Evaluating User-perceived Benefits of Content Distribution Networks

Claudia Canali
University of Parma
claudia@weblab.ing.unimo.it

Valeria Cardellini
University of Roma "Tor Vergata"
cardellini@ing.uniroma2.it

Michele Colajanni
University of Modena
colajanni@unimo.it

Riccardo Lancellotti
University of Modena
lancellotti.riccardo@unimore.it

Abstract

Content Distribution Networks (CDNs) are a class of successful content delivery architectures used by the most popular Web sites to enhance their performance. The basic idea is to address Internet bottleneck issues by replicating and caching the content of the customer Web sites and to serve it from the edge of the network. In this paper we evaluate to what extent the use of a CDN can improve the user-perceived response time. We consider a large set of scenarios with different network conditions and client connections, that have not been examined in previous studies. We found that CDNs can offer significative performance gain in normal network conditions, but the advantage of using CDNs can be reduced by heavy network traffic. Moreover, if CDN usage is not carefully designed, the achieved speedup can be suboptimal.

Keywords: Content delivery, Caching, End-to-end performance, Edge servers.

1 INTRODUCTION

When client requests have to reach the origin server and the responses must travel backwards, many network and server bottlenecks may affect the user-perceived Web performance. There are three main possible network bottlenecks. The *first mile* that is, the network link between the origin Web server and the Internet, can become congested if it is underprovisioned. The *last mile* that is, the connection between the end user and the Internet, is another known source of performance problems. The third critical network point is represented by the *peering points* among Autonomous Systems, because they are seldom over-sized due to large costs that few providers are interested to or can afford. The server side represents another potential bottleneck, also because of the continuously increasing demand for more complex services.

There are two opposite approaches to face the performance problems of Web content delivery. In the *core model* a Web cluster, consisting of locally distributed servers [3], can solve most problems related to the server side, but it is unable to address network-related issues.

The *network edge model* aims to a complete or partial replication of the Web site content over geographically distributed servers. There are two main approaches to the so called *edge delivery*: the distributed architecture of multiple servers is managed by the content provider or it is delegated to a third party. The former solution can be convenient for a Web site that has a permanent popularity, even although a minority of content providers can afford the complexity of setting up and managing a geographically distributed architecture. As a consequence, the latter seems the most viable solution, especially when the Web site has to deal with flash crowds and short periods of intensive traffic. This outsourcing alternative has created a new market of *Content Distribution Network* (CDN) companies. Many have appeared (and disappeared), and now two or three share the largest part of the market [9]. The largest companies (i.e., Akamai, Mirror Image, Speedera) provide an infrastructure of thousands of geographically distributed Web farms (called *edge servers*), most of which are placed at the edge of the Internet.

The basic philosophy of CDN architectures is to improve performance by cooperative pro-active caching. With respect to traditional proxy caches [10], a CDN solution can achieve much higher cache hit rates. The CDNs must not deal with all Web content, because their working set is limited to the content of the customer Web sites. Moreover, the edge servers work in cooperation with the origin servers, hence they can use mechanisms typical of the reverse proxy technology.

The main limitation of CDN services is due to their costs that are still expensive. Their performance gain has not been widely studied due to the proprietary and closed nature of CDN architectures: indeed, most studies have been carried out by CDN companies themselves. Hence, it is of key importance to give independent evaluations of the CDN companies claims. Our purpose is to study when this outsourcing solution provides real performance benefits to the end users.

We have developed a new tool (called **CDNperf**) that analyzes and compares the user-perceived response time of content delivery achieved with and without the use of CDNs. Our study considers a large set of network and system conditions during different periods, covering in all a length of time of almost two years. To the best of our knowledge, no other study have analyzed CDN performance for such a long time. This allows us to provide some original insights to the performance of CDN-based delivery. Moreover, this paper integrates and extends some recent works [1, 8, 9].

This paper extends from a different perspective the work by Krishnamurthy et al. [9], which represents to now the most extensive study on CDN performance evaluation. First, we

employ the real Web pages encountered in actual Web sites, not using a canonical Web page. Second, we evaluate the benefits deriving by the introduction of CDNs comparing the page response time measured when using or not the CDN service. Third, we examine the contribution of DNS lookup time to the page response time. The Medusa proxy tool [1, 8] has been used to evaluate the performance of CDNs limited to the Akamai company. Our network-oriented focus considering different Internet traffic conditions, client locations, and last mile bandwidths extends the analysis of CDN performance to a wider range of parameters.

