Dissecting Video Server Selection Strategies in the YouTube CDN

Dissecting Video Server Selection Strategies in the YouTube CDN

Ruben Torres*, Alessandro Finamore†, Jin Ryong Kim*, Marco Mellia†, Maurizio M. Munafò† and Sanjay Rao*
*Purdue University, †Politecnico di Torino
{rtorresg,jessekim,sanjay}@purdue.edu, {alessandro.finamore,marco.mellia,maurizio.munafo}@polito.it

***Abstract*—In this paper, we conduct a detailed study of the YouTube CDN with a view to understanding the mechanisms and policies used to determine which data centers users download video from. Our analysis is conducted using week-long datasets simultaneously collected from the edge of five networks - two university campuses and three ISP networks - located in three different countries. We employ state-of-the-art delay-based geolocation techniques to find the geographical location of YouTube servers. A unique aspect of our work is that we perform our analysis on groups of related YouTube flows. This enables us to infer key aspects of the system design that would be difficult to glean by considering individual flows in isolation. Our results reveal that while the RTT between users and data centers plays a role in the video server selection process, a variety of other factors may influence this selection including load-balancing, diurnal effects, variations across DNS servers within a network, limited availability of rarely accessed video, and the need to alleviate hot-spots that may arise due to popular video content.**

***Keywords*-Content distribution networks; Web and internet services**

I. Introduction

Over the last few years, video traffic has become prominent on the Internet. A recent report [1] shows that 15 to 25% of all Inter-Autonomous System traffic today is video. YouTube is probably the main source of video on the Internet today, with 2 billion videos viewed each day and hundreds of thousands of new video uploads daily [2]. It is the third most visited website in the Internet, according to www.alexa.com.

The rapid growth in popularity of YouTube has made it the subject of several research studies. Much of the research to date has focused on understanding user behavior, usage patterns and video popularity [3]–[5], while others [6] have looked at social networking aspects related to YouTube. Relatively fewer works have looked at the YouTube infrastructure itself, and large parts of its architecture and design remain unknown to the research community. A recent notable work [7] has greatly contributed to the understanding of the YouTube Content Distribution Network (CDN) through an in-depth analysis of traffic traces of a tier-1 Internet Service Provider (ISP). However, much of this analysis has focused on the architecture prior to the acquisition of YouTube by Google Inc. It is unclear to what extent these observations continue to hold today.

In this paper, we aim to obtain a detailed understanding of the YouTube CDN and to quantify its effectiveness. Specifically, we are interested in studying how users' video requests are mapped to YouTube data centers. We are interested in exploring the various factors that can influence the decision, such as user proximity, server load, and popularity of video content. Such insights can aid ISPs in their capacity planning decisions given that YouTube is a large and rapidly growing share of Internet video traffic today. A better understanding could enable researchers to conduct what-if analysis, and explore how changes in video popularity distributions, or changes to the YouTube infrastructure design can impact ISP traffic patterns, as well as user performance.

Obtaining such understanding is challenging given the proprietary nature of the YouTube system. Even information such as the location of the data centers that store content is not publicly known. To tackle these challenges, we conduct an analysis of traffic from the edge of five networks - two university campuses and three ISP networks - located in three different countries and two distinct continents. We consider a one week-long dataset from each vantage point, all collected at the same time. This allows us to study the server selection algorithm under different scenarios, so that different phenomena may appear in some datasets but not in others. While prior work has analyzed traffic at the edge of a single campus network (for e.g., [3], [4]), our work goes far beyond in terms of the number and diversity of vantage points used.

As a first step, we map YouTube server IP addresses obtained from our datasets to the nearest data centers. Prior efforts at doing so [7], [8], have either relied on geolocation databases [9], or on reverse Domain Name System (DNS) lookup that can provide information regarding the server location. However, while these techniques worked with the earlier YouTube architecture, we find they do not apply or perform poorly in the new design. Consequently, we use CBG [10], a well known delay-based geolocation algorithm to learn server locations.

