of key management problems for Web proxy caching and pre-fetching and present state-of-the-art solutions to these problems. Our focus is on the distribution of conventional Web objects, such as HTML pages and images, but we also address issues arising from emerging applications

II. WEB ACHING TECHNIQUE

Web caching technology improves client download times and reduces network traffic by caching frequently accessed copies of Web objects close to the clients. The primary research issues in Web caching are where to cache copies of objects (cache placement), how to keep the cached copies consistent (cache consistency), and how to redirect clients to the optimal cache server (client redirection). A proxy is usually deployed at a network's edge, such as at an enterprise network's gateway or firewall. The proxy processes internal client requests either locally or by forwarding the requests to a remote server, intercepting the responses, and sending the replies back to the clients. Because this proxy is shared by internal clients who tend to have similar interests, it's natural to cache commonly requested objects on the proxy.A client-side browser typically retrieves a Web object by initiating an HTTP GET command with the object's address. The browser first attempts to satisfy the request from its local cache; if it fails, it sends the unresolved request to its proxy. If the proxy finds the requested object in its cache, it returns the object to the client; otherwise, the request is forwarded to the object's origin server, which as the authoritative source of the requested object returns the object to the proxy. The proxy then relays the object to the client and, if needed, saves a copy in its cache. If a request is satisfied from the proxy cache, it is called a *cache hit*; otherwise, it's a *cache miss.* On the other hand, the delivery of diverse streaming media contents on IP networks in a cost effective manner, while maintaining high quality, is still very challenging. Thus, today most of Internet media objects are still accessed via downloading or pseudo streaming instead of streaming, which cause roughly 56% and 32% of wasted bandwidth according to study. In a web service environment, a continuous streaming session (often with a duration of minutes or hours, compared to milliseconds or seconds for traditionalWeb pages) keeps consuming network bandwidth and disk bandwidth on the hosting server. Multiple concurrent streaming sessions can easily exhaust the available network bandwidth and overload the media content server. Placing multimedia objects closer to clients is an effective solution that will relieve the network bottleneck and reduce the loadon the media content server.

A. Types of Web Caching

Web caching keeps a local copy of Web pages in places close to the end user. Caches are found in browsers and in any of the Web intermediate between the user agent and the origin server. Typically, a cache is located in client (browser cache),

1) *Browser cache:* It is located in the client. The user can notice the cache setting of any modern Web browser such as Internet Explorer, Safari, Mozilla Firefox, Netscape, and Google chrome. This cache is useful, especially when users hit the "back" button or click a link to see a page they have just looked at. In addition, if the user uses the same navigation images throughout the browser, they will be served from browsers' caches almost instantaneously.

2) *Proxy server cache:* It is found in the proxy server which located between client machines and origin servers. It works on the same principle of browser cache, but it is a much larger scale. Unlike the browser cache which deals with only a single user, the proxies serve hundreds or thousands of users in the same way. When a request is received, the proxy server checks its cache. If the object is available, it sends the object to the client. If the object is not available , or it has expired, the proxy server will request the object from the origin server and send it to the client. The object will be stored in the proxy's local cache for future requests.

3) *Origin server cache:* Even at the origin server, web pages can be stored in a server-side cache for reducing the need for redundant computations or database retrievals. Thus, the server load can be reduced if the origin server cache is employed. Availability of Web access logs files that can be exploited as training data is the main motivation for adopting intelligent Web caching approaches. The second motivation, since Web environment changes and updates rapidly and continuously, an efficient and adaptive scheme is required in Web environment. The machine learning techniques can adapt to the important changes through a training phase. Although there are many studies in Web caching, enhancement of Web caching performance using intelligent techniques is still fresh. Recent studies have shown that the intelligent approaches are more efficient and adaptive to the Web caching environment compared to other approache

B. *Need For Web Caching*

Caching helps bridge the performance gap between local activity and remote content. In the short term, caching helps to improve Web performance by reducing the cost and end-user latency for Web access. In the long term, even as bandwidth costs continue to drop and higher end-user speeds become available; caching will continue to reap benefits for the following reasons:

_ *Bandwidth will always have some cost*.The cost of bandwidth will never reach zero, even though increased competition, a growing market, and economies of scale reduce end-user costs.The cost of bandwidth a the core has stayed relatively stable, requiring ISPs to implement methods such as caching to stay competitive and reduce core bandwidth usage so that edge bandwidth costs can be low.