The CDN ability to select the edge server with the minimum latency to the client has been analyzed in [5]; however, the authors analyze only the relative performance of server selection within a given CDN. A performance comparison of CDNs against traditional Web delivery and peer-to-peer file sharing systems has been conducted in [11]. Other performance studies have been carried out through simulation: Kangasharju et al. show that the retrieval of objects in the same Web page from multiple servers may cause a performance degradation [6]. However, due to the complexity of the real infrastructure to be modeled, we believe that analytical and simulation techniques are well suited to evaluate new research ideas, rather than to analyze the performance of existing CDN architectures. A commercial evaluation of CDN performance is provided by Keynote Systems [7], whose service allows to compare the downloads of single objects and full pages served or not by the CDN using a global infrastructure of measurement computers.

The rest of this paper is organized as follows. Section 2 provides an overview of the main routing and delivery mechanisms adopted by CDN architectures. Section 3 describes the evaluation methodology we used to collect and process our performance data. Section 4 discusses the main features of the CDNperf tool. Section 5 presents our study on a significant set of Web sites whose content is delivered by a CDN. Section 6 concludes the paper with some final remarks.

2 ROUTING AND DELIVERY MECHANISMS

In this section we review the main phases involved in the content service from the CDN infrastructure to the consumers. We identify three core phases: the *selection* phase to determine the edge server/s that is/are considered best suited to respond, the *request routing* phase in which a proper mechanism is used to direct the client request to the target edge server(s), and the *delivery* phase during which the requested content is transferred to the client.

The selection and request routing phases may be interleaved, as a CDN can adopt some complex routing mechanism acting at different network levels. The server selection phase typically chooses the "nearest" server to the requesting client. The evaluation of the proximity among clients and edge servers is usually a function of network topology and dynamic link characteristics. It is likely that CDNs apply even more sophisticated algorithms taking into account server-related factors; however, in our analysis the consequences of these algorithms never emerged.

As a main focus of this paper is to evaluate the performance impact of CDN architectures at the network level, we review the most used request routing mechanisms in CDNs, that can be divided into the classes of DNS-based and application-layer mechanisms. The use of the authoritative DNS server of the Web site as request dispatcher has been initially proposed for locally and geographically distributed Web-server systems [3]. In the CDN case, the authoritative DNS server of the origin site delegates to the modified authoritative DNS server of the CDN company the resolution for those hostnames whose content is delivered through the CDN. In [9] this approach is referred to as *full-site content delivery*, being the origin server completely hidden to clients. Besides the well-known DNS limitation on request control due to the address caching mechanisms, the full-site content delivery approach has the drawback of a coarse-grained, content-blind routing decision [3, 10]. Moreover, as in the full-site content delivery approach the origin server is completely hidden, it is not possible to compare the user-perceived performance gain achieved by CDNs.

Therefore, in this paper we focus on the performance evaluation of the alternative class of architectures, where CDNs use DNS-based routing in combination with some application-layer routing mechanism. Another motivation is that this solution has a wider spread usage and a major flexibility with respect to the full-site content delivery approach. In this scheme, called *partial-site content delivery* [9], a client request first reaches the origin server, which then applies an application-layer routing mechanism [10] to redirect subsequent requests to an edge server. Through URL rewriting, which is the application-layer mechanism adopted by all Web sites considered in our analysis (HTTP redirection is the other commonly used one), the origin server changes dynamically the links for the embedded objects within the requested Web page, so that they point to another node. In such a way, the container page is returned by the origin server and all (or most) embedded objects are served by some other node(s). The URL rewriting mechanism is typically used in combination with DNS-based routing [4, 10]. The hostnames of the embedded object URLs are rewritten with those of the edge servers, which are resolved to a corresponding IP address in a subsequent step by the CDN's DNS infrastructure, possibly by using multiple tiers of proprietary name servers combined with very low Time-To-Live values [4].

During the content delivery phase, which completes the service, all the objects composing the requested Web page are transferred to the client. The number of entities that carry

out the delivery clearly depends on the selection phase and the request routing mechanism adopted by the CDN. Under the partial-site content delivery approach, the origin server provides the container page, while the embedded objects are delivered by one or more edge servers.

3 EVALUATION METHODOLOGY

As our goal is to measure to what extent the use of CDNs improves the user-perceived response time with respect to a non CDN-based service and the performance dependency on different network and site parameters, we investigate the following issues, that have been grouped on the basis of the three phases of CDN service.

- *Server selection*: we aim to understand how CDN performance is influenced by external factors such as end-user geographical location, network conditions, time of the day, day of the week.
- *Request routing*: we aim to single out the DNS contribution to the user-perceived performance.
- *Delivery*: we aim to determine the number of distinct edge servers used for a single Web site, and how this number affects the user-perceived performance.

