Improving the Service Time of Web Clients using Server Redirection

Oscar Ardaiz, Felix Freitag, Leandro Navarro
Computer Architecture Department,
Universidad Politécnica de Catalunya, Spain
{oardaiz, felix, leandro}@ac.upc.es

Abstract

This paper describes and evaluates experimentally a web server infrastructure, which consists of a small number of servers that redirect client requests based on the estimated client service time. The web servers have replicated content, are located in geographically different regions, and redirect clients between servers. The web servers use metrics obtained from server logs to estimate the service time of a client. Based on the estimated service time the server redirects the web client. The implementation of the measurement and redirection mechanism is done in the web servers and is independent of the clients. Using server logs the measuring mechanism does not introduce traffic into the network. We have experimentally evaluated the proposed web server infrastructure. In our experiments the client service time improved from 4 to 40 % when using the proposed mechanism. The web server infrastructure could be applied to improve the service time of selected clients, which frequently access a web server to retrieve a significant amount of data.

1. INTRODUCTION

Web clients experience a different service time from a server on every connection depending on many parameters, some of them with infinite variance or long-range dependencies [6]. Higher values of service time can be due for instance to heavy server load, network congestion, or a combination of both. Experimental evidence of different service times is given in Figure 1. It shows the time measured at a server when a client retrieves a 100 kb document every 10 minutes during 5 days. A cyclical component of 144 samples, i.e. a period of 1 day, in the service time can be observed.

When a Web client connects to servers located in different geographical regions, it may find that the peaks and valleys of the experienced client service time from two servers are time-shifted, see Figure 2, since the work hours of different time zones do not coincide. It appears that the congestion of these servers and their local networks do not happen at the same time intervals.

Mechanism to improve the service time of clients could be implemented in the client or in the web server/service provider's infrastructure. Considering the first case a client could be equipped with a measuring and decision taking mechanism, facilitating for instance to access mirrors and to retrieve the objects of documents from different servers. In this case, however, the client becomes more complex. More hardware resources will be needed to execute such a client, which may not be desirable.

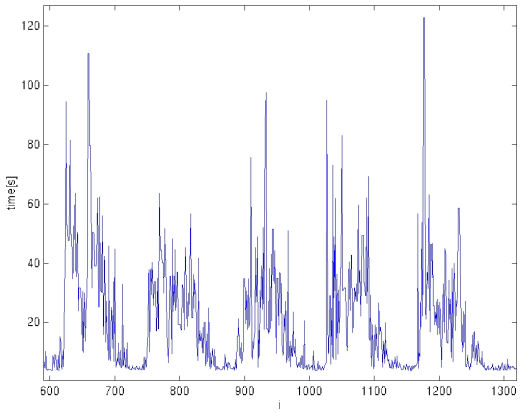


Figure 1. Time measured at server S_{es} to send a 100 kb document to client C_{es}. The measurement shown refers to 5 days, 720 samples, and client request every 10 minutes.

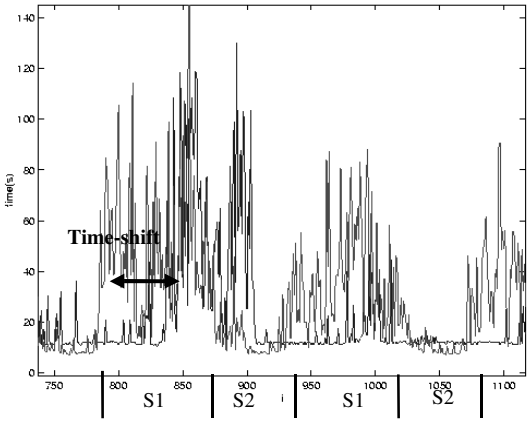


Figure 2. Service time experienced by client when requesting a document from 2 geographically different servers. Time-shift of peaks and valleys is observed.

In order to avoid the need of complex clients, another approach is to implement strategies for client service time reduction in the web server or service provider's infrastructure. Strategies on the server side to improve the client service time have the advantage that they are transparent to the client, requiring no additional effort from the client. Examples of mechanisms, which improve the service time of clients, are content distribution networks [4], as for instance Akamai [1].

