Volume 3, Issue 10, October 2013 ISSN: 2277 128X

International Journal of Advanced Research in Computer Science and Software Engineering

Research Paper

Available online at: www.ijarcsse.com

Optimized Cache Performance for Peer to Peer Network

Sumathi Sivaraj
M.E Computer Science and Engineering
Sri Eshwar College of Engineering
Anna University, India

S. Yasotha
M.E., (Ph.D.) Assistant professor,
Sri Eshwar College of Engineering,
India

Abstract— Peer-to-peer systems generate a major fraction of the current Internet traffic, and significantly increase the load on ISP networks and the cost of running and connecting customer networks to the Internet. It mitigates negative impacts, several previous works within the literature have proposed caching of P2P traffic, but very few have considered designing a caching system to actually do it. Caching P2P traffic is more complex than caching other internet traffic and needs several new algorithms and storage systems. The paper presents the design and evaluation of a entire, running, proxy cache for P2P traffic, referred to as pCache. pCache transparently intercepts and serves traffic from different P2P systems. A replacement storage system is projected and enforced in pCache. Storage system is optimized for storing P2P traffic, and shown to outperform other storage systems. A new algorithm to infer the information required to store and serve P2P traffic by the cache is proposed. Extensive experiments to evaluate all aspects of pCache using actual implementation and real P2P traffic are presented. Based on the analysis, proposed a new algorithm, called Least Grade Replacement (LGR), takes recency, frequency, perfect-history, and document size under consideration for Web cache optimization.

Keywords— Peer-to-peer systems; Cache; Least Grade Replacement; Web Cache Optimization; proxy; ISP Networks; Web Cache; LRV; LUV;

I. INTRODUCTION

A computer network, often simply referred to as a network, is a collection of hardware parts and computers interconnected by communication channels allow sharing of resources and information. One process in one device is able to send/receive data to at least one process residing in a remote device, the two devices are said to be in a network. A network is a group of two or more computer system linked together. Types of computer network are:

Local-Area Networks (LANs) :
The computers are father apart and close together.

Wide-Area Networks (WANs) :
The computers are farther apart and are close together by telephone lines and radio waves.

Campus-Area Networks (CANs):
The computers are within a restricted geographic region, includes campus and military base.

Metropolitan-Area Networks (MANs):
A data network designed for a town and city.

Home-Area Networks (HANs):
A network contained within a users home connects a person's digital devices. In addition to this type, the subsequent characteristics are also used to classify different types of networks are:

Topology :
The geometric arrangement of a computer system. Common topologies have a bus, star, and ring.

Protocol :
The protocol defines a common set of rules and signals computers on the network use to communicate. One of the most familiar protocols for LANs is called Ethernet.

Architecture :
Networks can be generally classified as using either a peer-to-peer and client server architecture.

Computers on a network are sometimes called nodes. Computers and devices allocate resources for networks are called servers. A general phrase for software is designed to help set up, manage, and monitor computer networking

software applications are available to manage and monitor networks of all sizes, from the tiniest home net works to the large enterprise networks. A class of firewalls design to filter network and Internet traffic based upon the applications and traffic types using specific ports. The application-specific harsh security policies provided by Next Generation Firewalls help them detect application-specific attacks, giving them the potential to catch additional malicious activity than more traditional firewalls.

In many ways a Next Generation Firewall combines the capabilities of first-generation network firewalls and network intrusion prevention systems (IPS), while also offering additional features like SSL and SSH review, reputation-based malware filtering and Active Directory integration support. Networks may be classified according to a wide variety of characteristics the medium used to transport the info, link protocol used, scale, topology, and structure scope.

Figure1. Model network between two systems

Communications protocols define the rules and data formats for exchanging information in a computer network, and supply the basis for network programming. Figure 1 refers the common model network between two systems. Well-known communications protocols are Ethernet, a hardware and Link Layer standard is ubiquitous in local area networks, and the Internet Protocol Suite, defines a set of protocols for internetworking, for data communication between multiple networks, additionally as host-to-host information transfer, and application-specific data transmission formats. Computer networking is usually thought of a sub discipline of electrical technology, telecommunications, computer technology and information technology, since it depends upon the theoretical and practical application of these disciplines.