As regards the server selection phase, to analyze the user-perceived impact of the factors external to the CDN infrastructure, we consider three client locations having different types of network connection, and three measurement periods (covering two years), characterized by a different worldwide network usage. Indeed, previous studies about CDNs [9] have highlighted a considerable variability of the results obtained in experiments carried out during distinct time periods, also due to changes in the CDN infrastructure. On the other hand, no previous study has analyzed the hourly and daily dependence of measured CDN performance. We conduct the performance analysis using real pages of Web sites that adopt CDN services rather than using a canonical Web page that reflects the statistical distribution of static objects typically served by a given CDN [9], or client collected traces [1].

DNS redirection impact on CDN performance has been investigated in recent studies [1, 2, 8, 9], which confirm that CDNs reduce mean response times, but that DNS-based request routing techniques add a noticeable overhead. As DNS redirection techniques are transparent to the client, it is difficult to understand their inner mechanism. We measure the DNS resolution time at the client side and analyze to what extent the DNS lookup cost affects the page response time and how this contribution changes with the network conditions.

As main performance metrics we use the cumulative distribution, the median and 90-percentile of the response time perceived by end users. Indeed, the mean response time, which is the most common performance metric, may be not meaningful in an Internet context, where response times show heavy-tailed distributions. As observed in [1], the overall page performance is the crucial metric which users are most interested in and the content providers should also focus on it, as it directly correlates with the user perception of the quality of service of a CDN system. The page response time, which corresponds to the interval between the submission of the page request and the arrival at the client of all objects related to the page request, includes the DNS resolution time, the TCP connection time, all delays at the servers, and the network transmission time.

Another problem that may affect a fair performance comparison is related to the fact that Web pages have an extremely different number of embedded objects. Hence, we normalize the CDN page response time with respect to the time when all files are downloaded from the origin server. To this purpose, our tool retrieves two versions of the same Web page: the former in which objects served by the CDN infrastructure are requested to the CDN edge servers, the latter in which requests are forced to reach the origin server(s). The metric, called *speedup*, is defined as the ratio between the non-CDN and the CDN usage case, hence $speedup > 1$ means a performance improvement determined by the CDN usage.

4 EVALUATION TOOL

The CDNPerf tool, which implements the above evaluation methodology, consists of three parts, each one related to a specific step of the testing process that is, experiment configuration, request generation, and output analysis and report.

The main engine of CDNPerf is the browser emulator. This program is an HTTP client that supports most functionalities of the HTTP protocol including persistent connections, chuncked encoding transfers, and request redirection. CDNPerf is able to recognize CDN-served embedded objects and hence it can download them from both the customer origin servers and the CDN edge servers.

Given a URL referring to a Web page, CDNPerf downloads all its components and records different performance metrics in a log file. After having retrieved the container page, CDNPerf parses it to identify each embedded object and to determine whether it is CDN-enabled or not. Then, for each identified server (both origin and edge) CDNPerf downloads each served object by (re)using a persistent connection, and records the DNS resolution time and the TCP connection setup time (in case of multiple connections due to network problems or lack of persistent connection support, multiple connection times are stored). The retrieval of embedded objects by the origin server is forced using the original URL rather than the rewritten one. For each requested object the tool records the total download time and the latency. Finally, when all the objects in the page have been retrieved, CDNPerf records the aggregate response time for the whole page (obtained by summing the total download time for all objects),

by distinguishing those obtained by a CDN server and those by the origin server. For resources not served by the CDN, the origin server statistics are used for both aggregate times.

Some characteristics of our browser emulator resemble those provided by the Medusa proxy [1, 8]. Specifically, both tools retrieve two versions of the same Web page, the former served by the CDN infrastructure, the latter served only by the origin server. However, the Medusa proxy ignores DNS refresh effects due to a fairly small inter-request interval and its transformation feature is limited to Akamaized URLs [4].

To simplify the measurement analysis task of the log file of the experiments, we used a structured format with a syntax similar to YAML. The log is composed by a series of *stanzas*, each one describing one URL download attempt. Each stanza is composed of multiple second level entities (one for each server); the last line contains the aggregate performance data.

The data analyzer component of CDNPerf is responsible for processing the log files and producing the percentiles and cumulative distributions of the selected performance metrics.

5 EXPERIMENTAL RESULTS

Many network and system parameters can affect the performance of CDN services. These parameters are in part internal to the CDN architecture (e.g., server selection, routing and delivery mechanisms, server placement, percentage of objects served from edge servers) and in part are out of the CDN company control (e.g., Internet traffic, bandwidth of the client connection). Most of the previous studies have focused on the former aspects; in this paper, we include the internal parameters, but their analysis starts from the external aspects. To this purpose, we consider two Internet traffic conditions and three types of client connection to the CDN architecture.