Improving the client service time requires an infrastructure that measures, compares, and identifies resources for a certain client at a given moment, both in the case that this mechanism is implemented close to the client or in the client itself, or if there is an infrastructure transparent to the client in the service provider's infrastructure. The approaches to improve the client service time on the service provider's side can be classified according to [3] into DNS request-routing, transport-layer request-routing, and application-layer request routing. Systems, which were proposed and use one of these mechanisms are for instance Pinger [5], which offer measurements at the network layer, Idmaps [8] to create a global network of measurement elements, and Sonar [9] to modify the DNS service with measurements of proximity between Internet points.

In is paper we evaluate experimentally a web server infrastructure consisting of a small number of web servers located in geographically different regions with the goal to improve the service time of a limited number of clients, which frequently access the web server. Similar to the approaches above mentioned our infrastructure needs to measure and a mechanism to perform dynamic optimization of the client service time. Our approach is located at the application layer. We measure the total elapsed service time of objects at the web server, which is a good measure [7]. In comparison to other approaches, our measuring mechanism does not introduce traffic into the network. It provides information at the application layer, which includes information about network and server load, and it is used to redirect clients to the most adequate server.

The rest of the paper is structured as follows: In section 2 we describe the implementation of the proposed system and how performance is measured. In section 3 the experimental setup and preliminary results, which motivate our approach, are described. Section 4 presents the evaluation the proposed system. In section 5 we discuss our results and approach. In section 6 we conclude the paper.

2. IMPLEMENTATION

2.1 Metric for estimating the client service time

We wish to estimate on the server the client service time for a certain document. The estimation of the service time is used to decide which server will deliver the required document. Figure 3 illustrates the metric we use to measure the client service time. The process of establishing the TCP connection and the delivery of a single document can be seen. We define the client service time as the time measured in the client from the beginning of establishing the TCP connection till the client has received the document. The client perceives the client service time (named "total client time" in Figure 3), while on the server side different measures are available in the server's log file.

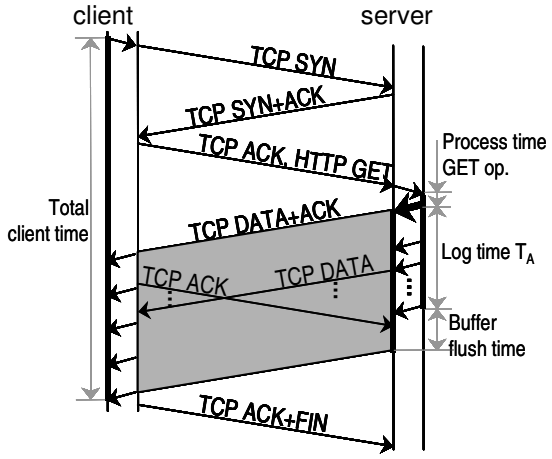


Figure 3. Client service time and time TA available in the log file of the web server.

We use the time TA available in the server logs, see Figure 3, to estimate on the server the client service time. Although the time TA is affected by several parameters, one important parameter is the bandwidth available at the server. The bandwidth influences on the client service time, for which the time TA appears to contain sufficient information to estimate the client service time. An evaluation of the time TA to estimate the perceived client service time can be found in [2].

2.2 Location of the server with replicated content

To establish the proposed web server infrastructure we need a small number of web servers with replicated content. There should be a correlation with a time shift between the characteristics of the different servers/connections. The characteristics of the server/connection are detected by the TA values measured in the server. Having a time shift between the server characteristics indicates that the traffic congestion in the nodes close to the available servers do not happen at the same time instants. Then, a client can access at each instant at least one server, which provides good service characteristics. In our experiment we have selected two servers with replicated content in geographically different regions with a different time zone, assuming that the time shift also shifts traffic and server congestion.

2.3 Selection of the best available server and redirection

Comparing the estimated client service time of each server, the server with the best-estimated service time is chosen to deliver the document to the client. The proposed server infrastructure may need a common database about the past performance of the web clients to estimate the client service time of each server. The server to which the client connects carries out the redirection of the client to the estimated best server.

