http://ansinet.com/itj

ITJ

ISSN 1812-5638

INFORMATION TECHNOLOGY JOURNAL

ANSInet
Asian Network for Scientific Information
308 Lasani Town, Sargodha Road, Faisalabad - Pakistan

Information Technology Journal 11 (3): 364-367, 2012
ISSN 1812-5638 / DOI: 10.3923/itj.2012.364.367

Cost-benefit Analysis of the Web Hierarchy Caching Model

[1]Lei Shi, [1]Peijuan Yao, [2]Lin Wei and [1]Yongcai Tao
[1]School of Information Engineering, Zhengzhou University, Zhengzhou 450001, China
[2]School of Software, Zhengzhou University, Zhengzhou 450002, China

Abstract: Web proxy caching hierarchy is one of the main solutions used to improve Internet QoS. Focusing on the modeling of Web proxy hierarchy caching, this paper applies cost function to analyze Web caching performance. The experiments evaluate the caching performance of different combinations of replacement policy (LRU, LFU, GDS) across different levels within Web hierarchy caching model. The experimental results demonstrate that when the lower level uses the LFU or LRU replacement policy and the upper level uses the GDS, the two-level hierarchical proxy model achieves higher caching performance.

Key words: Web hierarchical caching model, cost function, hit rate, byte hit rate

INTRODUCTION

Web proxy caching has been proved to be one of the most prevalent approaches to improving user-perceived performance and reducing network traffic (ElAarag and Romano, 2009). In order to avoid single point of fault and improve caching performance, several caches being put at different levels, can cooperate. If the request is not satisfied by the lower level cache, then it will be forwarded to the upper level cache only if the copy of object is found.

In recent years, the research on Web hierarchical caching model has received increasing attention. Some of the most commonly known Web cache placement policies are LFU, LRU, GDS. The representative cache protocols are ICP, HICP, CARP, WCCP and so on (Gupta and Tokekar, 2009). The new algorithm Filter was proposed, nevertheless, Filter incurred additional complexity (Che *et al*., 2001). The researchers mainly pay close attention to improving the existing replacement policies and caching protocols, very little research has been made on the Web cache performance for the different combinations of replacement policy. This paper keeps an eye on cost function which evaluates Web caching performance via using different combinations of replacement policies in Web hierarchical caching model.

COST FUNCTION ANALYSIS OF THE WEB HIERARCHICAL CACHING MODEL

In Web caching hierarchical model, if the requested object has been already in cache, the cache is able to satisfy request, resulting in a cache hit. Otherwise, the cache miss occurs. If the request is not found at any cache level, it will be send to the Web server (Chan *et al*., 2009). Mostly Web evaluation performance metrics are captured by HR and BHR (Busari and Williamson, 2002; Shi and Zhang, 2008). Besides, the symbols of Eq. 1 and 2 are described in Table 1.

Table 1: Symbol definitions

Notation	Description
S_t	The size of object t
Q_t	$Q_t = 1$, the cache hit
	$Q_t = 0$, the cache miss
N	The total number of requests originated at the clients
B	The total bytes of N requests
HR_j	HR at j-th- level ($1 \le j \le m$)
BHR_j	BHR at j-th- level ($1 \le j \le m$)

HR (Hit Rate) is the number of requests satisfied by cache divided by the N cacheable requests seen by the cache.

$$HR = \frac{\sum_{t=1}^{N} Q_t}{N} \times 100\% \quad (1)$$

BHR (Byte Hit Rate) emphasizes the total bytes saved by caching certain objects.

$$BHR = \frac{\sum_{t=1}^{N} Q_t \times S_t}{\sum_{t=1}^{N} S_t} \times 100\% \quad (2)$$

If the request cannot be retrieved from lower level cache, it will be directly sent to the upper level cache. In this case, it will cause data transfer or delay. Here, assume that each cache has the same caching capacity. C_1 indicates the cost which is generated by the request being sent to the first-level cache. In addition to C_1, define C_j