The experiments referring to the normal traffic lasted over two distinct periods: nearly two months from October 5, 2002 to November 30, 2002 and two weeks at the beginning of February 2004. In these periods, we did not observe any special peak of Internet traffic (we excluded December and periods of special politics and sport events). We repeated the same experiments for about one month (from March 14, 2003 to April 10, 2003) in a worldwide Internet condition characterized by heavier traffic, due to international political events (i.e., Iraq crisis). The main visible effect was a round-trip time between the same points clearly higher than that observed in the other periods. In all periods, we considered three types of client-to-CDN connections.

- High Quality (**HQ**) location (in USA): very large connection bandwidth (16 Mbps), 8 network hops to the closest edge server, and a RTT ranging from 1.6 to 12.7 ms with an average of 1.9 ms over 24 hours (measured in a period of high traffic).
- Medium Quality (**MQ**) location (in Europe): medium connection bandwidth (4 Mbps), 11 network hops to the closest edge server, and a RTT ranging from 13 to 29 ms with an average of 22 ms.
- Low Quality (**MQ**) location (in Europe): low connection bandwidth (1 Mbps), 13 network hops to the closest edge server, and a RTT ranging from 26 to 120 ms with an average of 52 ms.

Due to space limits, we report the most significant results related to medium and low bandwidth for normal traffic, including also the large bandwidth case for high traffic. We denote the combinations through Normal-MQ and Normal-LQ, High-HQ, High-MQ, and High-LQ.

For the server side of our tests, we selected the home page of 20 Web sites served by two CDN providers (75% and 25% by Akamai and Speedera, respectively), that are indicated among their most popular customers. We do not report their names for the sake of privacy and also because no previous authorization has been asked to.

5.1 Overall performance of CDN services

The first important test is to verify for which external factors the usage of a CDN service can effectively reduce the user-perceived response time.

Table 1: Response time and speedup (Normal-MQ).

Content provider	CDN		No CDN		Speed-up	
	median	90-perc	median	90-perc	median	90-perc
CP_1	2.635	8.252	8.943	16.683	3.393	2.021
CP_2	2.897	12.542	6.195	17.194	2.137	1.370
CP_3	2.747	16.775	6.062	17.693	2.206	1.054
CP_4	5.943	17.021	13.034	17.664	2.193	1.037
CP_5	12.243	35.576	11.671	16.499	0.953	0.463
CP_6	2.697	7.540	12.343	19.130	4.575	2.537
CP_7	1.376	5.974	3.134	9.495	2.277	1.589
CP_8	5.808	11.043	5.753	9.229	0.990	0.8357
CP_9	3.426	11.143	5.402	11.142	1.576	1.000
CP_{10}	1.636	6.974	5.704	10.542	3.485	1.511
CP_{11}	4.951	8.649	6.648	11.501	1.342	1.329
CP_{12}	1.409	8.280	4.767	12.249	3.383	1.479
CP_{13}	4.225	9.279	4.722	8.055	1.117	0.868
CP_{14}	2.390	7.586	2.822	7.797	1.180	1.027
CP_{15}	3.626	7.431	9.716	13.703	2.679	1.843
CP_{16}	1.594	6.184	4.558	8.962	2.859	1.449
CP_{17}	2.017	6.724	3.453	8.407	1.711	1.250
CP_{18}	2.506	9.009	3.307	11.210	1.319	1.244
CP_{19}	3.167	9.693	7.975	13.814	2.517	1.425
CP_{20}	1.618	11.795	3.097	13.266	1.914	1.124

Let us first consider the situation of normal traffic. During the two periods referring to the normal traffic we observe consistent measurements for most of Web sites, although almost two years pass between the first and the last experiment. Hence, the first interesting result is that, even if CDN techniques have evolved over this time, the perceived performance in terms of speedup is not changed significantly. This consistency is an important reference point to understand how CDN performance changes under different system and network conditions. Table 1 reports in columns from 2 to 5 the median and the 90-percentile of the user response time for all

Web sites (all time values are in seconds). Columns 6 and 7 show the speedup for the two performance metrics. For three content providers (CP_7, CP_8 and CP_{20}) we report results only for the first period, because afterward they dismissed partial site distribution with CDN. Table 1 shows the performance for the MQ location, but the speedup values are very similar for the HQ and LQ locations (for Normal-HQ the response time is one order of magnitude lower). Table 1 shows that in most of the cases CDN offers a significative performance advantage. Considering the 90-percentile of response time, in 10% of the observed sites the speedup is higher than 2, up to 2.5 for CP_6. For 25% of the sites, the speedup is higher than 1.5, and for 50% higher than 1.3. However, in 15% of our observations a CDN service can be slower than that provided by the origin server. In particular, for CP_5 the speedup is 0.46 that is, the CDN response time doubles with respect to that of the origin server. We have identified that the possible cause of this performance penalty is the high number of edge servers used to deliver objects to the same client.