II. PEER TO PEER NETWORK

Based on their layout, two types of networks are available. A network is referred to as Peer-to-Peer if most computers are similar and run workstation operating systems. P2P is a distributed application architecture partitions tasks and workloads among peers. Peers area unit equally privileged, strong participants within the application. Each pc within the network is referred to a node. The owner of every pc on a P2P network would set aside a portion of its resources, processing power, disk storage and network information measure, directly available to different network participants without the need for central coordination by servers and stable hosts. Peers are both suppliers and customers of resources, in distinction to the standard client–server model only servers supply (send), and clients consume (receive). Figure 2. Shows the simple peer to peer network examples.

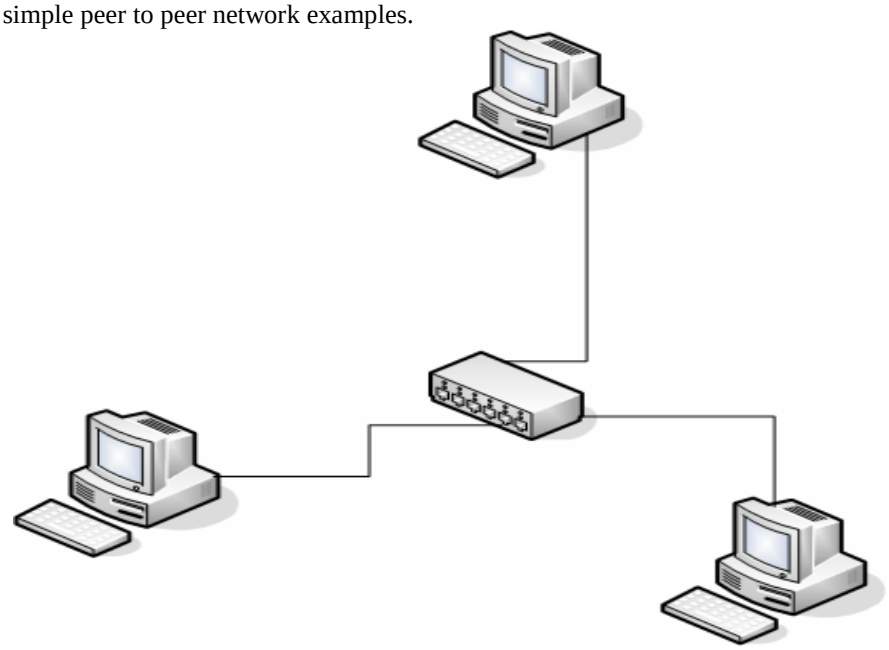

Figure 2. Peer to peer networks

Peer-to-peer networks are more commonly implemented less ten computers are involved and strict security is not necessary. All computers have the equal status, hence the term peer, and communicate with each other on an equal footing. Files, word processing and spreadsheet, can be transferred across the network and all the computers on the network can share devices, printers or scanners, are connected to any one computer.

