Efficient Document Replacement Policy by Web Site Popularity

Jun-Tak Han
Dept.of Computer Engineering
Hanzhong University, Kangwon, Korea

ABSTRACT

General replacement policy includes document-based LRU or LFU technique and other various replacement policies are used to replace the documents within cache effectively. But, these replacement policies function only with regard to the time and frequency of document request, not considering the popularity of each web site. In this paper, we present the document replacement policies with regard to the popularity of each web site, which are suitable for modern network environments to enhance the hit-ratio and efficiently manage the contents of cache by effectively replacing documents on intermittent requests by new ones.

Key words: *replacement policy, Web cache, hit ratio*

1. INTRODUCTION

As the use of the Internet gets increased[1], various problems occur as follows; the increase of network traffic, load of server and delay of search. To solve effectively the problems and to enhance the performance of the Internet, a Web cache has been used[2]. As a result of it, the whole network traffic has been reduced and reaction time has been effectively improved. If a user requests a document and if it exists in a Web cache, the Web cache requested will transmit the document to the user. However, if it doesn't exist in it, the Web cache will ask for new document from a server, copy it into the cache and return it to the user. And then if the capacity of the cache exceeds, a replacement policy for changing effectively the existing document into new document will be used. In general, LRU(Least Recently Used) and LFU(Least Frequently Used) policy are used as a Web cache replacement policy. However, the above policies have considered the request time and the size and frequency of document in the cache, not considering the popularity of a web site. To consider frequently requested site, the site's object can be stored in a cache to raise hit-ratio of the cache and to control effectively the content of cache. At the same time, a document seldom requested can effectively be replaced. Therefore, the paper suggests a document replacement considering popularity of a site, which is a cache policy proper for the modern network environment. The policy can reduce network traffic by lessening the quantity of document which a server requests.

2. RELATED WORKS

The Web cache replacement policy can lessen access to a Web server by replacing effectively pages and improve the performance of the Internet by raising hit-ratio of cache when a client requests a page. In this chapter, the existing replacement policies are surveyed and the more effective Web cache policy is sought for.

2.1 LRU replacement policy

LRU(Least Recently Used)[4] is the most general replacement policy. When the capacity of a Web cache exceeds and a document needs to be replaced, the first oldest document of the existing documents is replaced by a new document.

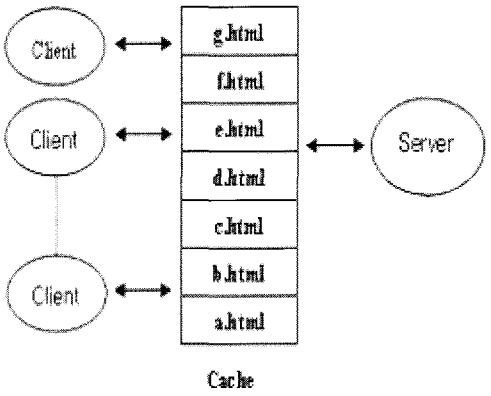


Fig. 1. A basic cache structure on the Web

Fig. 1. shows a basic cache structure. According to the structure, when a new document requested by a client isn't in a cache, the cache requests a server to send the document and copies the new document sent to the cache. And then it transmits the document to the client. In the case that the capacity of the cache exceeds, a document replacement policy will be carried out. To use LRU policy in here, suppose that document A, B, C, D, E, F and G are stored in order of reference time as in Table 1. and newly requested page is document H, document G, whose reference time is oldest among the documents, will be replaced by new document H as in Fig. 2. However, LRU Replacement Policy has some

* *Corresponding author. E-mail : okebary513@hanzhong.ac.kr*
Manuscript received Jan. 26, 2007 ; accepted Mar. 06, 2007

problems. As 2/3 of total-reference have accesses to a document just once on the Web, the Web can be overflowed by the documents which has been referred once.

In addition, as the information on all pages' reference time, whether a document exists or not, should be kept, the capacity of memory is requested a lot.

Table 1. Reference time of documents

<Document>	<Reference Time>
Document G	
Document F	
Document E	
Document D	
Document C	
Document B	
Document A	

cache size = 7

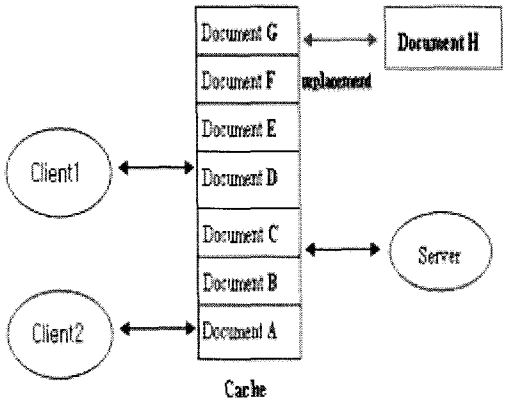


Fig. 2. Document Replacement Policy

2.2 LFU Replacement Policy

Unlike LRU Replacement Policy, LFU is that least frequently used document is replaced by newly requested document when a replacement occurs. It means that a document whose number of reference is least is replaced by new document. Suppose that newly requested page is document H, and document A, B, C, D, E, F and G are stored in a cache in order of number of reference as in Table 2. , document A whose number of reference is least, that is, whose frequency of reference is least as in Fig. 3. will be replaced by new document H.