Armed with server location information, we evaluate how user requests are mapped to YouTube data centers. We show that there are two mechanisms: The first is based on DNS resolution which returns the server IP address in a data center; the second relies on application-layer mechanisms in which the server initially contacted can redirect the client to another server in a possibly different data center. Our results indicate that, given a network, most requests are directed to a preferred data center. This is in contrast to [7] which indicated that the earlier YouTube infrastructure would direct requests from a network to a data center proportional to the data center size.

Further, our results indicate that the RTT between data centers and clients in a network may play a role in the selection of the preferred data center.

More surprisingly however, our results also show that there do exist a significant number of instances where users are served from a data center that is not the preferred. Our analysis is informed by techniques we employ to identify groups of YouTube flows that correspond to a single video request. A deeper investigation reveals a variety of causes. These include load balancing across data centers, variations across DNS servers within a network, alleviation of hotspots due to popular video content, and accesses of sparse video content that may not be replicated across all data centers. Overall the results point to the complexity of server selection algorithms employed in YouTube, and the myriad factors that must be considered for the successful design of a large video content distribution network.

II. YouTube Basics

YouTube is the most popular video-sharing website on which users can watch videos on demand. It was bought by Google Inc. in November 2006 and it is now integrated in the Google offering. In this section we present a high level description of the steps to retrieve a video from the YouTube system as sketched in Figure 1.

When accessing videos from the YouTube site at www.youtube.com, the user either browses the portal based system looking for the desired content, or accesses directly the video web page following a video page URL (step 1). Until the actual video web page is accessed, mostly static information and small thumbnails of suggested videos are presented.

Once the actual video has been selected, the front-end replies with a HTML page in which the video is embedded using an Adobe Flash Player plugin, that takes care of the download and playback of the video (step 2). The name of the server that will provide the video is among the parameters provided for the plugin and it is encoded using a static URL. Then, the content server name is resolved to an IP address by the client via a DNS query to the local DNS server (step 3). Finally, the client will query the content server via HTTP to get the actual video data (step 4).

We further elaborate on steps 3 and 4. First, the selection of the IP address by the local DNS server in step 3 is not arbitrary. In fact, the DNS resolution is exploited by YouTube to route clients to appropriate servers according to various YouTube policies, some of which we will discuss in this paper. Second, it is possible that the preferred server cannot provide the content and the client will be "redirected" by this server to a different one, possibly in a different data center.

III. Methodology

To understand the internal mechanisms of the YouTube CDN, we need to analyze the interactions between the user and the content servers. We introduce our data collection tool in Section III-A, and describe our datasets in Section III-B.

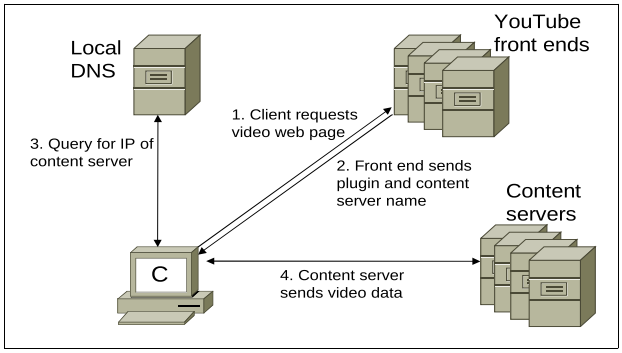


Fig. 1. High level sequence of steps to retrieve content.

A. Collection tool

Our traces are collected using Tstat [11], an Open Source passive sniffer with advanced traffic classification capabilities. Tstat identifies the application that generates TCP/UDP flows using a combination of Deep Packet Inspection (DPI) and statistical classifiers. Tstat was found to perform well in [12].