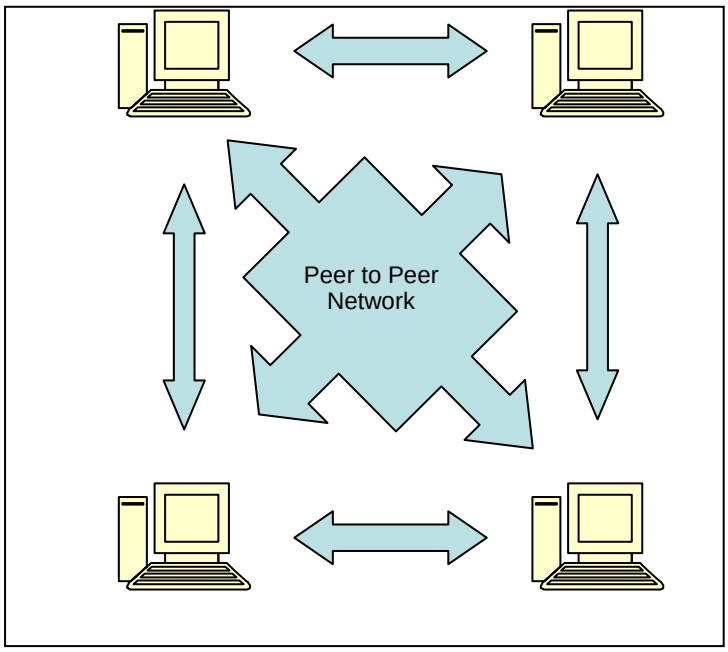

Figure 3. Peer to Peer Networking

III. RELATED WORK

A scheme of peer to peer network is explained. It means will show all research before our project starting. The research should be about example of peer to peer network, layout of the hardware system, layout of the software interface.

In December 1997 cost aware www proxy caching algorithms presented by peicao [1].The paper introduces GreedyDualSize, incorporates locality with cost and size concerns in a simple and non-parameterized fashion for high performance. Trace-driven simulations display that with the appropriate cost definition, GreedyDualSize outperforms existing web cache replacement algorithms in many aspects, together with hit ratios, latency reduction and network cost reduction. In addition, GreedyDualSize can potentially improve the performance of main-memory caching of Web documents. In November 1998 wild- area Internet Traffic Patterns and characteristics by Kevin Thompson[2]. The paper presents observations on the patterns and characteristics of wide-area Internet traffic, as maintained by MCI's OC-3 traffic monitors. It report on measurements from two OC-3 trunks in MCI's commercial Internet backbone over two time ranges (24-hour and 7-day) in the presence of up to 240,000 flows. It reveals the characteristics of the traffic in terms of packet sizes, flow length, volume, and proportion composition by protocol and application, moreover as patterns seen over the II time scales.

In May 1999 web caching and zipf-like Distributions: evidence and implications Presented by Breslau [9]. The paper addresses two unresolved issues about web caching. The issue is about web requests from user community are distributed according to Zipf's law. Many early studies have supported this claim while other recent studies have suggested otherwise. The second issue relates to variety of recent studies on the characteristics of web proxy traces, shown that the hit-ratios and temporal locality of the traces exhibit certain asymptotic properties are uniform across the different sets of the traces. In June 1999 Reducing the disk I/O of web proxy server caches presented by chrlos maltzahn[4]. The paper evaluates ways to reduce the amount of required disk I/O. First compare the file system interactions of two existing web proxy servers, CERN and SQUID. Then show how design adjustments to the current SQUID cache architecture can dramatically reduce disk I/O. Our findings recommended two that strategies can significantly reduce disk I/O: (1) preserve locality of the HTTP reference stream while translating these references into cache references, and use virtual memory instead of the file system for objects smaller than the system page size. The assess techniques reduced disk I/O by 50% to 70%. In October 1999 using full reference history for a efficient document replacement in web caches presented by Bahn[6]. The paper Presents new such algorithm that is efficient robust called Least Unified Value (LUV) evaluates a web document based on its cost normalized by the likelihood of it being re referenced. The results in a normalized assessment of the contribution to the value of a document leading to a fair replacement policy.

In October 1999 exploiting result Equivalence in caching dynamic web content Presented by ben smith[10]. The paper presents the classification of locality in dynamic web content into three kinds identical requests, Equivalent Requests, Partially equivalent Requests it presents new protocol Dynamic Content Caching Protocol to allow individual content generation. In April 2000 replacement policies for a proxy cache presented by luigi rizzo[5]. The paper analyzes access traces to an online proxy, looking at mathematical parameters to be used in the design of a replacement policy for

documents held in the cache. In the first part of this paper presented a number of properties of the lifetime and statistics of access to documents, derived from two big trace sets coming back from very different proxies and spanning over time intervals of up to five months.