_ *Non uniform bandwidth and latencies will persist*. Because of physical limitations such as environment and location as well as financial constraints, there will always be variations in bandwidth and latencies. Caching can help to smooth these effects.

_ *Network distances are increasing*. Firewalls, other proxies for security and privacy, and virtual private networks for telecommuters increase the number of hops through which content must travel and slow Web response times.

_ *Bandwidth demands continue to increase*. Growth in the user base, in popularity of high-bandwidth media, and in user expectations of faster performance guarantee that demand for bandwidth will not end.

_ *Hot spots in the Web will continue*. Intelligent load balancing can alleviate problems when high user demand for a site is predictable, but a Web site's popularity can also come as a result of current events, desirable content, or word of mouth. Distributed Web caching can help alleviate these "hot spots" resulting from flash traffic loads.

_ *Communication costs exceed computational costs*. Communication is likely to always be more expensive (to some extent) than computation. We use memory caches because CPUs are much faster than main memory. Likewise, we will continue to use caches as computer systems and network connectivity both get faster

III. PRE-FETCHING

Web caching technology has been widely used to improve the performance of the Web infrastructure and reduce user perceived network latencies. Proxy caching is a major Web caching technique that attempts to serve user Web requests from one or a network of proxies located between the end user and Web servers hosting the original Copies of the requested objects. This paper surveys the main technical aspects of proxy caching and discusses recent developments in proxy caching research including caching the "uncacheable" and multimedia streaming objects, and various adaptive and integrated caching approaches. than the Web servers that publish the original copies of the requested objects . Recent years have seen significant growth in the Web caching literature and a large number of commercial offerings from both established network vendors and start up companies that exclusively focus on caching-related hardware and software solutions (see http://www.web-caching.com/ for a partial list of Web caching products). In effect, caching is ubiquitous in today's computing environment. All of the major Internet backbone providers and Internet service providers (ISPs) now implement Web caching as part of their infrastructure, often transparent to end users and service subscribers. Many medium-to-large enterprises are using a variety of caching products and services to improve the network performance and reduce networking connection costs. Many end-user programs, including Web browsers, also maintain their local caches to reduce user-perceived network latencies. According to the location of caches, Web caching systems can be classified into three types: browser caches, proxy caches, and surrogate caches. Browser caches are located within user browser programs. Surrogate caches are typically located near the Web servers and are owned and operated by the Web content providers . Proxy caches are located between end-user client sites and original Web servers, typically closer to the clients than to the servers. Proxy caches are typically configured and operated by ISPs and enterprises operating internal networks that are connected to the Internet. This paper mainly focuses on proxy caching for the following four reasons. First, a dominant portion of

the current caching literature is directly related to various technical aspects of proxy caches. Although surveys on Web caching technology exist in the literature new developments in proxy caching and its extended applications in areas such as caching "uncacheable" Web objects and differentiated services are of important practical significance, calling for a new updated survey. Second, from the point of view of system deployment, proxy caching does not require major changes in the networking environment and can achieve the economy of scale because multiple users are served. In addition, proxy caching does not rely on any major changes (e.g., with respect to protocols) to original Web servers and, in most cases, does not require much end-user configuration efforts.