Tstat has the capability to identify major components of the current HTTP Web 2.0 traffic, including in particular YouTube traffic. Classification is achieved by using DPI technology to inspect the packet payload and then to identify YouTube service specific strings. In this paper we rely on Tstat's ability to identify actual YouTube video traffic, corresponding to the download of the Flash Video (flv) or H.264 (MP4) video file to be played back to the user by the Flash plugin. YouTube video downloads embedded in third party sites such as news sites or blogs are also correctly classified, since the same mechanisms are adopted by the Flash plugin. For more details on the classification algorithm implemented in Tstat, we refer the reader to the source code available from [13].

To uniquely identify a YouTube video, Tstat records the *video identifier* (VideoID), which is a unique 11 characters long string assigned by YouTube to the video. This is the same ID that is used when accessing the video web page in the URL. Furthermore, Tstat also records the actual resolution of the video being requested. At the end, the VideoID and resolution identify the actual video stream served to the player.

B. Datasets

Using Tstat, we collected datasets corresponding to flow-level logs where each line reports a set of statistics related to each YouTube video flow. Among other metrics, the source and destination IP addresses, the total number of bytes, the starting and ending time and both the VideoID and the resolution of the video requested are available.

We collected datasets from five locations spread across three countries including Points-of Presence (PoP) in nation-wide ISPs and University campuses. In all cases, a high-end PC running Tstat was installed to analyze in real time all the packets going to and coming from all the hosts in the monitored PoPs. For all these datasets, we focus on a one week time period, between September 4th and September 10th,

TABLE I
TRAFFIC SUMMARY FOR THE DATASETS

Dataset	YouTube flows	Volume [GB]	#Servers	#Clients
US-Campus	874649	7061.27	1985	20443
EU1-Campus	134789	580.25	1102	1113
EU1-ADSL	877443	3709.98	1977	8348
EU1-FTTH	91955	463.1	1081	997
EU2	513403	2834.99	1637	6552

2010. The collection from all vantage points starts at 12:00am, local time.

Table I summarizes the characteristics of the datasets, reporting the name, the total number of YouTube video flows and corresponding downloaded volume of bytes. Finally, the number of distinct IP addresses considering both YouTube servers and clients in the PoP are reported. In total, more than 2.4 millions YouTube videos have been observed by more than 37,000 users in the whole dataset.

We can divide the 5 datasets collected into two categories:
• **ISP Networks**: The datasets have been collected from nation-wide ISPs in two different European countries. EU1-ADSL and EU1-FTTH refer to data collected from two distinct PoPs within the same ISP. The two PoPs differ in the type of Internet access technology of their hosted customers. In EU1-ADSL, all customers are connected through ADSL links and in EU1-FTTH, all customers are connected through FTTH links. The EU1 ISP is the second largest provider in its country. The EU2 dataset has been collected at a PoP of the largest ISP in a different country.
• **Campus Networks**: The datasets have been collected using a methodology similar to the ISP setting. The Tstat PC is located at the edge of each of the campus networks, and all incoming and outgoing traffic is exposed to the monitor. We collected datasets from two University campus networks, one in the U.S. and one in a European country.

IV. AS LOCATION OF YOUTUBE SERVERS

We start our analysis studying the Autonomous System (AS) in which YouTube video servers are located. We employ the `whois` tool to map the server IP address to the corresponding AS. Table II presents our findings for each dataset. The second group of columns shows the percentage of servers and bytes sent from the Google AS. Not surprisingly, most servers are hosted in the Google AS (AS 15169). For instance, for the US-Campus dataset, 82.8% of the servers are located in the Google Inc. AS, serving 98.66% of all bytes. The third group of columns shows that a small percentage of servers (and an even smaller percentage of bytes) are still located in the YouTube-EU AS (AS 43515). We therefore have an evidence that since 2009 Google has migrated most content from the YouTube original infrastructure (that was based on third party CDNs) to its own CDN. The traffic served from the YouTube networks is probably because of legacy configurations. This contrasts with earlier studies such as [7], [8], according to which the majority of servers were located in the YouTube AS (AS 36561, now not used anymore).