Let us now consider the situation of high traffic. We anticipate that when the Internet load becomes heavier, the conclusions about CDN performance change significantly. Table 2 reports the speedup for High-LQ, High-MQ, and High-HQ: as expected, the response time tends to increase. Moreover, by comparing Table 1 and 2, the speedup appears to be generally lower. In particular, the speedup on 90-percentile is reduced by an increase of pathological cases where the CDN cannot deal with congestion. For this reason, more than 70% of the content providers in High-LQ have a slowdown in response time in case of CDN usage, and the highest achieved speedup is 1.76 for CP_{17}. The results of the various locations are similar for median values (even if the better connectivity in High-HQ offers greater speedup), while, due to the less predictable nature of congested networks, the results of 90-percentile show some differences, with 18% of content providers with a speedup lower than 1 for High-MQ and nearly 40% for High-HQ. Comparing with the normal traffic, we note that few content providers increase their performance. In many cases this can be explained with changes in the CDN usage policy: for example, for CP_5 the performance is increased as the usage of edge servers is modified (see Section 5.3). Moreover, three content providers (CP_7, CP_8, and CP_{20}) choose to dismiss partial site distribution with CDN (hence they are missing in Table 2). However, there are many sites (such as CP_1) for which CDN usage (and to a limited extent also performance) is comparable in both experiments.

Another advantage of CDN usage verified in our experiments is the ability to reduce variance in response time. Figure 1 shows the cumulative probability distribution of user response time for a sample site (CP_1) in both network conditions (the data are related to the MQ location). We choose this content provider because it is the most popular site we studied, but the considerations can be applied to most of the sites. Notwithstanding the performance differences, the figure shows that the cumulative distribution of CDN is far steeper: this means that CDN can be effective in reducing performance variability.

Table 2: Speedup (High-LQ, High-MQ, High-HQ).

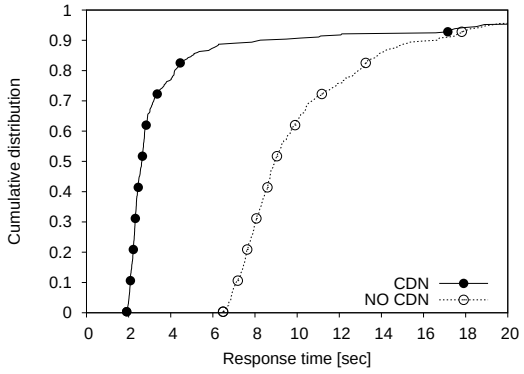
Content provider	High-LQ		High-MQ		High-HQ	
	median	90-perc	median	90-perc	median	90-perc
CP_1	2.652	0.826	2.678	2.201	5.518	4.825
CP_2	1.613	0.991	1.495	1.106	1.436	0.976
CP_3	0.985	1.064	0.694	0.752	n/a	n/a
CP_4	1.183	0.667	1.563	1.743	1.268	0.757
CP_5	1.031	0.770	0.845	0.867	0.930	0.729
CP_6	1.223	0.780	1.269	1.000	1.461	1.268
CP_9	1.509	1.266	1.317	1.746	1.083	0.927
CP_{10}	1.563	0.582	1.546	1.170	2.610	2.225
CP_{11}	1.812	1.023	1.679	1.221	2.767	1.264
CP_{12}	2.042	0.781	2.154	1.822	1.480	1.313
CP_{13}	1.034	0.576	1.054	1.245	0.855	0.870
CP_{15}	1.836	0.717	2.003	1.534	2.275	1.742
CP_{16}	1.422	0.581	1.734	1.291	2.693	1.592
CP_{17}	2.510	1.759	2.284	1.561	3.696	3.497
CP_{18}	1.116	0.850	1.026	0.993	1.554	1.269
CP_{19}	1.649	0.750	1.421	1.255	2.768	2.047

By comparing Figures 1(a) and 1(b) we have a visual confirmation of the Internet congestion: Figure 1(b) shows smoother curves, that tend to have a higher probability of pathological cases with a very high response time. This also reduces the CDN advantage on 90-percentile, up to the case where CDN services do not achieve any further performance improvement. From this figure we can conclude that in case of high network traffic, CDN can still offer some performance gain, but this advantage is much less evident with respect to the case of low-medium traffic. Even the CDN ability to limit performance variance is reduced as well.

