Available Online at www.ijcsmc.com

International Journal of Computer Science and Mobile Computing

A Monthly Journal of Computer Science and Information Technology

ISSN 2320–088X

IJCSMC, Vol. 4, Issue. 4, April 2015, pg.238 – 242

RESEARCH ARTICLE

Server Load Balancing in Content Distribution Networks by Transaction Least Work Left

Thejas S K

Department of Computer Science and Engineering, New Horizon College of Engineering, Bangalore, India

Email: skthejas@gmail.com

Ms. Gayathri G S

Asst. Professor, Department of Computer Science and Engineering, New Horizon College of Engineering, Bangalore, India

Email: gayathrigummaraj@gmail.com

Abstract—In the recent days the current server capacity is being overloaded by many online features like mobile computing and gaming, internet protocol television and so on. A crucial aspect here is the capacity of the server in terms of the user perception. Especially when it comes to the videos it is very important to balance the server load out of the available number of servers. For the networks where the contents are replicated in all the servers, this paper proposes a transaction least work left algorithm which helps to improve the network quality of the content distributed networks, as well as to select the better server which is left with the least number of work and helps the end user with the better experience with the server application.

Keywords—End User Perception; Network quality; work Done; Work Left

I. INTRODUCTION

The factors what the service providers and the network providers are concentrating from many years are delay, bandwidth and the other parameters which are related to quality of service. Even though it served as the good technology in improving the network quality, somewhere it was ambiguous when it comes to the end user perception. Improving the quality of the network, as perceived by the users, is a significant challenge for network operators and providers. One crucial element of this paper is routing the request to the server. The routing or the selection of the better server here is purely depends on the summation of the work left of the servers. The parameters for the routing and selection are total duration, work done, work left, join time and finish time of the video. Ultimately, the goal of this paper is to minimize the end user's regrets, improve the quality of experience of the user and to improve the balancing capacity of the servers where the contents have been replicated among the servers which are in the network.

II. RELATED WORK

Here, the existing techniques regarding the content distributed networks and its limitations will be explained. One of the existing systems used the information of the load of the server which used the reliable server pooling. The limitation of reliable server pooling is that it is ambiguous to access the content distributed network and more over it increases the frequency of the server load. Han et al. [1] proposed a method which used round trip times to decide which server to route and select. The traffic which includes for the calculation of round trip times was the drawback of this method because there might be a chance of the nearest server not being selected because of the increased round trip times due to the traffic between the client and its nearest

server. In both the cases the challenge was the network delay in the server selection. The delay may be due to the two reasons. One is the load states get out of date due to the network latency. Second one being the ambiguity in the cache. Than the performance metrics from provider's perspective is that provider should think that weather the server load is going to gain the revenue or not. And from the user's perspective, user should think that how much time does it takes to process his requests. The handling speed of server capacity depends on queuing of the requests and the start up delay.

Chellouche et al. [2] proposed a technique called as anycasting for the routing of the request to a server. An anycast group will be formed among the set of servers which is solely based on the network metrics, and then select the server randomly out of the anycast group. The drawback of the anycast technique is that it is difficult to deploy any cast system and though it is supported by IPV6 infrastructure, IPV4 did not support this. The reason being there is no address range set on the public IPV4 internet for this technique. By the literature survey of the techniques for the server selection, it is very clear that more work was done on the network parameters when compared to the end user perception.

Wendell et al. [3] proposed a system called DONAR that considers both the client performance as well as the server load. This technique maintains a division of end user requests for the similar content. One cannot gain optimized solution since weighted function for server selection is not the good decision making technique. In several projects on server selection, a round robin mechanism is also been used for server selection policy. Since the server is selected on the round robin fashion, both the network parameters and server load were ignored. This simple server selection method cannot be used for server load balancing which is a dynamic method. Almeida et al [4] proposed a method to minimize the objective function of a weighted sum of network and server bandwidth to select the server. Still user perception cannot be solved by linear sum of network parameters.

A fundamental observation is that all the above proposed methods consider only the network and service parameters and the concentration was less on the end user perception. The rational decision is that good network parameters induce a good end user perception. However, this is not always correct since content is delivered to end users through the network in different contexts.