TABLE II
PERCENTAGE OF SERVERS AND BYTES RECEIVED PER AS

Dataset	AS 15169 Google Inc.		AS 43515 YouTube-EU		Same AS		Others	
	servers	bytes	servers	bytes	servers	bytes	servers	bytes
US-Campus	82.8	98.96	15.6	1.03	0	0	1.4	0.01
EU1-Campus	72.2	97.8	20.3	1.6	0	0	7.5	0.6
EU1-ADSL	67.7	98.8	28	0.94	0	0	4.3	0.26
EU1-FTTH	70.8	99	24.2	0.83	0	0	5	0.27
EU2	62.9	49.2	28.6	10.4	1.1	38.6	7.4	1.8

The fourth group of columns in Table II shows the percentage of servers and bytes received from within the same AS where the dataset have been collected. Note that the values are 0 for all datasets except EU2. The EU2 dataset indeed shows that a YouTube data center is present inside the ISP network. This data center serves 38.6% of the bytes in the EU2 dataset. This results in the EU2 dataset having fairly different performance than other datasets, as our analysis will reveal later.

Finally, the last groups of columns aggregates the percentage of servers and bytes sent from other ASes, among which CW (AS1273) and GBLX (AS3549) are the most likely one. This confirms therefore that YouTube servers can be both present inside an ISP, or in the Google network.

In the rest of this paper, we only focus on accesses to video servers located in the Google AS. For the EU2 dataset, we include accesses to the data center located inside the corresponding ISP.

V. SERVER GEOLOCATION

In this section we present the techniques used to identify the geographical location of the YouTube servers seen in our datasets. The goal is to later use this information to analyze the video server selection policies.
• **Limitations of IP-to-location databases:** One common way to find the geographical location of an IP address is to rely on public databases [8]. While such databases are fairly accurate for IPs belonging to commercial ISPs, they are known to be inaccurate for geolocation of internal IPs of large corporate networks. For example, according to the Maxmind database [9], all YouTube content servers found in the datasets should be located in *Mountain View, California, USA*. To verify this, we perform RTT measurements from each of our vantage points to all content servers found in our datasets. Figure 2 reports the Cumulative Distribution Function (CDF) of the minimum RTT obtained to each server. We clearly observe that there is a lot of variation in the measurements, and in particular, many of the RTT measurements for the European connections are too small to be compatible with intercontinental propagation time constraints [14]. This indicates that all servers cannot be located in the same place.

We note that Maxmind was useful in [8], probably because most YouTube servers in the old infrastructure were reported as located in San Mateo and Mountain View, California, USA. Further, a recent work [7] adopts a different approach, where

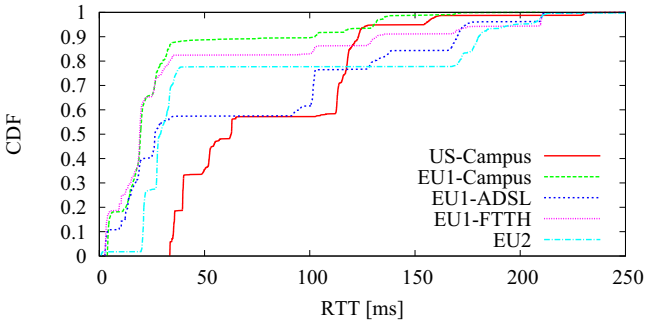


Fig. 2. RTT to YouTube content servers from each of our vantage points.