Table 2. Number of reference of documents

<Document>	< Number of Reference >
Document G	15
Document F	12
Document E	10
Document D	8
Document C	6
Document B	2
Document A	1

cache size = 7

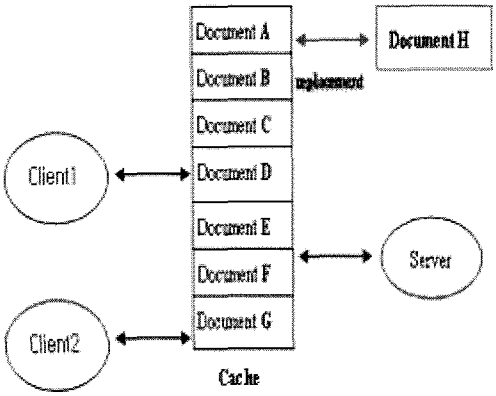


Fig. 3. Document Replacement Policy using LFU

However, as a document which has least number of reference should be replaced according to LFU Replacement Policy, it can result in the pollution of a cache. To avoid this state, a threshold parameter by which a document should be referred during a given period should be set up and the page used less than the parameter should be removed.

2.3 Other Document Replacement Policies

To complement general LRU or LFU replacement policy, various replacement policies have appeared; Size replacement policy by which a document with the most capacity is replaced, GD-Size(Greedy Dual-Size) replacement policy[5] mixing Size Policy with LRU policy, LRFU Replacement Policy[6] mixing the merit of LRU with that of LFU, SLRU(Size-adjusted LRU) Replacement Policy[7] which uses a document's frequency of reference in order to minimize the removal of a big-size and frequently required document when a document is replaced by the size of documents[7], and a Replacement Policy which stores all documents by the size[8]. A replacement policy based on a site can consume the capacity of a cache because objects requested once keep being stored by the site. Let us suppose that one frequently requested object exists in a site, the replacement possibility of the object will be higher than that of any other objects in other sites in the replacement policy. As a result of it, a cache's hit-ratio will be diminished.

3. WEB CACHE REPLACEMENT POLICY

The goal of Web cache replacement policy is to diminish access to Web servers through effective replacement of pages and to enhance the Internet's performance by increasing hit-ratio when a user requests a document[9]. The goal of this paper is to increase hit-ratio..

3.1 The suggested Web cache structure

According to general structure of Web cache, each object is stored in a cache. A cache's structure suggested in this paper is the structure in which a site and an object are mixed. The popularity of a site is considered and the frequently requested site stores the requested objects by the site while seldom required site stores only the requested object in that site. In other words, the possibility of access to a site whose objects are frequently requested by clients is higher. When a site which can't satisfy a threshold exists in a cache, it stores just an object which a client has requested. If the site is stored in a cache, the hit-ratio of this site can be raised. When a replacement policy is carried out due to the excess capacity of a cache, the least frequently requested object among objects within a site which satisfy a threshold or the least frequently requested object among objects which is seldom stored by the site in a cache is removed. And then newly requested object is inserted. If the frequency of requesting objects stored in a cache is lower than the frequency of requesting objects stored by the site, remove all objects stored by the site to reduce the pollution of the cache and secure the room for the cache to reduce the frequent use of replacement policy and to reduce latency time.

Even though it is a frequently requested site, the site will not store objects by the site but it will store a corresponding object in the case that a client requests the specified object in the site.

3.2 Document Replacement Policy

Existing replacement policies have been based on an object. On the other hand, policies in this study have set a threshold to consider frequently requested sites. If a site satisfying the threshold exists in a cache, the objects within the site will be stored by the site.