We use the TA values to estimate the client service time. A model is needed to compute the estimation based on the TA values. The purpose is to perform an estimation of the client service time of different servers in order to identify the best server. We do not need to know an accurate quantitative value of the client service time. The model, which we use, see equation (1), is obtained from empirical observation and assumes that the client service can be estimated from the last three TA values, where in our experiment the clients connect every ten minutes,

$$\hat{Ta}(i) = \frac{Ta(i-1) + Ta(i-2) + Ta(i-3)}{3} \quad . \qquad (1)$$

3. EXPERIMENTS AND PRELIMINARY RESULTS

3.1 Experimental setup

We evaluate experimentally the proposed web server infrastructure using two servers and two clients (Figure 4). We set up two web servers in different geographic regions and different time zones, where one server is located in Spain (server S_{es}) and the other server is located in Ecuador (server S_{ec}). We also use two clients where the first client is located in Argentina (client C_{ar}) and the second client is located in Spain (client C_{es}). There is a distance of several hops between the client C_{es} and the server S_{es}.

On the servers S_{es} and S_{ec} the Apache v1.3 web server is installed. The clients C_{es} and C_{ar} are implemented using the Webbot client [10]. Both servers S_{es} and S_{ec} are provided with replicated contents. In our experiment the clients request a document, which consists of a 100kb html file with a 1kb image located at the end of the html file. As will be discussed in section 3.2, the document may not be too small, since the latency due to redirection must be compensated by the better performance of the other server or connection.

The experiment to validate the proposed web server infrastructure is carried out during 14 days. The two clients connect to the servers every 10 minutes. We obtain 1550 useable logs during the experiment in the client and servers log file to evaluate the performance gain.

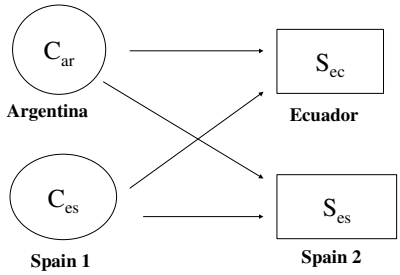


Figure 4. Web clients C_{es} and C_{ar} and web servers S_{es} and S_{ec}.

3.2 Influence of redirection on the client service time

The implementation of the redirection mechanism in the application layer introduces additional latencies. In this section we present some experimental results on the additional client service time, which introduces the redirection mechanism of the servers to the client service time.

Our experiment consists in measuring the additional time, which the redirection mechanism introduces to the client service time. In this experiment four clients C_{es} and four clients C_{ar} are created. The clients C_{es} and C_{ar} connect every 10 minutes to the servers S_{es} and S_{ec}. The document required by the client form the server is a 100kb html document with a 1kb image, which we described previously. The client service time of the following 4 connections are measured: 1) C_{es} to S_{es}; 2) C_{es} to S_{ec} and redirected to S_{es}, 3) C_{es} to S_{es}; 4) C_{es} to S_{es} and redirected to S_{ec}. We compute the additional cost of redirection as the difference between the service times measured in connections 1 and 2, and 3 and 4, respectively. The same experiment is carried out with client C_{ar}.

In Figure 5 the average obtained client service time measured in the client is shown. It can be seen that the service time of a redirected web client increases. In Table 1 the % increase of the service time experienced by the clients is shown when connecting to a server that redirects. In Table 1 it can be seen that in three cases the increase of service time due to redirection is less than 10 %. Only in the case C_{ar} - S_{ec} through redirection by S_{es} we obtain a 34% increase. In this case the client C_{ar} obtains the document from the server S_{ec} after C_{ar} connected initially to S_{ec} and is redirected by S_{es} to S_{ec}. The time added by redirection can be approximated as 0.5 RTT of the first server the client accesses, and 1.5 RTT of the second server, which then provides the document.

In order to make worth redirection, the cost of redirection must be compensated by an improved performance of the second server or connection. The challenge is to redirect

when the other server offers a significantly better service time.