Corresponding Author: Lei Shi, School of Information Engineering, Zhengzhou University, Zhengzhou 450001, China

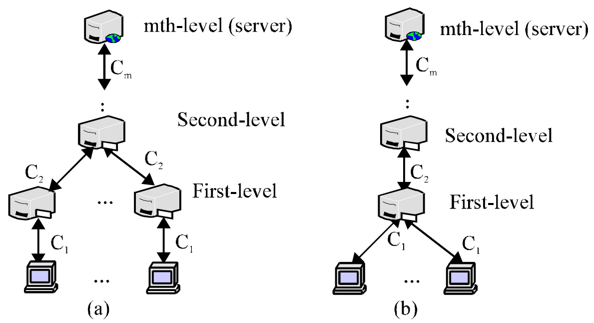

Fig. 1: Cost analysis of m-level hierarchical caching model

$(2 \le j \le m)$ as the cost associated with two adjacent levels. The requests are sent to the fist-level cache with the same probability, so the first-level cache can provide similar HR or BHR. In Fig. 1a, the total number of requests N_1 satisfied at the first-level cache is:

$$N_1 = \sum_{i=1}^{n}(HR_1 \times \frac{N}{n}) = HR_1 \times N$$

Similarly, in Fig. 1b, the total number of requests satisfied at the first-level cache is: $N_1 = HR_1 \times N$, we can conclude $N_1 = N_1$. It indicates that the Fig. 1b can take place of Fig. 1a for the cost analysis in Web hierarchical caching model.

N_j represents the total number of requests satisfied at the jth-level cache and RC_j signifies the cost of satisfied N_j requests. They can be respectively defined as:

$$N_j = HR_j \times (N - \sum_{i=1}^{j-1} N_i) \quad (3)$$

$$RC_j = (\sum_{i=1}^{j} C_i) \times N_j \quad (4)$$

B_j represents the total requested bytes at the jth-level cache. BC_j means the cost of satisfied B_j bytes. Both of them can be calculated as Eq. 5 and 6.

$$B_j = BHR_j \times (B - \sum_{i=1}^{j-1} B_i) \quad (5)$$

$$BC_j = (\sum_{i=1}^{j} C_i) \times B_j \quad (6)$$

The total cost of requests and bytes served from the caches can be expressed as RC and BC.

$$RC = RC_1 + RC_2 + ... + RC_m = \sum_{j=1}^{m}\{(\sum_{i=1}^{j} C_i) \times N_j\} \quad (7)$$

$$BC = BC_1 + BC_2 + ... + BC_m = \sum_{j=1}^{m}\{(\sum_{i=1}^{j} C_i) \times B_j\} \quad (8)$$

Supposing all the requests hit at the server, the total cost of requests and bytes can be expressed as RC_{worst} and Bc_{worst}.

$$RC_{worst} = RC_1 + RC_2 + ... + RC_m = (\sum_{i=1}^{m} C_i) \times N \quad (9)$$

$$BC_{worst} = BC_1 + BC_2 + ... + BC_m = (\sum_{i=1}^{m} C_i) \times B \quad (10)$$

In general, the requests hit at some cache or server. On this occasion, the cost of total requests and bytes can be expressed as RRC and RBC.

$$RRC = \frac{RC}{RC_{worst}} = \frac{\sum_{j=1}^{m}\{(\sum_{i=1}^{j} C_i) \times N_j\}}{(\sum_{i=1}^{m} C_i) \times N} \quad (11)$$

$$RBC = \frac{BC}{BC_{worst}} = \frac{\sum_{j=1}^{m}\{(\sum_{i=1}^{j} C_i) \times B_j\}}{(\sum_{i=1}^{m} C_i) \times B} \quad (12)$$

In order to facilitate the cost analysis, assuming $C_j = C$ $(1 \le j \le m)$. According to Eq. 11 and 12, we can deduce Eq. 13 and 14.

$$RRC = \frac{N_1 + 2N_2 + ... + mN_m}{mN} \quad (13)$$

$$RBC = \frac{B_1 + 2B_2 + ... + mB_m}{mB} \quad (14)$$

Analyzing the following two cases:

- If all the requests hit at the first-level cache, we can conclude RRC = RBC = 1/m. In two-level Web hierarchy caching model, we can conclude RRC = RBC = 1/3
- If all the requests are served at the server, then, RRC = RBC = 1. We can obtain RRC = RBC = 1 for two-level Web hierarchy caching model

In Web hierarchy caching model, the smaller the RRC or RBC is, the less the cost of requests is.