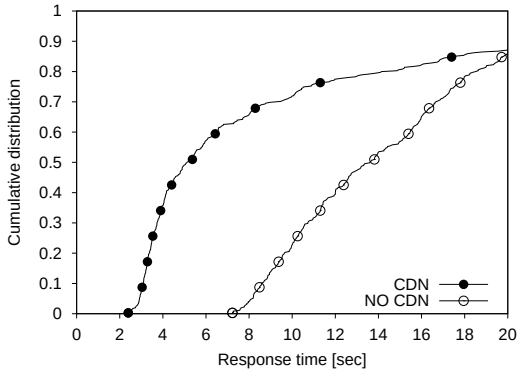
5.2 Hourly dependence of CDN performance

The hour of the day and the day of the week are other external factors that influence CDN performance and that have not been previously examined. Quite surprisingly, we found that most Web sites show a similar behavior, hence we report only two graphs (Figure 2) that show the median response times as a function of the hour of the day for normal and high traffic conditions, respectively. Another premise is in order: there is a strict correlation between night hours and weekend days performance, and between daylight hours and week days performance. We focus just on the day hours, but other conclusions can be easily obtained.

In the case of normal traffic, Figure 2(a) shows that CDN performance are clearly better than the no-CDN case: the median values are about one third of no-CDN, and the CDN curve has less and smaller spikes than the no-CDN one. This confirms the previous observation that CDN services can effectively reduce the variance of the response time when the network load is limited. In Figure 2(a) we consider Normal-

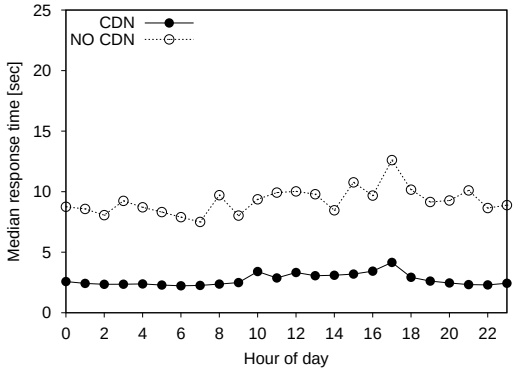


(a) Normal traffic

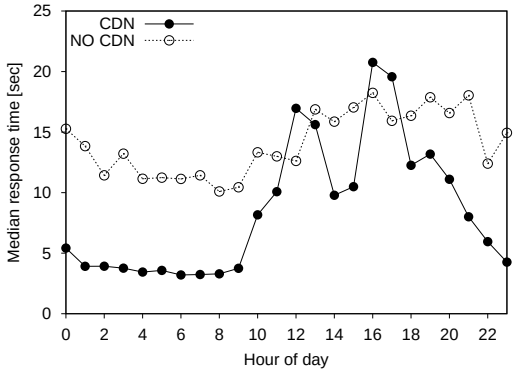


(b) High traffic

Figure 1: Cumulative distribution of response time for CP_1.



(a) Normal-MQ



(b) High-MQ, High-LQ

Figure 2: Hourly dependence of performance.

MQ, but same conclusions hold for the other client locations. We also note that edge servers belonging to the same physical region tend to show similar access patterns with respect to the hour of the day. This is the reason for the slight increase of the response time during day hours with respect to night hours. On the other hand, the origin server which receives requests from different geographic locations shows less predictable and regular response times.

In the case of high network traffic (Figure 2(b)), the client location has a significant impact on performance. When connections have low-medium quality, CDNs are no longer able to compensate possible congestion due to the network links. Hence, their response time shows a great difference between daytime and night. Moreover, CDNs seem unable to limit either response time or its variance, thus showing a behavior comparable with the no-CDN case. However, it is worth to note that heavy network load does not automatically result in bad performance. When the quality of the client connection is good (High-HQ), the performance results of CDNs are very similar to those shown in Figure 2(a).

5.3 Effects of internal mechanisms

As internal factors we consider the percentage of objects served by the edge servers and the number of edge servers used to deliver content to the same client. Table 3 shows the number of embedded objects in each home page (column 2), and the absolute number and percentage of embedded objects served from CDN edge servers (columns 3 and 4, respectively). The last column reports the number of edge servers used to deliver the respective embedded objects.

The most interesting result in Table 3 is that the large majority of CDNs use only one edge server. The use of multiple edge servers for the content providers CP_1 and CP_{10} is related to the logical subdivisions of the site content delivery. These sites use a main edge server for most embedded objects, and other edge servers for specific functions, such as dynamically generated images and/or advertising banners.

The most significative exception is the content provider CP_5, that uses a different edge server for each CDN-served embedded object. This is an interesting representative case that deserves some further discussion. The first observation is that this site performs poorly because of different reasons. If

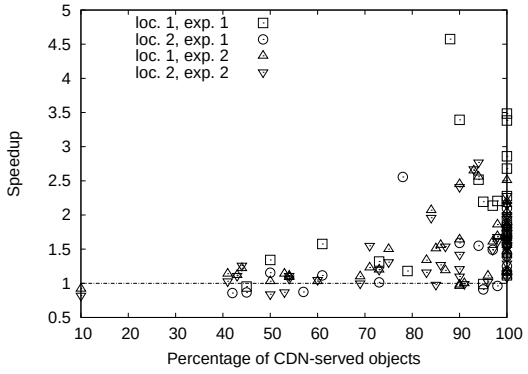
Table 3: Internal factors.