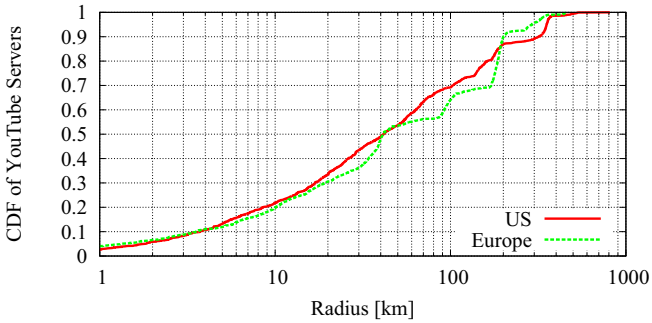


Fig. 3. Radius of the CBG confidence region for the YouTube servers found in the datasets.

the location of the server is obtained directly from the server name. However, this approach is not applicable to the new YouTube infrastructure, where DNS reverse lookup is not allowed. Therefore we decided to adopt a measurement-based approach to systematically localize YouTube servers.

• **Measurement based geolocation mechanism:** CBG [10] is a well-known geolocation algorithm that is based on simple triangulation. A set of landmarks is used to measure the RTT to a target. A simple linear function is then used to estimate the physical distance between each landmark and the target. This distance will become the radius of a circle around the landmark where the target must be located. The intersection among all circles is the area in which the target can be located.

We obtained the CBG tool from Gueye et al. [10] for our evaluations. We used 215 PlanetLab nodes as landmarks: 97 in North America, 82 in Europe, 24 in Asia, 8 in South America, 3 in Oceania and 1 in Africa. Then, we run RTT measurements from each landmark to each of the YouTube servers that have been found in our dataset, and identified the area in which they are placed.

In Figure 3 we evaluate the confidence region of CBG, i.e. the area inside which the target IP should be located. The picture shows the CDF of the radius of the confidence region for all servers found. Separate curves are shown for IPs in U.S. and Europe. Note that the median for both U.S. and European servers is 41km, while the 90th percentile is 320km and 200km, respectively. This is in the ballpark of the PlanetLab experiments presented in [10], where the 90th percentile for U.S. and Europe was about 400km and 130km. We can therefore consider the results provided by CBG to be more than adequate for our analysis.

• **Geolocation Results:** Table III details the result of using CBG to identify the location of all the destination IPs found in the datasets. The table shows the number of servers that are located in North America, Europe and other continents. Interestingly in each of the datasets, at least 10% of the accessed servers are in a different continent.

Finally, since several servers actually fall in a very similar area, we consider all the YouTube servers found in all the datasets and aggregate them into the same "data center". In particular, servers are grouped into the same data center if they are located in the same city according to CBG. We note that all servers with IP addresses in the same /24 subnet are always aggregated to the same data center using this approach. We found a total of 33 data centers in our datasets, 14 in Europe, 13 in USA and 6 in other places around the world. These results may not cover the complete set of YouTube servers since we are only considering those servers that appeared in our dataset.

TABLE III
GOOGLE SERVERS PER CONTINENT ON EACH DATASET.

Dataset	N. America	Europe	Others
US-Campus	1464	112	84
EU1-Campus	82	713	1
EU1-ADSL	518	769	51
EU1-FTTH	90	631	44
EU2	233	815	0

VI. EVALUATING YOUTUBE'S SERVER SELECTION ALGORITHM

In the previous section, we have shown how IP addresses of YouTube servers may be mapped to the appropriate YouTube data centers. Armed with such information, we now try to understand how user video requests are mapped to YouTube data centers. We are interested in exploring the various factors that can influence the decision, such as user proximity, server load, and popularity of content. We begin by discussing the various types of flows in a YouTube session, and then discuss how content servers are selected.

A. Video flows and sessions

In conducting our analysis, it is important to note that when a user attempts to download a video, the overall interaction may include a group of distinct flows, not all of which involve transfer of video. In the normal scenario, each YouTube video request corresponds to a HTTP message exchanged between the Flash Plugin and a content server. If the request succeeds, then the content server starts to deliver the video inside the open connection. It is possible however that the server may not serve the content. In such a case, it would simply *redirect* the user to another content server and close the connection. There may be other possible responses from the server, for e.g., a response indicating that change of video resolution is required. Thus, more generally, according to the reply of the