Table 3. Requested documents

request	requested document
request 1	http://A/a.html
request 2	http://B/a.html
request 3	http://C/a.html
request 4	http://A/b.gif
request 5	http://D/a.jpg
request 6	http://A/c.html
request 7	http://E/d.html
request 8	http://F/a.html

cache size = 7

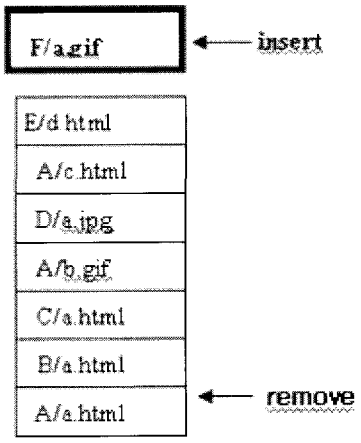


Fig. 4. Existing Replacement Policy

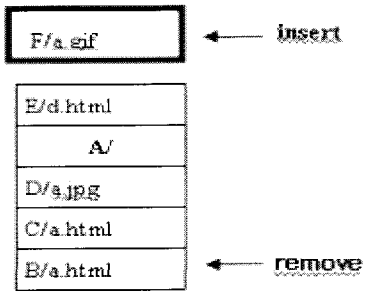


Fig. 5. The Suggested Replacement Policy

Table. 3. shows that documents of A site are frequently requested. When a cache's capacity is 7 and when documents are requested as in Table 3. Fig. 4. shows that request 1, 2, 3, 4, 5, 6 and 7 are stored into a cache. However, if a client request 8, A/a.html requested to request 1 will be removed due to excess capacity of the cache, and newly requested F/a.html document of request 8 will be inserted. The documents of popular A site, which are frequently requested, are removed. However, in Fig. 5. a.html of B site is removed when a document is replaced due to excess capacity of a cache. That's why documents are stored by the site and documents of less popular site are removed.

4. IMPLEMENTATION OF THE REPLACE-MENT POLICY AND PERFORMANCE EVALUATION

The Web server log files of Dongguk University were analyzed to build up the replacement policy considering popularity of site, as suggested in this study. The number of total requests were 64,993. Fig. 6. shows the statistics of the number of requests according to the site.

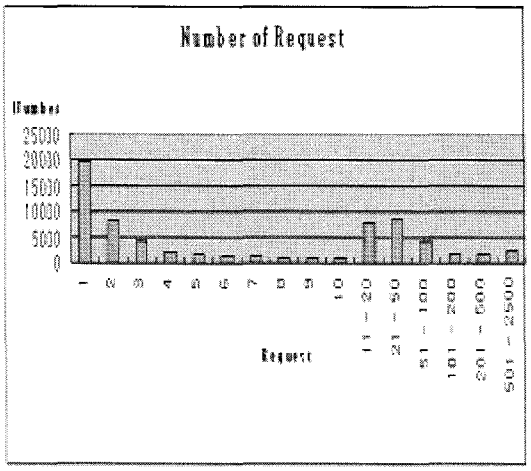


Fig. 6. The analysis of Web server of Dongguk University

Fig. 6. shows 500, 1000 and 2000 input data composed of after the patterns of sites had been analyzed.

The input data by the log statistics of Web servers of Dongguk University consisted of site and object as in Fig. 7.

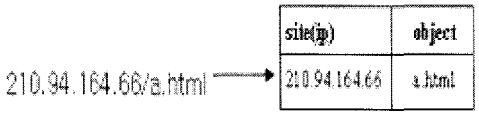


Fig. 7. Input Data

The test for implementation was performed under the environment of CPU P-4 1.7GHz, RAM°| 256MB and HDD 80GB. The Windows XP professional was used as an operating system. It was assumed that TTL(Time To Live) was consistent and a cache's size and the requested document's size were equal. After making input data into a log by implementing a cache, hit-ratio of LRU replacement policy used in the cache was compared to hit-ratio of document replacement policy considering the popularity of sites suggested in the study. To be objective, 500, 1000 and 2000 requests were surveyed, respectively. The changes of hit-ration of LRU replacement policy and document replacement policy considering the popularity of sites were examined.

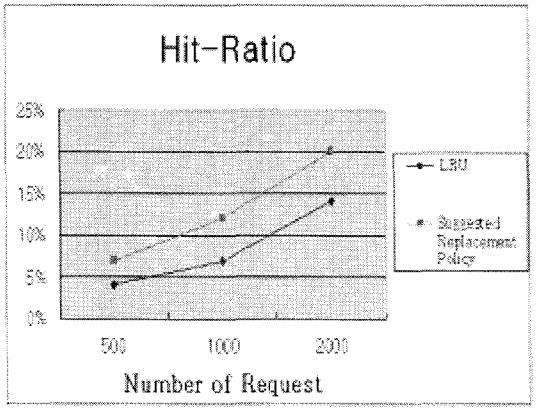


Fig. 8. Comparison between LRU replacement policy and the suggested replacement policy

Fig. 8. shows that the hit-ratio of replacement policy considering the popularity of sites was enhanced by about 6% more than that of existing LRU replacement policy. In other words, if several objects in the frequently requested site were stored in a cache by the site, the possibility of next access would be higher because the site had been frequently requested and the possibility by which a client would request an object from the popular site would be higher. Therefore, the hit-ratio of the suggested replacement policy could be higher than that of the existing replacement policy.