The amount of traffic over the Internet has experienced tremendous growth in recent years largely due to the wide adoption of the World Wide Web technologies and the resulting explosion of Web-based content development and dissemination The Internet bandwidth capacity expansion, on the other hand, is lagging behind, making the Web a major performance bottleneck. The gap between the Web infrastructure capacity and demand will continue to exist, if not expand, as information search and business transactions are being increasingly conducted over the Web. Another compounding factor is related to the recent developments in the Web technologies such as Web services, which will potentially bring in new classes of distributed applications in large numbers that will communicate among one another over the Internet, consuming network bandwidth. caching is an established approach to meet the important Web capacity challenge and address related issues such as user-perceived network latencies. Broadly speaking, caching can be defined as serving user Web requests from places other than the Web servers that publish the original copies of the requested objects Recent years have seen significant growth in theWeb caching literature and a large number of commercial offerings from both established network vendors and start up companies that exclusively focus on caching-related hardware and software solutions (see http://www.web-caching.com/ for a partial list of Web caching products). In effect, caching is ubiquitous in today's computing environment. All of the major Internet backbone providers and Internet service providers (ISPs) now implement Web caching as part of their infrastructure, often transparent to end users and service subscribers Many medium-to-large enterprises are using a variety of caching products and services to improve the network performance and reduce networking connection costs. Many end-user programs, including Web browsers, also maintain their local caches to reduce user-perceived network latencies. According to the location of caches, Web caching systems can be classified into three types: browser caches, proxy caches, and surrogate caches. Browser caches are located within user browser programs. Surrogate caches are typically located near the Web servers and are owned and operated by the Web content providers. Proxy caches are located between end-user client sites and original Web servers, typically closer to the clients than to the servers. Proxy caches are typically configured and operated by ISPs and enterprises operating internal networks that are connected to the Internet. This paper mainly focuses on proxy caching for the following four reasons. First, a dominant portion of the current caching literature is directly related to various technical aspects of proxy caches. Although surveys on Web caching technology exist in the literature new developments in proxy caching and its extended application in areas such as caching "uncacheable" Web objects and differentiated services are of important practical significance, calling for a new updated survey. Second, from the point of view of system deployment, proxy caching does not require major changes in the networking environment and can achieve the economy of scale because multiple users are served. In addition, proxy caching does not rely on any major changes (e.g., with respect to protocols) to original Web servers and, in most cases, does not require much end-user configuration efforts.

IV. RELATED WORK

The WWW continues to grow at an amazing rate as an information gateway and as a medium for conducting business. Web mining is the extraction of interesting and useful knowledge and implicit information from artifacts or activity related to the WWW. Based on several research studies we can broadly classify Web mining into three domains content , structure and usage mining . This work is concerned with Web usage mining. Web servers record and accumulate data about user interactions whenever requests for resources are received. Analyzing the Web access logs can help understand the user behavior and the web structure. From the business and applications point of view, knowledge obtained from the Web usage patterns could be directly applied to efficiently manage activities related to e business, e-services, e-education and so on. Accurate Web usage information could help to attract new customers, retain current customers, improve cross marketing/sales, effectiveness of promotional campaigns, track leaving customers and find the most effective logical structure for their Web space. User profiles could be built by combining users' navigation paths with other data features, such as page viewing time, hyper- link structure, and page content. What makes the discovered knowledge interesting had been addressed by several works. Results previously known are very often considered as not interesting. So the key concept to make the discovered knowledge interesting will be its novelty or unexpected appearance. Whenever a visitor accesses the server, it leaves the IP, authenticated user ID, time/date, request mode, status, bytes, referrer, agent and so on. The available data fields are specified by the HTTP protocol. There are several commercial software that could provide Web usage statistics. These stats could be useful for Web administrators to get a sense of the actual load on the server. However, the statistical data available from the normal Web log data files or even the information provided by Web trackers could only provide the information explicitly because of the nature and limitations of the methodology itself. Generally, one could say that the analysis relies on three general sets of information given a current focus of attention past usage patterns, degree of shared content and inter-memory associative link structures. After browsing through some of the features of the best trackers available it is easy to conclude that rather than generating statistical data and texts they really do not help to find much meaningful information .For small web servers, the usage statistics provided by conventional Web site trackers may

be adequate to analyze the usage pattern and trends. However as the size and complexity of the data increases, the statistics provided by existing Web log file analysis tools may prove inadequate and more intelligent knowledge mining techniques will be necessary. In the case of Web mining, data could be collected at the server level, client level, proxy level or some consolidated data. These is collected etc. The usage data collected at different sources represent the navigation patterns of different segments of the overall Web traffic, ranging from single user, single site browsing behavior to multi-user, multi-site access patterns. Web server log does not accurately contain sufficient information for inferring the behavior at the client side as they relate to the pages served by the Web server.