Content provider	Embedded objects			Edge servers
	Total	CDN-served	%	
CP_1	31	29	90%	3
CP_2	58	56	97%	1
CP_3	44	43	98%	1
CP_4	44	42	95%	1
CP_5	29	13	45%	13
CP_6	33	29	88%	1
CP_7	9	9	100%	6
CP_8	21	20	95%	1
CP_9	18	11	61%	1
CP_{10}	29	29	100%	2
CP_{11}	26	13	50%	1
CP_{12}	13	13	100%	1
CP_{13}	19	19	100%	1
CP_{14}	29	23	79%	1
CP_{15}	29	29	100%	1
CP_{16}	25	25	100%	1
CP_{17}	10	10	100%	1
CP_{18}	26	19	73%	1
CP_{19}	31	29	94%	1
CP_{20}	14	14	100%	1

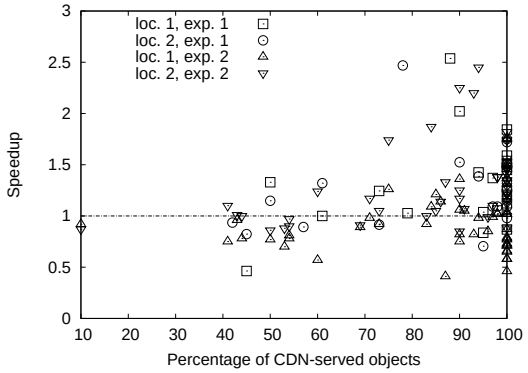
each edge server has to start a new TCP connection for each embedded object, the positive effects of persistent connections cannot be exploited. Moreover, as each edge server belongs to the same network area, the possible benefits of parallel download are reduced. The poor performance of CP_5, together with the choice of the large majority of CDN providers to use only one edge server, confirms the results found by Kangasharju et al. [6], showing that the retrieval of objects in the same Web page from multiple servers may cause a performance degradation. It is interesting to observe that the performance of CP_5 shows great improvements in the period of high traffic, when its internal architecture was changed to a single server delivery basis.

The relationship between speedup and percentage of CDN-served objects is shown in Figure 3. We report the speedup vs. percentage of CDN-served objects for both median (Figure 3(a)) and 90-percentile (Figure 3(b)). Both graphs show that: (1) to maximize the CDN performance benefits, CDN must be heavily used; (2) heavy usage of CDNs is only a necessary condition, and it is not sufficient to guarantee high speedup. From the first observation, we can conclude that CDNs are a powerful solution to increase Web performance, confirming the results in [8].

CDNs must be used carefully in order to maximize their potential benefits. From Figure 3 we can see that the sites relaying completely on the CDN infrastructure for the delivery of embedded objects show large speedup differences. The best performing sites seem to be those with a low percentage of origin-served objects. This apparently counter-intuitive result can be motivated by the fact that, when some objects are served by the origin server, there is a load distribution between the edge servers and the origin server itself. However, this solution requires a careful tuning to avoid congestion at the origin server. The same relationship between the fraction of CDN-served embedded objects and performance has been observed under different network conditions and from different client locations. Hence, our observations have a general validity because they are neither related to geographic location nor to the network traffic conditions.



(a) Median



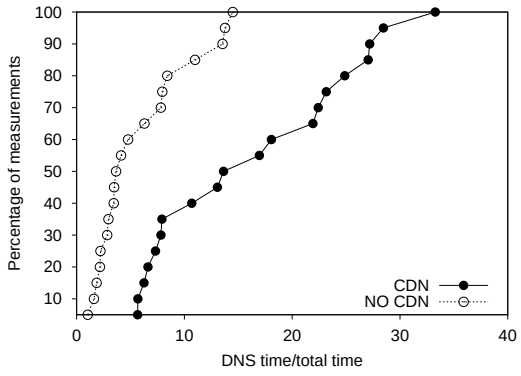
(b) 90-percentile

Figure 3: Speedup vs. percentage of CDN-served objects.