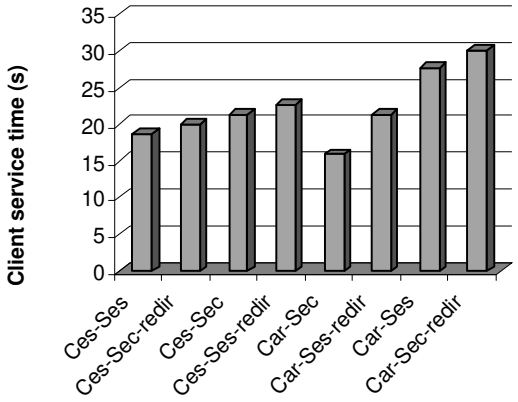


Figure 5. Service time perceived by the clients C_{ar} and C_{es}.

	% increase service time client to server S_{es} redirected by S_{ec}	% increase service time client to server S_{ec} redirected by S_{ec}
C_{es}	7,51 %	6,21 %
C_{ar}	8,90 %	34,31 %

Table 1. % increase of service time when server redirects clients.

3.3. Obtaining the best service time for the clients

In the previous section we have shown the average service time perceived by the clients C_{ar} and C_{es} for eight possible connections. In order to motivate our approach we show in this section that the clients could obtain a significantly better service if they always connect to the better server.

In Figure 6 we compare the service time perceived by the clients C_{ar} and C_{es} with the service time perceived when the clients connect always to the server, which provides a better service. It can be seen in Figure 6 that a client can obtain a better service if it connects always to the most appropriate server. Since the average service time is smaller if the client connects to different servers, it can be deduced that for a certain client at some instants the server S_{es} provides a better service, at other instants it is the server S_{ec}. During certain time periods a certain server provides a better service to a client, at other times a different server provides a better service. In this case, such as it can be seen in Figure 6, the ideal case offers the best client service time. This observation is important since it gives reason to look for a mechanism, which redirects clients if another server is expected to provide a better service.

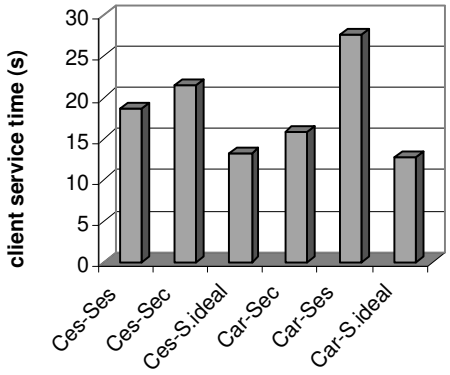


Figure 6. Comparison with the ideal service time (S.ideal=ideal server).

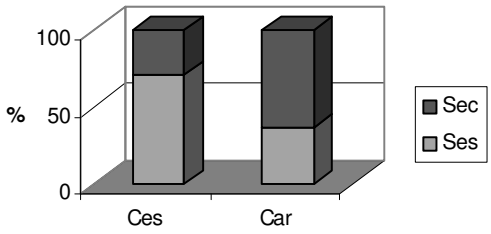


Figure 7. Distribution of client requests in order to obtain the best service time.

In Figure 7 we show in more detail how the client connections to servers are distributed in order to obtain the best service time. It can be seen that in more than 60% of the connections the server S_{ec} is better for the client C_{ar}. The client C_{es} obtains a better delivery from the server S_{es} in about 70% of the connections.

3.4 Evaluation of server log measure

Our redirection decision is based on the data available in the server logs. In this section we verify the accuracy of the TA values we use to estimate the client service time.

The servers have in their log files data about their own performance to a certain client in past accesses. We use the TA metric available in the server log file to know about the perceived performance of a client (see section 2.2). The metric is described in detail in [2]. In Figure 8 we show the accuracy of the TA measure to determine which server is better. In order to obtain this result we compare the TA values from both servers and identify the better server by the lower TA value. If the client service time measured in the client is also lower, then the TA measure in the server has identified correctly where the client obtained a better service. It can be seen that by comparing the TA values from the servers we identify correctly with about 80% and 75% accuracy, on which server the client perceived better performance.