To demonstrate the efficiency of the proposed frameworks, Web access log data at the Monash University's Web site were used for experimentations. The University's central web server receives over 7 million hits in a week and therefore it is a real challenge to find and extract hidden usage pattern information. To illustrate the University's Web usage patterns, average daily and hourly access patterns for 5 weeks are shown. The average daily and hourly patterns nevertheless tend to follow a similar trend the differences tend to increase during high traffic days (Monday - Friday) and during the peak hours (11:00 - 17:00 Hrs). Due to the enormous traffic volume and chaotic access behavior, the prediction of the user access patterns becomes more difficult and complex.

Previous work presented approaches for discovering and tracking evolving user profiles. It also describes how the discovered user profiles can be enriched with explicit information need that is inferred from search queries extracted from Web log data. Profiles are also enriched with other domain-specific information facets that give a panoramic view of the discovered mass usage modes. An objective validation strategy is also used to assess the quality of the mined profiles, in particular their adaptability in the face of evolving user behavior. However the previous work concentrated only on user profiling at the application level data but not associating it to the web server. The user profile maintained by the web server enriches the user's session of authenticity at different spatial entities. The previous work used conventional web log profile analyzers weakened at the linkage of web user profiling to its server.

V. INTEGRATING WEB CACHING AND PRE-FETCHING

Web proxy caching and pre-fetching are the most popular techniques which play a key role in improving the Web performance. Since the Web proxy caching exploits the temporal locality and the web pre-fetching utilizes the spatial locality of the Web objects, Web proxy caching and pre-fetching can complement each other. Thus, combination of the caching and the pre-fetching helps on improving hit ratio and reducing the user-perceived latency. However, if the web caching and pre-fetching are integrated inefficiently, this might cause increasing the network traffic as well as the Web servers' load. Moreover, the cache space is not used optimally .Therefore, the pre-fetching approach should be designed carefully in order to overcome these limitations.

Basically, the web pre-fetching requires two steps: anticipating future pages of users and preloading them into a cache. This means the web pre-fetching involves also the caching. However, the web caching and pre-fetching are addressed separately by many researchers in the past. It is important to take into consideration the impact of these two techniques combined together. Few studies were discussed integration of web caching and web pre-fetching together. studied effect of a combination of caching and pre-fetching on end user latency. They concluded that the combination of web caching and pre-fetching can potentially improve latency up to 60%, whereas web caching alone improves the latency up to 26%. suggested an application of web log mining to obtain web-document access patterns and used these patterns to extend the well-known GDSF caching policies and pre-fetching policies. proposed cache replacement algorithm called IWCP for integrating Web caching and Web pre-fetching in client-side proxies. They formulated a normalized profit function to evaluate the profit from caching an object either a non implied object or an implied object according to some pre-fetching rule.

Similar to used ANN in both pre-fetching policy and Web cache removal decision. This approach depended on the keywords of URL anchor text to predict the user's future requests .The most significant factors (recency and frequency) were ignored in web cache replacement decision. Moreover, since the keywords extracted from web documents were given as inputs to ANN, applying ANN in this way may cause extra overhead on the server. Presented a compact set of algorithms for integrating web caching and pre-fetching for wireless local area network, including sequence mining based prediction algorithm, context-aware pre-fetching algorithm and profit-driven caching replacement policy. proposed a framework for combining Web caching and pre-fetching on mobile environment. They proposed hybrid technique (Rough Neuro-PSO) based on combination of ANN and PSO for classification Web object. Then, rules from log data are generated by Rough Set technique on the proxy server. In pre-fetching side, pre-fetching approach based on XML is suggested to be implemented on mobile device to handle communication between client and server.