In june 2003 Replace Problem in web cache presented by seda cakiroglu[3]. The Paper Presents focuses on the replacement problem in Web caching, it arises due to limited storage. It seeks the best configuration for a fully-connected network of N caches. The problems formulated as a discrete optimization problem. A number of low-complexity heuristics are studied to obtain approximate solutions. Performances are tested under fictitious probabilistic request sequences and access logs of real Web traffic. Longest-Forward-Distance (LFD) the classical optimal off-line paging algorithm is observed not to be optimal. In March 2004 A Goal Oriented self tuning caching algorithm presented by Ganesh[8]. The paper presents the performance matching and exceeding the best performance of the known greedy dual-size algorithms for either object or byte hit ratios across different web workloads. GD-GhOST consistently outperforms the other algorithms tested, at its worst observed performance GD-GhOST exhibited equivalent miss rates to those of the best applicable Greedy-Dual variant, while achieving miss rates that were 25% lower than the worst performing variant. For byte miss rates, GD-GhOST consistently demonstrated rates lower than the best applicable Greedy-Dual variant.

The initial use of P2P networks in business followed the deployment in the early 1980s of free-standing PCs. In contrast to the mini mainframes of the day, the VS system from Wang Laboratories Inc., served up word processing and other applications to dumb terminals from a central computer and stored files on a central hard drive, the PCs had self-contained hard drives and built-in CPUs. The sensible boxes conjointly had aboard applications, meant could be deployed to desktops and be useful without an umbilical cord linking them to a mainframe. Diagram shows a P2P network operation. The solid lines represent physical, hard-wired network cables. The dotted lines indicate each PC can communicate and share files with every other PC on such a network. Figure 4. Shows peer to peer network between various systems using network devices. A printer attached to one PC can be used by other PCs on the network—if printers PC allow such use.

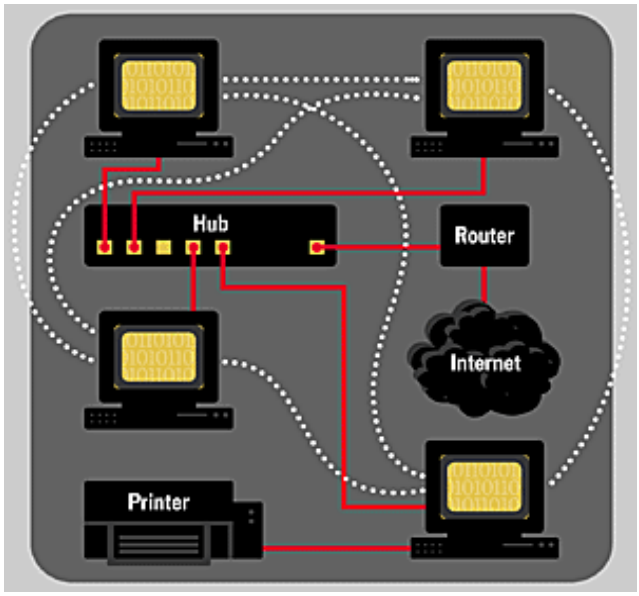

Figure 4. Peer-to-Peer network

IV. PROBLEM DEFINITION

The Internet is rapidly growing in number of users, traffic levels, and topological quality. The same time it is increasingly driven by economic competition. These developments provide the characterization of network usage and workloads more difficult, and yet more critical. The existing system has lot of problems for p2p web cache. Current P2P web caching schemes can solve the problems of proxy partly drawbacks respectively. The building block of substrate of existed P2P web caching systems is the notion of peer-group, in a number of nodes are organized by Distributed Hash Table algorithms or physical locations. DHT organization is not appropriate for the P2P web caching application, since it assigns file storage based on a random hash function, not based on files users already possess, in P2P web caching network, users share cache have, and it would be impractical to imagine users storing files other than their own. In the other organization based on physical locations, to find a specific object, peer must search the whole P2P space, is time-consuming. More, the above methods cannot make use of semantic information among the sharing contents of peers. The proposed system solved all the above problems and increases the performance of p2p web cache.