EXPERIMENT AND ANALYSIS

Web hierarchical caching model is an effective way to reduce the user perceived latency as well as resource consumption for network applications. For simplicity, this

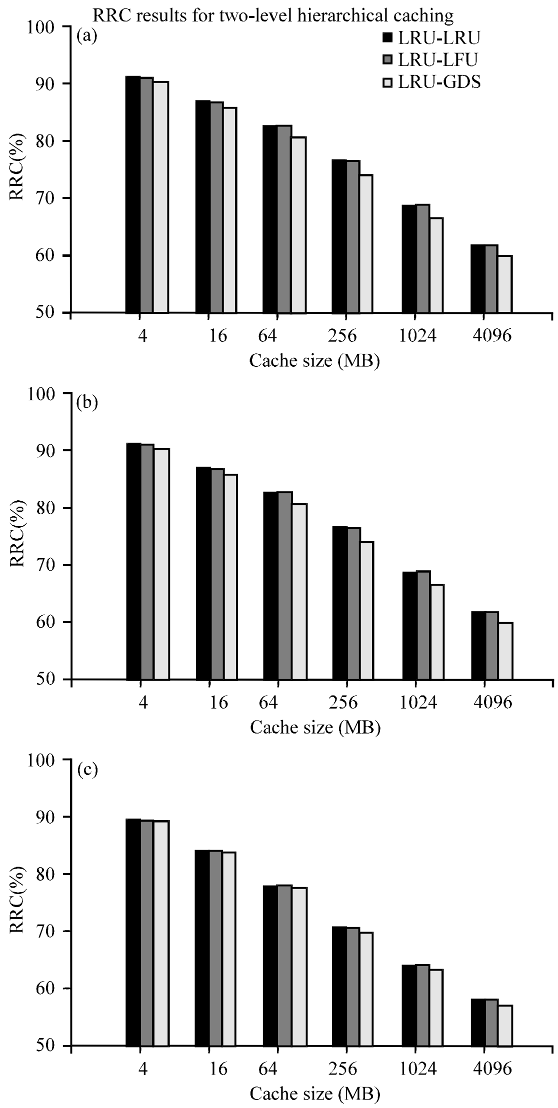

Fig. 2(a-c): The results for RRC

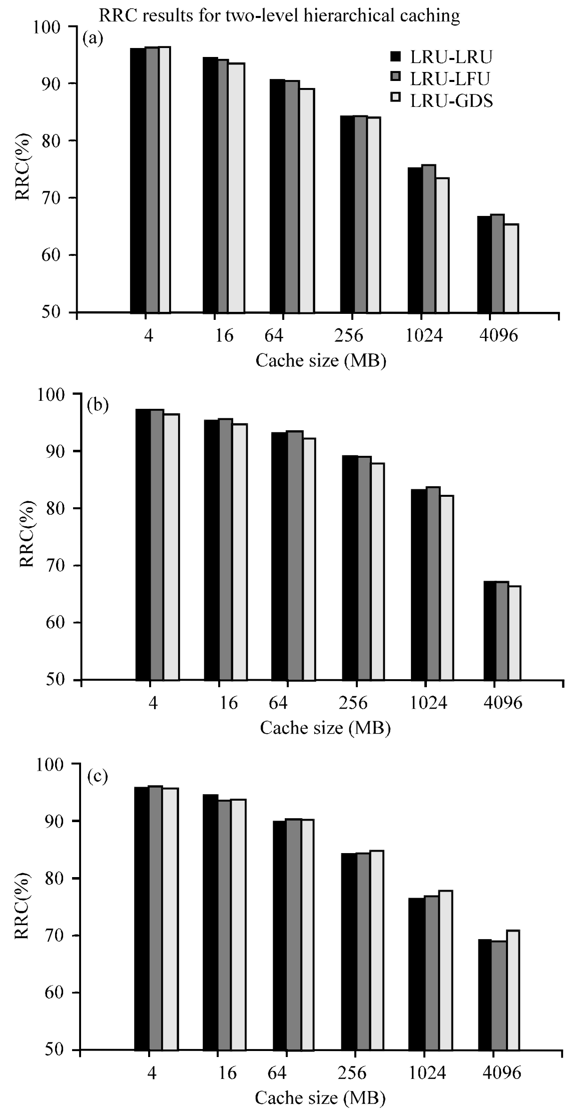

Fig. 3(a-c): The results for RBC

paper focuses on the two-level caching hierarchy and evaluates the caching performance by using the synthetic workloads (Busari and Williamson, 2002). Only consider three different policies which are LRU, LFU and GDS. Suppose lower level cache can not send a request to the upper level cache and the objects are static. There is no exchange information between each cache. LFU-LFU indicates that the former LFU represents lower level cache using LFU, the latter LFU indicates the upper level cache using LFU.

In Fig. 2a, given the same cache size, the combination LRU-GDS can achieve higher performance in terms of RRC, compared with the other two combinations (LRU-LRU and LRU-LFU). Clearly, Fig. 2b shows the combination LFU-GDS has lower RRC than the other two combinations (LFU-LFU and LFU-LRU). It is visible that the combination GDS-GDS can outperform both GDS-LFU and GDS-LRU, as shown in Fig. 2c.

As shown in Fig. 3a, given the same cache size, the combination LRU-GDS can achieve higher performance in terms of RBC, compared with the other two combinations (LRU-LRU and LRU-LFU). It can be drawn from Fig. 3b that the combination LFU-GDS has lower RRC than the other two combinations (LFU-LFU and LFU-LRU). In Fig. 3c, it is obvious that the combination GDS-GDS is the worst among the other two combinations GDS-LFU and GDS-LRU.