In summary, the previous works integrated the web pre-fetching with caching; However, these approaches are still not efficient enough . Most previous works used association rules for pre-fetching approach, which are inaccurate and inefficient since these works predict a particular page depending on patterns observed from all users' references. Moreover, these approaches employ the conventional replacement policies that are not efficient in web caching.

VI. CONCLUSION

Web caching and prefetching are two effective solutions to lessen Web service bottleneck, reduce traffic over the Internet and improve scalability of the Web system. The Web caching and prefetching can complement each other since the web caching exploits the temporal locality for predicting revisiting requested objects, while the web prefetching utilizes the spatial locality for predicting next related web objects of the requested Web objects. Thus,

combination of the web caching and the web prefetching doubles the performance compared to single caching. This paper reviews principles and some the existing web caching and prefetching approaches. Firstly, we have reviewed principles and existing works of web caching. This includes the conventional and intelligent web caching. Secondly, types and categories of prefetching have presented and discussed briefly. Moreover, the history-based prefetching approaches have been concentrated and discussed with review of the related works for each approach in this survey. Finally, this survey has presented some studies that discussed integration of web caching and web prefetching together.

REFERENCES

[1] H.T. Chen, *Pre-fetching and Re-fetching in Web caching systems: Algorithms and Simulation*, Master Thesis, TRENT UNIVESITY, Peterborough, Ontario, Canada(2008).

[2] T.Chen, "Obtaining the optimal cache document replacement policy for the caching system of an EC Website", *European Journal of Operational Research*.181(2),(2007), pp. 828. Amsterdam.

[3] T. Koskela, J. Heikkonen, ,and K. Kaski, (2003). "Web cache optimization with nonlinear model using object feature", *Computer Networks journal, elsevier* , 43(6), (2003), pp. 805-817.

[4] J. Cobb, and H. ElAarag, "Web proxy cache replacement scheme based on back-propagation neural network", *Journal of System and Software*, 81(9), (2008), pp. 1539-1558.

[5] R. Ayani, Y.M. Teo, and Y.S. Ng, "Cache pollution in Web proxy servers", *International Parallel and Distributed Processing Symposium (IPDPS'03)*, 22-26 April 2003, pp.7.

[6] A.K.Y. Wong, " Web Cache Replacement Policies: A Pragmatic Approach", *IEEE Network magazine*, 20(1), (2006), pp.28–34.

[7] C. Kumar and J.B. Norris, "A new approach for a proxy-level Web caching mechanism", *Decision Support Systems, Elsevier*, 46(1), (2008), pp.52-60.

[8] I. R. Chiang, P. B. Goes, and Z. Zhang, "Periodic cache replacement policy for dynamic content at application server", *Decision Support Systems, Elsevier*, 43 (2), (2007), pp. 336- 348.

[9] H.k. Lee, B.S. An, and E.J. Kim, "Adaptive Prefetching Scheme Using Web Log Mining in Cluster-Based Web Systems", *2009 IEEE International Conference on Web Services (ICWS)*, (2009), pp.903-910.

[10] L. Jianhui, X. Tianshu, Y. Chao. "Research on WEB Cache Prediction Recommend Mechanism Based on Usage Pattern", *First International Workshop on Knowledge Discovery and Data Mining(WKDD)*, (2008), pp.473-476.

[11] A. Abhari, S. P. Dandamudi, and S. Majumdar , "Web Object-Based Storage Management in Proxy Caches", *Future Generation Computer Systems Journal* , 22(1-2), (2006). pp. 16-33.

[12] T. M. Kroeger, D. D. E. Long, and J. C. Mogul, "Exploring the bounds of web latency reduction from caching and prefetching", *Proceedings of the USENDC Symposium on Internet Technology and Systems*, (1997), pp. 13-22.