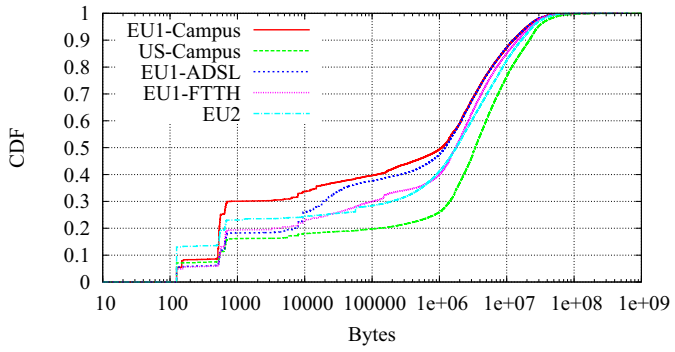


Fig. 4. CDF of YouTube flow sizes.

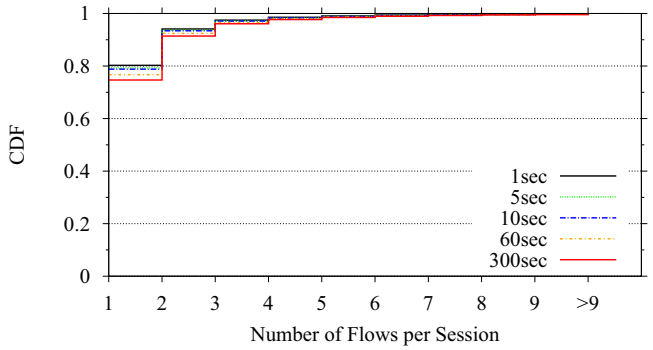


Fig. 5. Number of flows per session with different values of T for the US-Campus dataset.

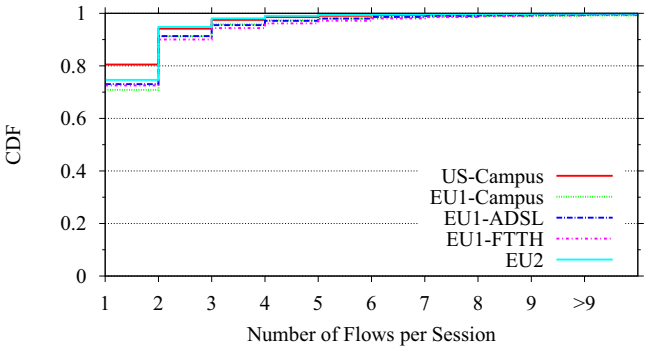


Fig. 6. Number of flows per session for all datasets using T=1 second

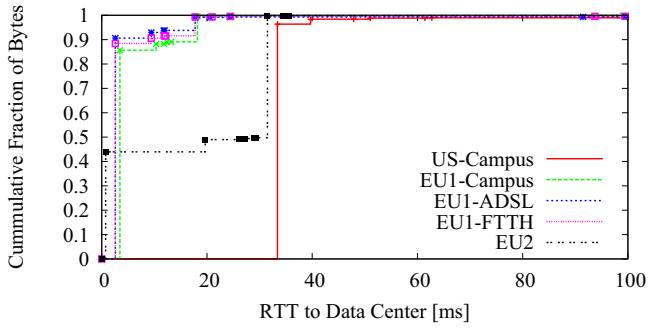


Fig. 7. Fraction of the total YouTube video traffic served by a data center with an RTT less than a given value from the dataset collection point.

content server, we can distinguish between *video flows*, i.e., long connections carrying the requested video, and *control flows*, i.e., short connections carrying signaling messages.

Knowledge of control flows associated with a video flow can help provide important insights for our analysis. For instance, a video flow from a user to a given server preceded closely (in time) by a control flow to another server is an indication of *redirection*. In contrast, an isolated video flow not preceded by other control flows is an indication that the request was directly served by the contacted server. We refer to such a group of related flows as a *video session*. Identification of video sessions aid our analysis as we will see later.