5. CONCLUSION AND FUTURE WORK

In this paper, we presented the document replacement policies with regard to the popularity of each web site, which are suitable for modern network environments to enhance the hit-ratio and efficiently manage the contents of cache by effectively replacing documents on intermittent requests by new ones. The hit-ratio raised by using the replacement policy considering the popularity of a site suggested in the paper could help to reduce the network traffic, as well as to reduce the request from a server. Nevertheless, even the replacement policy suggested in this paper has a problem of consistency of cache. The problem of consistency between objects in the popular site and objects in a server will be solved by Client Polling Method, Invalidation Protocol Method, TTL, Active Refresh Method[10] and Bloom Filter.

REFERENCE

[1] Time Berners-Lee, Robert Cailliau, Ari Luotonen, Henrik Frystyk Nielsen, and Arthur Secret, "The World-Wide-Web" **Communications of the ACM**, vol, 37, No. 8 pp.76-82, Aug, 1994

[2] Ari Lutotonen and Kevin Altis, "World-Wide-Web Proxies" **The 1st Inter. Conf, on the WWW**, Geneva, May, 1994

[3] P.B. Danzig, R, S. Hall, M.F. Schwartz "A Case for Caching File Objects Inside Internetworks" in **Proceeding of the ACM SIGCOMM**, Sep, 1993

[4] Levy, H,; Morris, R,J,T; "Extra analysis of Bernoulli superposition of streams into a least recently used cache" **IEEE Trans on, Software Engineering,** Vol.21 Issue:8, Aug. pp.682-688, 1995

[5] Murta,C.D; Almeida, V.A.F. "Using performance maps to understand the behavior of Web caching policies" **The 2th Proceeding of IEEE Workshop, Internet Applications**, pp50-56, 2001

[6] D. Lee et al., "LRFU: A Spectrum of Policies that Subsumes the Least Recently Used and Least Frequently Used policies," **IEEE Trans on, Computers**, Vol.50, No.12. pp.1352-1361, 2001

[7] Kai Cheng; Kambayashi, Y,; "LRU-SP: a size-adjusted and popularity-aware LRU replacement algorithm for web caching" **The 24th Inter, Conf. Computer Software and Applications** pp.48-53, 2000

[8] Kin Yeung Wong; Kai Hau Yeung "Site-based approach to Web cache design" **IEEE Internet Computing**, Vol.5, 2001

[9] Stephen Williams, Marc Abrams, Charles R. Standridge, Ghaleb Abdulla, Edward A. Fox, "Removal Policy in Network Caches for World-Wide-Web Documents", in **Proceeding of the ACM SIGCOMM**, Aug, 1997

[10] Gang Zhang; Yantai Shu; Zheng Zhao; Zhijie Guan;

Yang,O.W.W. "An active refresh method in Web caching" **Electrical and Computer Engineering, Canadian Conf.** on, Vol.1, 2001

[11] "Industry research on actual state research of game industry laying stress on technological level", **Dongguk Univ., Industry technical cooperation center**, 2002

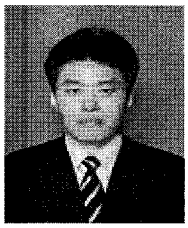
[12] Nam-Jae Lee, Duck-Woo Seo, Hoon-Sung Kwak, "Distributed Server Configuration Scheme for Internet Online Game" **The 2th Proceeding of the Korea Game Society**, pp. 31-36, 2001

[13] Wen tong Cai, Percival Xavier, Stephen J. Turner, Bu-Sung Lee, "A Scalable Architecture for Supporting Interactive Games on the Internet" **The 16th Workshop on Parallel and Distributed Simulation**", pp.60, May 12-15, 2002

[14] Jong-Jung Woo "A Study on an Adaptive Cache Algorithm for Web Server to Reduce Loads" **The Journal of Basic Science**, Vol.20, pp.77-92, 2003

[15] Bahn, H "Web cache management based on the expected cost of web objects" **Information and Software Technology**, Vol.47, No.9, pp.609-621, 2005

[16] Chen. T "A new cache document replacement policy considering the contribution to sales" **Electronic Commerce Research and Applications**, Vol.4, No.3, pp.283-297, 2005



Jun-Tak Han

He received the M.S. degree in Computer Science from Hanyang University, in 1994, and Ph.D. in Computer Engineering from Dongguk University, in 2003. Since 1995 he is an associate professor in Department of Computer Engineering, at Hanzhong University, Donghae, Republic of Korea. From 2004 to 2006, he was chief at computer center of Hanzhong University. His main research interests include parallel/distributed processing, I/O disk system, and computer game engineering.