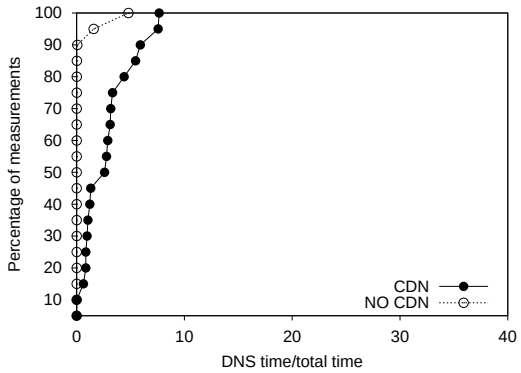
5.4 Impact of the routing mechanism

Previous studies [2, 9] have found that, when sophisticated DNS servers are used, the DNS lookup time tends to increase with respect to traditional DNS systems and our experiments confirm this result. For space limits, we focus on the comparison of DNS lookup time in the case of normal and heavy network traffic. Figure 4 shows the cumulative distribution of the median contribution of DNS lookup time to the page response time over the various content providers.

In the case of normal traffic, for any client location (Figure 4(a) refers to Normal-MQ, but the results are very similar for Normal-HQ and Normal-LQ), we observe that the DNS



(a) Normal traffic



(b) High traffic

Figure 4: DNS lookup time vs. page response time.

lookup time can be up to nearly 40% of the global response time in the case of CDN usage, while it is no more than 20% when CDNs are not used. Moreover, the DNS lookup time contributes for more than a tenth in 70% of the cases in which a CDN is used, while the percentage decrease to less than 40% in case of no CDN usage. This greater influence of the DNS lookup time in case of CDN usage is the sum of two combined effects: the DNS delay tends to increase because of the use of CDN; the global response time is reduced by the use of CDN in the case of normal traffic. When the traffic is higher, the network impact tends to increase the download time much more than the DNS lookup time (Figure 4(b)): the consequence is that the DNS lookup time has almost a negligible impact on the response time.

6 CONCLUSIONS

This paper addresses the evaluation of user-perceived CDN benefits. Using the developed CDNperf tool, we have studied two CDN companies using partial site delivery and our experiments, performed on real sites over two years, cover a significant range of network traffic and client situations.

Our experiments show that CDNs can offer significative performance gains, also reducing the response time variance, over the traditional solution with a centralized, and possibly far, Web server. However, we found that under heavy network traffic, the CDN benefits are reduced and CDNs can show heavy time-dependent behavior, with response times far higher during the busiest hours of the day. We found a strong correlation between the achieved speedup and the fraction of the CDN-served site. However, a heavy usage of CDN-enabled delivery is not sufficient to achieve high speedup. We also confim two results presented in previous studies: CDNs achieve better performance when only few edge servers are used, and the DNS resolution time can be a significative part of the total response time under normal traffic condition.

We conclude that CDNs are a powerful mechanism to increase the user-perceived Web performance. However, they are not a *panacea* that allow to arbitrarily improve the performance of a Web site: careful site and content distribution design is required to fully exploit the CDN potentialities. Moreover, in case of critical network conditions, the CDN performance can be reduced as well as for every node in a congested network.

References

[1] L. Bent and G. Voelker. Whole Page Performance. In *Proc. of 7th Int'l WCW Workshop*, Aug. 2002.

[2] A. Biliris, C. Cranor, F. Douglis, M. Rabinovich, S. Sibal, O. Spatscheck, and W. Sturm. CDN Brokering. In *Proc. of 7th Int'l WCW Workshop*, Aug. 2002.

[3] V. Cardellini, E. Casalicchio, M. Colajanni, and P. S. Yu. The State of the Art in Locally Distributed Web-server Systems. *ACM Computing Surveys*, 34(2):263–311, June 2002.

[4] J. Dilley, B. Maggs, J. Parikh, H. Prokop, R. Sitaraman, and B. Weihl. Globally Distributed Content Delivery. *IEEE Internet Computing*, 6(5):50–58, Sept./Oct. 2002.

[5] K. L. Johnson, J. F. Carr, M. S. Day, and M. F. Kaashoek. The Measured Performance of Content Distribution Networks. *Computer Commun.*, 24(1-2):202–206, Feb. 2001.

[6] J. Kangasharju, K. W. Ross, and J. W. Roberts. Performance Evaluation of Redirection Schemes in Content Distribution Networks. *Computer Commun.*, 24(1-2):207–214, Feb. 2001.

[7] Keynote Systems. http://www.keynote.com/.

[8] M. Koletsou and G. Voelker. The Medusa Proxy: A Tool for Exploring User-Perceived Web Performance. In *Proc. of 6th Int'l WCW Workshop*, 2001.

[9] B. Krishnamurthy, C. E. Wills, and Y. Zhang. On the Use and Performance of Content Distribution Networks. In *Proc. of SIGCOMM IMW 2001*, Nov. 2001.

[10] M. Rabinovich and O. Spatscheck. *Web Caching and Replication*. Addison Wesley, 2002.

[11] S. Saroiu, K. P. Gummadi, R. J. Dunn, S. D. Gribble, and H. M. Levy. An Analysis of Internet Content Delivery Systems. In *Proc. of OSDI 2002*, Dec. 2002.