Traffic Balance using SNMP for Multimedia Service (TBSMS) Architecture

Seok Kuen Lim, Hyun Pyo Lee, Jae Yong Lee, Kyun Ha Lee
Department of Computer and Science Engineering, Inha University
{sklim | hplee | jylee}@blue.cse.inha.ac.kr, khlee@dragon.inha.ac.kr

Abstract
Currently, lots of research is going on in the field of the load distribution within HTTP. RR-DNS and SWEB are the most representative load distribution research. But, there are still many problems: unbalancing of load, load increase of web server and cost increase. Also, clients that require lots of data like multimedia happens to increase network load. To solve these, research about client/server architecture is going on. But, the clients must know the server’s address and there are complexity and migration problems for design of such as protocol.

This paper proposes TBSMS that is capable of choosing the optimal server considering the service capacity of the server as well as the network load. This paper demonstrates that TBSMS uses the web to solve the problem that client must know the server’s address and uses SNMP to solve the complexity and migration problem.

1. Introduction
Recently, Internet is growing up rapidly. Demand of multimedia data is increasing also. As these reasons, on the Internet load distribution problem is coming to the fore with the increase in volume and quantity of data.

But, a lot of difference are being come to the fore because of the server’s performance and the partial load.

To solve these, there are a lot of network load distribution research on the Internet, for example RR-DNS (Round Robin DNS), SWEB (Scalable Webserver) and An Agent Protocol for Multimedia over IP[4][6]. But, there are a lot of problem including load balancing and web server’s load. Also, existing multimedia service on the Internet has a lot of restriction different from the data based on character. Multimedia service requires the wide bandwidth because it is very large volume. And both the client and server side requires a lot of resource and high performance capacity. The high-speed network and the advanced computer H/W and S/W are solving these requirements [1][2].

But, the researches about this multimedia service considering both server status and network parameter is feeble still [2][3]. This paper proposes and implements TBSMS architecture that is implemented and migrated easily without the design of a new protocol on the Internet and will distribute the load concentrated on one site with low cost.

2. Existing Researches

2.1 RR-DNS
RR-DNS architecture include several DNS that have same contents and each unique IP address. Also, RR-DNS registerate a domain name that represent all DNS. It is operated by the Round Robin method. In this method, a representative response each IP address according to custom's request.

But, RR-DNS has load unbalance problem and don't have capacity that can cope with when system down.

2.2 SWEB
SWEB architecture include application scheduling method and clustered servers base on RR-DNS. Application scheduling method is method supplementing load unbalance with load distribution agent. And, clustering method can be operated in other system when any system down.

But, SWEB method go down system's performance because systems must have load information of other systems.

3. Architecture of TBSMS

394
3.1 TBSMS Architecture

TBSMS is consisted of clients requesting services, a Central Management System (CMS) that gathers information about the current status of multimedia servers through SNMP, and multimedia server that offer media services to clients.

Upon the request of a client, the CMS with the current status information of the multimedia server informs the client.

3.2 Operation of TBSMS

Clients connect to the site providing the multimedia service and request the service through HTTP. For the multimedia service requested, in order to find the optimal media server that can provide multimedia service, Optimal Server Selection (OSS) algorithm is performed.

The chosen IP address of the media server is send to client in a metafile format. The protocol, the optimal media servers' IP address and the requested multimedia file name are included in this metafile. The metafile is sent to the client and interpreted by the client and the client requests connection. After the client is connected to the multimedia server the process that the client take is the same as the process taken before the CMS.

That is, client is connected to the multimedia server and the multimedia data is provided on the RTP/RTSP.

3.3 Operation of CMS

CMS requests a certain parameter information per a special time given by the contab file. This requested parameter is information about the current status and load.

CMS updates the special DB file with the management object information that the SNMP agents collect.

The media status information that the CMS requests the SNMP agent is consisted of the load information, volume of available memory, CPU use rate of the specific user and CPU use rate of the system. Also, the media load information that CMS requests the SNMP agent is information about volume of system.

3.4 Information of TBSMS

TBSMS uses the following information to find the most optimal multimedia server.

In Formula. 1 ifInOctets, ifInOctetsx, ifOutOctets, ifOutOctetsx are volume of the Octets that is input to or output from multimedia server till a special time from system boot. SysUpTime, sysUpTimeX are in millisecond from system boot to the special time t [5].

\[
\text{Ethernet Utilization}(%) = \frac{(total\;bits\;sent + total\;bits\;received)}{bandwidth} \times 100
\]

\[
C = \text{ifSpeed} = (\text{ifInOctets} - \text{ifInOctetsx})
\]

\[
B = (\text{ifOutOctets} - \text{ifOutOctetsx})
\]

\[
A = (\text{sysUpTime} - \text{sysUpTimeX})
\]

The SNMP agents send CMS this requested information. CMS finds the optimal multimedia server base on the updated file.

4. Implementation and Experimentation

4.1 Experimentation Environment

Operation System of CMS used Sun OS version 5.5.1 and system used a Sun SPARC computer. Web server and media server installed and experienced on the same subnet. Media server experienced above Pentium II 350MHz, 64MB memory, and network card of 10Mbps. OS of the media server used Red Hat Linux 6.0 and media...
This paper proposed, implemented, and designed TBSMS architecture that service multimedia. As a result of the experiment TBSMS architecture solved the load central problem in the existing client/server architecture. The Web solves the problem that clients must know the agent's IP address in the client/agent/server. Without new protocol design, TBSMS architecture can migrate easily. Additionally, this architecture can be installed easily and is low cost with the ordinary server. Also, this architecture can acquire basic data to determine whether to add a new media server because it collects the request count and the reject count on the Web and also to determine which data to store, how many to store, and where to store it in.

To solve the load-balancing problem, the experiment result is shown in Figure.5. In Figure.5, according to increasing user count, it shows that each multimedia server undertaken is regular.

But, we can not know which media server to operate. To solve this problem, more research need in the Trap-PDU.

6. Reference

5. Conclusion

Load Balance of the TBSMS

In Figure.3, we can see that the traffic rate of each multimedia server is constant increasing user count.