III. PROPOSED METHOD

A. Overview

This paper proposes a server load balancing mechanism called as transaction least work left (TLWL). It helps to the client for selecting appropriate server, which is having video files. Two main parameters play vital role in this paper one is selection layer and the second one being the routing layer. The server selection layer selects the appropriate server among the replica servers in the network to improve the end user perception and also to improve the network quality based of subject to this paper. The selection layer is independent of the upcoming routing layer. After the selection layer selects the better server the routing layer has to route the new end user request to the selected server by the previous layer. Transaction Least Work Left routes a new call to the server that has the good end user perception, and it calculates the least work left by summing up the work left in each of the server and then the server which is having the least work left will be given the new request. Here work is based on relative estimates of transaction costs. The functions performed in the proposed method are it reduces the server overhead by replicating the contents to the replica servers. By using the transaction least work left, this paper achieves efficient server load balancing also. Ultimately the goal, end user perception can be efficiently achieved.

B. Architecture

The proposed architecture is as shown in the Fig. 1. Where, flow starts from the client request. When the client wishes his request to the server, the request is navigated to the network controller, where the network controller is the gateway which contains the server profiles, that is nothing but the details of the number of available servers in terms of port number and the IP address.

After the request is received from the network controller it checks the work details of all the available servers in the list. The work details of the servers include the number of requests that the server is processing, the work done in seconds, work left in seconds. Based on the number of work left, the new request will be navigated to the server which is having the least amount of work left based on the number of requests. Then, the TLWL technique starts implementing the above explained theoretical concepts. Each time when new requests arrive to the network controller, it starts the actual calculations of the technical aspects based on the actual length of the video. Here the technical aspects in terms of video are, the joining time of the video, finishing time of the video, and when there is a new request from the end user. It calculates the duration of the video played, and the duration yet to be played from the total duration of the video. Here based on the TLWL algorithm the server starts responding to the client request immediately without any delay. Irrespective of the number of servers, the proposed technique has to calculate the technical aspects of the videos being played by the server. The work done of the server will be the difference between the current time of the local host and the join time of the video. And work left will be the difference between total duration of video and work done (duration of video in seconds).

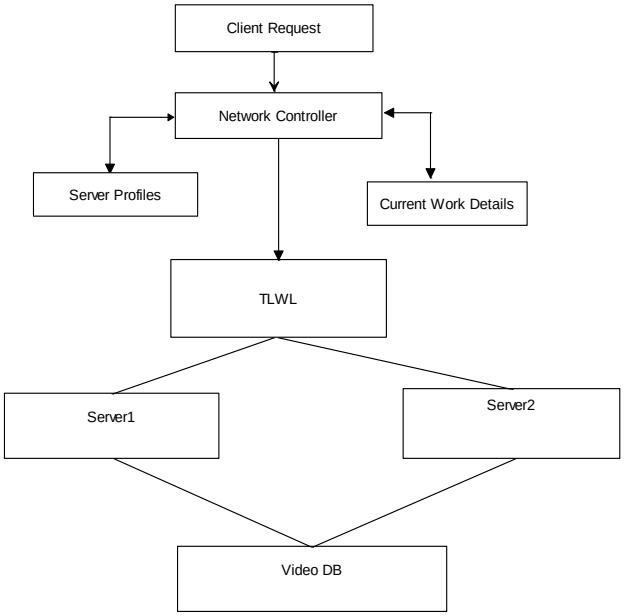

Fig. 1 Proposed Architecture

For the storage of the videos, the technique gives you two options based on the amount of available data in both. First one is to store the videos in the specific cloud storage system. This would be better for the applications which takes larger databases and which requires huge amount of storage. Second option is that if the application does not need a large amount of storage space the videos can be stored in the local host.

C. Working

The actual working of the TLWL technique keeps updating as and when the new requests for playing video arrives. When the first request arrives, since all the servers are idle, the request will be assigned randomly assigned. In the scenario where the requests keep arriving instantly than the actual working of the TLWL starts. The technique fetches the details of the duration of the video. Consider for example, if there are two replica servers, two videos are running in both the servers, than rather going for the number of requests, it calculates the work done and work left in both the servers. For two videos it sums up the work left for both the servers, and it compares the work left (in seconds) among both the servers, and then it redirects the new requests to the server which is having the least number of work left. There by achieving the efficient server load balancing as well as appropriate server selection. Before this it fetches the time from the local host, this helps when the user wishes to watch a video, the time is taken as joining time of the video, and then based on the total duration of the video, and the finishing time is calculated.

So the calculations of all the above mentioned points will help in the navigation of request to the appropriate server. In the scenario where a single user wishes to view multiple videos or the multiple users viewing different videos, each of the available servers, which reduce the overhead in handling loads for the servers.

As described above regarding process to optimize the data delivery tasks, the server selection layer carries out the server selection process that chooses the optimal replica server with respect to performance measure that needs to be defined.