CONCLUSIONS

This paper applies cost function to investigate Web cache performance of combinations of replacement policy

(LRU, LFU, GDS) at different levels of the two-level Web hierarchy model. The results manifest that when the lower level cache uses the LFU or LRU and the upper level cache uses the GDS, the two-level hierarchical cache model obtains higher Web caching performance.

REFERENCES

Busari, M. and C. Williamson, 2002. ProWGen: A synthetic workload generation tool for simulation evaluation of web proxy caches. Comput. Networks, 38: 779-794.

Chan, E., W. Li and D. Chen, 2009. Energy saving strategies for cooperative cache replacement in mobile ad hoc networks. Pervasive Mobile Comput., 5: 77-92.

Che, H., Z. Wang and Y. Tung, 2001. Analysis and design of hierarchical web caching systems. Proceedings of 20th Annual Joint Conference of the IEEE Computer and Communications Societies, April 22-26, 2001, Anchorage AK, USA., pp: 1416-1424.

ElAarag, H. and S. Romano, 2009. Comparison of function based Web proxy cache replacement strategies. Proceedings of the 12th International Conference on Symposium on Performance Evaluate on of Computer and Telecommunication Systems, July 13-16, 2009, Istanbul, pp: 252-259.

Gupta, R. and S. Tokekar, 2009. Pair of replacement algorithms MFMR and AF-LRU on L1 and L2 cache for proxy server. Proceedings of the Annual IEEE India Conference, December 18-20, 2009, India, Gujarat, pp: 1-4.

Shi, L. and Y. Zhang, 2008. Optimal model of web caching. Proceedings of the 4th International Conference on Natural Computation, October 18-20, 2008, Zhengzhou, Jinan, pp :362-366.