V. PROPOSED WORK

Caching is a technique first used by memory management to reduce bus traffic and latency of data access. Net traffic has increased tremendously since the starting of the 1990s. With the numerous increase of net traffic, caching techniques

are applied to web caching to decrease internetwork traffic, user-perceived latency, and server load by caching the documents in local proxies. The analyzed both advantages and disadvantages of some current Web cache replacement algorithms including lowest relative value rule, least weighted usage algorithm and Least Unified-Value (LUV) algorithm. Based on our analysis, proposed a new algorithm, called Least Grade Replacement (LGR), takes recency, frequency, perfect history, and document size under consideration for net cache optimization. The optimum recency coefficients were resolute by using 2- and 4- way set associative caches. The cache size was different from 32k to 256k in the simulation. The simulation results showed the new algorithm (LGR) is better than LRU and LFU in terms of hit ratio (BR) and Byte Hit Ratio (BHR).

A. Proxy Server Implementation

A Proxy server, runs with mentioned features, inherently helps speeder browsing of web pages with use of least grade page replacement algorithms. Server is successfully implemented with a few numbers of clients but it could be implemented for more of them. It is more reliable, more advantageous than the existing one uses the old Data structures concept. It will add a bigger network and also maintains load balancing system application is executable under any platform and with any number of clients too.

B. Web Crawler

A web crawler is a program, given one or more seed URLs, downloads the web pages associated with these URLs, extracts any links having them, and recursively continues to transfer the web pages identified by these links. Web crawlers are an important component of web search engines, used to collect the corpus of web pages indexed by the search engine. used in many other applications process large numbers of web pages, as web data mining, comparison shopping engines. Despite their conceptual simplicity, implementing high-performance web crawlers poses major engineering challenges due to the scale of the net. In order to crawl a considerable fraction of the surface web in a reasonable amount of time, web crawlers must download thousands of pages per second, and are typically distributed over tens or hundreds of computers. Their two main data structures – the frontier set of yet-to-be-crawled URLs and the set of discovered URLs – typically do not fit into main memory, efficient disk-based representations need to be used. Finally, the need to be polite to content providers and not to overload any particular web server, and a desire to prioritize the crawl towards high-quality pages and to maintain corpus freshness impose additional engineering challenges. Figure 4.1 shows the format and components of the web crawler.

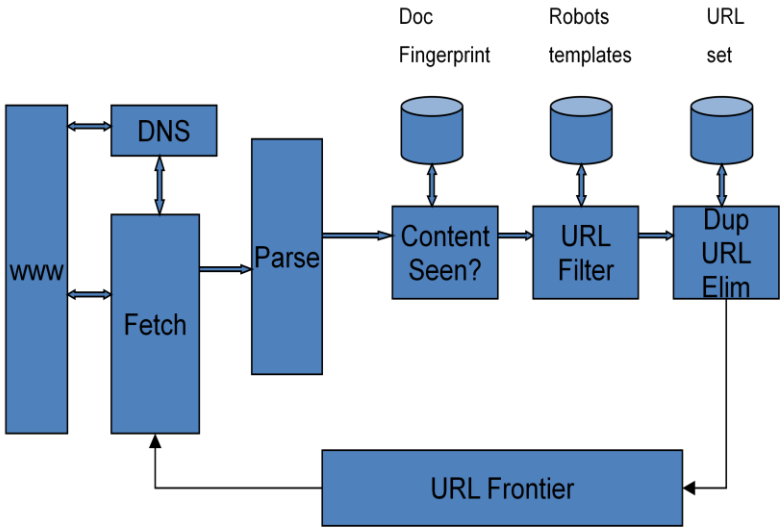

Figure 5. Web crawler

C. Web Cache Replacement Polices

According to the above conclusions, proposed a new replacement algorithm. The algorithm grades each in-cache document basing on its passed history, are recency, frequency, perfect-history and size. The set is full, the least grade document will be replaced, and then its grade will be stored in a Perfect History Grade Depository (PHGD) for future references. Due to the survey, can draw a conclusion the relatively most important factor is its recency, and frequency, perfect-history and size. Consider these four factors for grading because are relatively most important factors for real network traffic. The n-way set associative caches is employed. The least grade page replacement algorithm is given below.