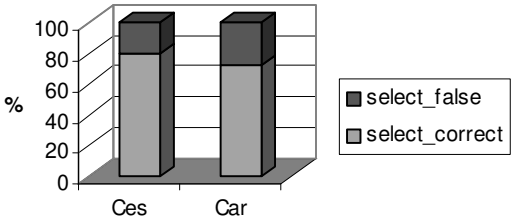


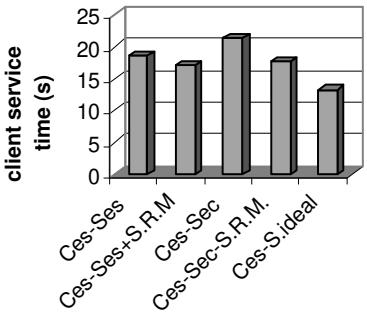
Figure 8. Accuracy of TA metric to determine the better client service time.

4. EVALUATION

In this section we analyze and evaluate the client service time gain when using the proposed server infrastructure and redirection mechanism.

4.1 Improvement of the client service time

In Figure 9 the average service time perceived by the clients C_{es} and C_{ar} are shown for different web server infrastructures. It can be seen that the client service time improves when using the proposed mechanism. The improvement obtained is important for the cases Ces-Sec-S.R.M and Car-Ses-S.R.M. As it is shown in the following section, these clients are redirected most of the time. From the numbers shown in Figure 9 we compute the improvement of the client service time when using the proposed mechanism. We obtain an improvement of the client service time from 3.5 % up to 39.8 % with the proposed system, see Table 2, compared to a web server infrastructure where the mechanism is not implemented. In the cases Car-Sec-S.R.M and Ces-Ses-S.R.M. a less important improvement is achieved. In these two cases the client and first accessed server are in a similar time zone or same geographic region, respectively. The experimental data, however, indicates that even for these cases it is beneficial for the client that it connects to a geographically distant replicated web server at certain times instead of to the physically closest provider.



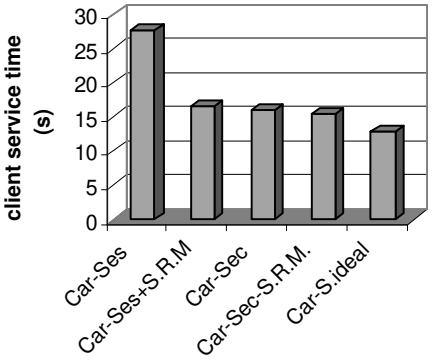


Figure 9. Perceived client service time for different server configurations (with S.R.M.= Server redirection mechanism, S.Ideal =with ideal redirection).

	Ces-Ses-S.R.M.	Ces-Sec-S.R.M.	Car-Ses-S.R.M.	Car-Sec-S.R.M.
Improvement (%)	8,00	17,28	39,85	3,55

Table 2. Improvement of the client service time.

4.2 How often redirection is used?

For this experiment we consider two clients C_{es} and two clients C_{ar}. The four clients establish the following connections: 1) C_{es}-S_{es}, 2) C_{es}-S_{ec}, 3) C_{ar}-S_{es}, 4) C_{ar}-S_{ec}. Our experiment consists in evaluating how often a client is redirected by the initially accessed server to the other server.

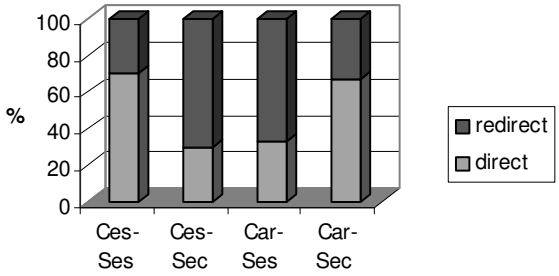


Figure 10. Redirection percentage of web clients.

In Figure 10 the experimental results are shown. It can be seen that the clients establishing the connections C_{es}-S_{es} and C_{ar}-S_{ec} are redirected in only about 30% of their requests. On the other hand, clients establishing the connection C_{es}-S_{ec} and C_{ar}-S_{es} are redirected about 70% of the times to the server S_{es} and S_{ec}, respectively.