D. Equations

1. Calculation of work done.

$$WD = CT - JT$$

2. Calculation of work left.

$$WL = VD – WD$$

Where, WD = work done in seconds.
CT = current time in seconds.
JT = Join time in seconds.
WL = work left in seconds.
VD = video duration in seconds.

E. Server selection and Routing layers

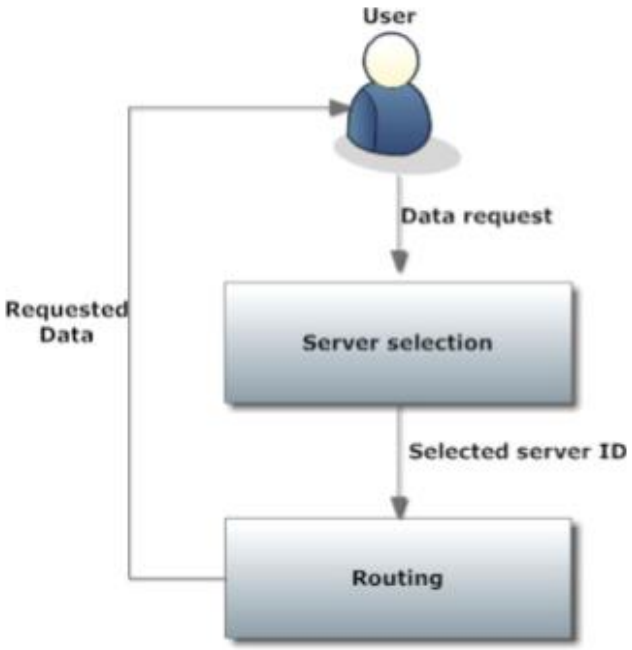

Fig 2. Two layers in replica server architecture.

As explained in the previous sections, it is two-layer architecture. Where the user's data request will be processed to the server selection layer. Server is assigned a request with the affiliated server id and then request is routed to the routing layer which helps the server to respond to the client.

IV. CONCLUSION

This paper proposed a server load balancing method based on the transaction least work left technique. This approach efficiently balances the load of the replica servers. Also the technique optimally increases the end user's perception. Adding to this, network parameters are not solely sufficient for the optimization of the end user perception and satisfaction, which is a crucial aim of the network operators.

V. FUTURE WORK

Though the TLWL method improves the end user perception, it lags in fetching the duration of the video. One has to rely on the end user who is uploading the video is assumed that he provides a proper duration. In future the methods can be proposed and work on the fetching of length of the video.

REFERENCES

[1] Z. Wang and J. Crowcroft, "Quality of Service Routing for Supporting Multimedia Applications," IEEE J. Selected Areas in Comm., vol. 14, no. 7, pp. 1228-1234, Sept. 1996.
[2] X. Zhou, T. Dreibholz, and E. Rathgeb, "A New Server Selection Strategy for Reliable Server Pooling in Widely Distributed Environments," Proc. Second Int'l Conf. Digital Soc., pp. 171-177, 2008.
[3] H. Tran, A. Mellouk, and S. Hoceini, "User to User Adaptive Routing Based on QoE," Proc. 10th Int'l Symp. Programming and Systems (ISPS), pp. 39-45, 2011.
[4] P. Brooks and B. Hestnes, "User Measures of Quality of Experience: Why Being Objective and Quantitative Is Important," IEEE Network, vol. 24, no. 2, pp. 8-13, Mar./Apr. 2010.
[5] J. Shaikh, M. Fiedler, and D. Collange, "Quality of Experience 1Metrics and Performance Evaluation," Annals of Telecomm., vol. 65, no. 1-2, pp. 47-57, 2010.

[6] S. Mohamed and G. Rubino, "A Study of Real-Time Packet Video Quality Using Random Neural Networks," IEEE Trans. Circuits and Systems for Video Technology, vol. 12, no. 12, pp. 1071-1083, Dec. 2002.
[7] S. Chikkerur, V. Sundaram, M. Reisslein, and L. Karam, "Objective Video Quality Assessment Methods: A Classification, Review, and Performance Comparison," IEEE Trans. Broadcasting, vol. 57, no. 2, pp. 165-182, June 2011.
[8] S. Hemminger, "Network Emulation with NetEm," Proc. Linux Conf. Au, Apr. 2005.
[9] P. Wendell, J. Jiang, M. Freedman, and J. Rexford, "Donar: Decentralized Server Selection for Cloud Services," ACM SIGCOMM Computer Comm. Rev., vol. 40, no. 4, pp. 231-242, 2010.
[10] J Hilbe, Logistic Regression Models. CRC Press, 2009.