ALGORITHM :: ReplaceLeastGradePage

/ In-cache document based on its passed history */*
ICDR: In-Cache Documents Records
/ Discarded document in cache */*
PDDG: Previously Discarded Documents Grade
/ The de facto hit ratio beginning with 0 */*
F : Frequency of ICDR ;
R : Recency Set of ICDR; */* R=< r1;r2;…;rn> */*


```
L : Length of document;
BG : Bonus Grade of document)
{ IF (document k in cache)
FOREACH doc in ICDR DO
WHILE(doc.F in F
& doc.R in R
& Size(doc) in L)
doc ← newValue;
PDDG ←weightα , α ∈(0, 1);
ELSE
FetchDocFromOrigina();
DiscardInCachedDoc();     }
/*Fetch document k from original site*/
PROCEDURE: FetchDocFromOrigina()
{           IF(L is NOT NULL)
INSERT k into Cache;
UPDATE each ICDR;
k.BG = 0;
ELSE
FOREACH g in grade DO
g =R 1δ + × F 2 δ + ×C 3 δ + × BG 4 δ ;       }
/*discard in-cached doc with the least grade*/
PROCEDURE: DiscardInCachedDoc()
{           PDDG ← PHGD.grade;
INSERT k into Cache;
IF ( PDDG of k in PHGD)
k.BG←PDDG;
Delete its PDDG in PHGD;
ELSE
k.BG=0;
Update each ICDR and PDDG;       }
```

VI. SYSTEM IMPLEMENTATION

A principal activity of the development phase is coding and testing until the user requirement specification fulfilled by the each modules and component of the overall system. Other important activities include implementation planning, equipment acquisition and system testing. The development phase concludes with a development phase report and user review. A software application in generally implemented after navigating the complete life cycle method of a project. Various life cycle processes like requirement analysis, design module phase, verification, testing and at last followed by the implementation phase results in a successful project management. The software application is basically a web based application has been successfully implemented after passing various life cycle processes mentioned above.

The software is to be implemented in a high standard industrial sector, various factors includes application environment, user management, security, reliability and finally performance are taken as key factors throughout the design part. These factors area unit analyzed step by step and the positive as well as negative outcomes are noted down before the final implementation. The applications validations area unit created, taken into account of the entry levels accessible in various modules. Possible restrictions like number data format, date formatting and confirmations for both save and update options ensures the correct data to be fed into the database. Therefore all the aspects are charted out and the complete project study is practically implemented successfully for the end users.

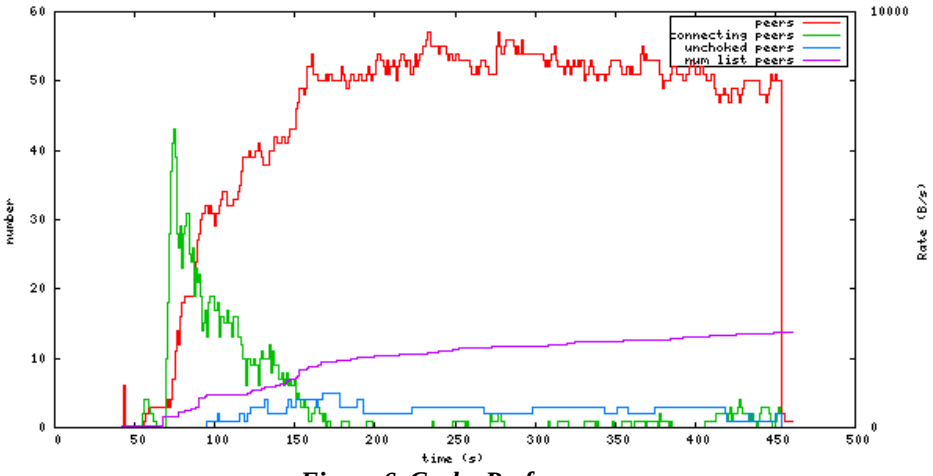

Figure 6. Cache Performance

VII. FUTURE SCOPE

In future, plan to fully implement our scheme, and further optimize the similarity measurement algorithms. Designing a cooperative P2P caching protocol, proxy caches in different cooperate to serve requests for each other in the hope of minimizing WAN traffic. Designing efficient cache zoning techniques different workloads occupy different zones in the cache with different cache spaces. The same cache could also be tuned to minimize the outgoing traffic by allocating a larger zone for large video objects. In all cases, different algorithms might work better for different zones and could be run in the same cache concurrently. Cache zones might also cooperate by evicting the least valuable object from any of them to optimize a common desired metric.