5. DISCUSSION AND FUTURE WORK

In the previous sections we have shown experimentally that we obtain an improvement of the client service time when using the proposed server infrastructure. In this section we discuss some aspects of this approach, where further study may provide additional insight:

Cost of redirection: Though it was shown experimentally that the client service time improved even when adding the cost of redirection, the proportion of the redirection cost increases when the size of the documents to be delivered becomes small. Redirection can become particularly beneficially when large replicated documents need to be delivered, so that the cost of redirection is compensated by a better performance of the chosen connection.

Selection of replicated server: In our experiments we have selected two servers considering that they are in geographically different regions such that they belong to different time zones. With this selection we have achieved that the peak traffic hours in the nodes close to the servers do not coincide, making it beneficial for clients to connect to a geographical distant server in the peak traffic hours instead of to the physically closest server. Creating maps and taking into account locations of particular clients may improve the selection of the server with replicated content.

Estimation of performance: The estimation of the client service time by the servers was done using the average of the last three connections, see equation (1). Although this estimation worked in our experiment, other ways to estimate the future performance may provide more accurate results.

Scalability: In our experiment the improvement of the client service time was shown for a configuration with two servers and two clients, where the clients access the servers frequently. The clients, who can be accommodated with the service improvement, are those, about which the servers are able to estimate future performance. Currently, we have done the estimations using historical data. In order to extend to other clients, the knowledge about network topology could be helpful for clustering clients. Concerning the increase of the number of server, we expect that the improvement on the service time will reach a saturation point, once there is a server available with a valley in service time at every time instant.

6. CONCLUSIONS

We have presented a web server infrastructure consisting of a number of servers where servers implement a measuring and redirection mechanism in the application layer in order to improve the client service time. The proposed mechanism requires the estimation of the future performance a client perceives by the servers. It is shown how the future client service time is estimated. The client request is redirected to another server if this is estimated to provide a better service. In the experimental evaluation of the proposed system with two servers and two clients we obtain an improvements of the perceived client service time from 4 to 40 %, compared to the case when the mechanism is not implemented. The experimental results indicate that small content providers could use the proposed server infrastructure and the redirection mechanism to improve the client service time of selected clients.

ACKNOWLEDGEMENTS

This study was partially financed by the CICYT-MCYT. Our gratitude to Narcís Vives de Espais Telematics S. L. (Spain), to Claudio Righetti from Prima (Argentina), Joseph Turró from Eucanex (Ecuador), and Julián Casasbuenas from Colnodo (Colombia) for the permission to carry out tests from their machines and networks.

REFERENCES

[1] Akamai Inc., "Freeflow", Dic 1999, http://www.akamai.com/

[2] O.Ardaiz, F.Freitag, L.Navarro. "Estimating the Service Time of Web Clients using Server Logs", ACM Sigcomm Workshop on Data Communication in Latin America and the Caribbean, San Jose, April 2001.

[3] A. Barbier, B. Cain, F. Douglis, M. Green, M. Hofmann, R. Nair, D.Potter, O. Spatscheck, "Known CDN-Request-Routing-Mechanisms" IETF draft draft-barbir-cdnp-known-request-routing-00.txt, 2001.

[4] J. Chase, M. Rabinovich, "Web Caching and Content Distribution: A view from the Interior", 5th Web Caching Workshop, Lisbon, May 2000.

[5] L. Cotrell, "PingER project at Stanford", December 1999 http://.slac.stanford.edu/comp/net/wan-mon/

[6] M. Crovella, A. Bestavros, "Self-Similarity in World Wide Web Traffic: Evidence and Possible Causes". In Proc ACM Sigmetrics'96, pages 160-169, May 1996.

[7] S. Dykes, K Robbins, C. Jeffery "An Empirical Evaluation of Client side Server Selection Algorithms", Proc. of IEEE Infocom'00, March 2000

[8] P. Francis, S. Jasmin, V. Paxson, L. Zhang, D.F. Gryniewicz, Y. Jin "An Architecture for a Global Internet Host Distance Estimation Service", Proceedings INFOCOM99, 1999.

[9] K. Moore, J. Cox, S. Green, "Sonar – a network proximity service", available at http://www.netlib.org/utk/projects/sonar/ February 1996.

[10] Webbot. http://www.w3.org/Robot