VIII. CONCLUSION

Based on perfect-history LFU and LRU, proposed a new algorithm (LGR) by considering recency, frequency, perfect-history and size in replacing policy. Experimental results show the proposed algorithm can reduce network traffic and latency of data access efficiently. The potential gain of cooperative caches areanalyzed for P2P traffic. Proposed two models for cooperation are: 1) among caches deployed in different AS and 2) among caches deployed within an oversized AS. In each models, caches collaborate to save bandwidth on expensive WAN links. The traces describe object requests would have been seen by many caches if were deployed in ASs operating in different geographical regions and have different number of clients. Many trace-based simulation experiments are designed to rigorously analyze various aspects of cooperative caching. Results show cooperative caching is viable for P2P traffic because it could improve the byte hit rate by up to 330 percent in some cases. Considering the large volume of the P2P traffic, even one percent improvement in computer memory byte hit rate accounts to saving in the order of terabytes of traffic on the expensive WAN links. In addition, proposed simple models for object replacement policies in cooperative caching systems.

REFERENCE

[1] Peicao, Snady Irani(1997), 'Cost-Aware WWW Proxy Caching Algorithms' in Proceedings of the USENIX Symposium on Internet Technologies and Systems.
[2] Thompson. K, Miller. G and Wilder. R(1998), 'Wilde-Area Internet Traffic Patterns and Characteristics', in Proceedings of third International Conference Web caching.
[3] Seda Cakiroglu, Erdal Arikan (2003), 'Replace Problem in Web Caching', in Proceedings of IEEE Symposium on Computers and Communications.
[4] Chrlos Maltzahn, Kathy, Richardson. J, Dirk Grunwald (1999), 'Reducing the Disk I/O of Web Proxy Server Caches', in Proceeding of the 1999 USENIX Annual Technical Conference, Monterey, California.
[5] Luigi Rizzo and Lorenzo Vicisano (2000),'Replacement Policies for a Proxy Cache', IEEE/ACM Trans Networking. Apr. 2000, pp: 158-170.
[6] Bahn. H, Noh. S and Koh. K (1999), 'Using Full Reference History for Efficient Document Replacement in net Caches', in Proceedings of the 2nd USENIX Symposium on Internet Technologies & Systems.
[7] Ying Shi, Edward Watson, and Ye-sho Chen (1997), 'Model-Driven Simulation of World-Wide-Web Cache Policies', in Proceeding of the 1997 Winter Simulation Conference, June,1997, pp: 1045-1052.
[8] Ganesh, Santhanakrishnan, Ahmed, Amer, Panos. K, Chrysanthis and Dan Li (2004), 'GDGhOST: A Goal Oriented Self Tuning Caching Algorithm', in continuing of The 19th ACM Symposium on Applied Computing.
[9] Breslau. L, Cao. L, Phillips. G and Shenker. S(1999), 'Web Caching and Zipf-like Distributions: Evidence and Implications', In continuing of Infocom 99.
[10] Ben Smith, Anurag Acharya, Tao Yang, and Huican Zhu (1999), 'Exploring Result Equivalence in Caching Dynamic Web Content', second USENIX Symposium on Internet Technologies and Systems.

BIOGRAPHY

S. SUMATHI received her MCA Degree from Bharathiyar University Coimbatore, Tamilnadu, India and pursuing M.E (Computer Science and Engineering) degree from Sri Eshwar College of Engineering Coimbatore, India. Her area of interest is Network and Cloud Computing.

S.YASOTHA obtained her B.E (Computer Hardware and Software engineering) from Avinashilingam University for Women, Coimbatore (Tamilnadu, India) and received her M.E (Faculty of Information and Communication Engineering in VLSI Design) from Anna University of Technology, Coimbatore. She is pursuing Ph.D (Information and Communication Engineering in ECE Department). She is currently serving as Assistant professor of Computer Science and Engineering at Sri Eshwar college of Engineering, Coimbatore (Tamilnadu, India). Her area is Network Security and Wireless Sensor Network.