We now discuss how we identify video flows and sessions. Since Tstat classifies YouTube video flows based on the URL in the HTTP requests, it is not able to distinguish between successful video flows and control messages. To overcome this limitation, we employ a simple heuristic based on the size of the flows involved. Figure 4 presents a CDF of YouTube video flow sizes. Log-scale is used on the x-axis. We notice the distinct kink in the curve, which is due to the two types of flows. Based on this, we separate flows into two groups according to their size: flows smaller than 1000 bytes, which correspond to *control flows*, and the rest of the flows, which corresponds to *video flows*. We have conducted manual experiments which have confirmed that flows smaller than 1000 bytes are indeed control messages.

A *video session* aggregates all flows that i) have the same source IP address and VideoID, and ii) are overlapped in time. In particular, we consider two flows to overlap in time if the end of the first flow and the beginning of the second flow are separated by less than T seconds. In general, we find that small values of T will group flows triggered by the system, while large values of T may also group flows generated by user interactions with the video player, such as changing the video resolution and pausing or fast-forwarding a video. Since we are interested in capturing server redirections, which are triggered by the system, we want to use a small value of T, but that is large enough to avoid artificially separating related flows. Hence, we perform sensitivity to the value of T in our traces. We show results for the US-Campus dataset in Figure 5 and note that other traces show similar trends. Results indicate that values of T equal to 10 seconds or less generate similar number of sessions. So we pick the smallest value of T in our evaluations, $T = 1$ second.

Figure 6 reports the CDF of the number of flows per session for each dataset, assuming $T = 1$ second. It shows that $72.5 - 80.5\%$ of the sessions consist of a single (long) flow. Therefore, normally there is no need to iterate over different servers to download the video data. However, $19.5 - 27.5\%$ of the sessions consist of at least 2 flows, showing that the use of application-layer redirection is not insignificant.

B. Understanding server selection strategy

In Table III we have shown that the users in each dataset contact content servers all over the world. It is now interesting to investigate how the volume of traffic downloaded is spread across the different data centers. Figure 7 reports the fraction of traffic served by each data center versus the RTT between

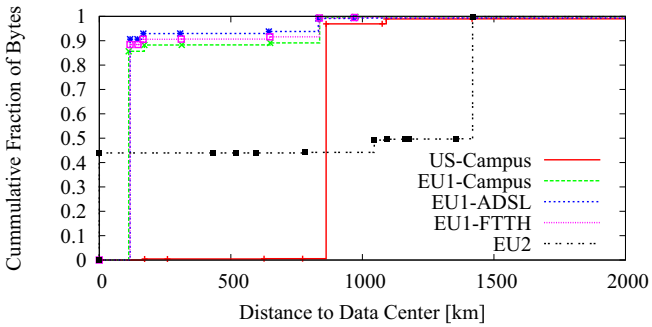


Fig. 8. Fraction of the total YouTube video traffic served by a data center with a distance less than a given value from the dataset collection point.

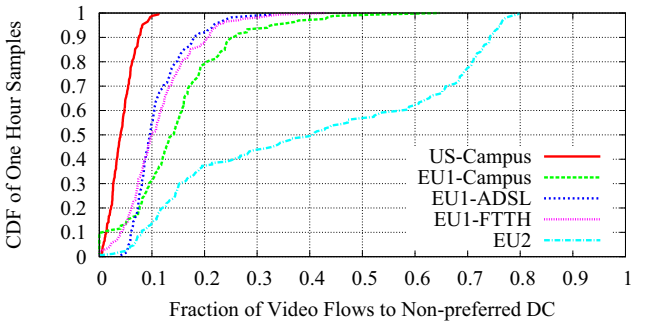


Fig. 9. Variation of the fraction of video flows directed to a non-preferred data center over time. One hour long time periods are considered.

the vantage points and the data centers itself. In particular, we consider the minimum RTT seen by pinging all servers in each data center from the probe PC installed in the PoP. We observe that except for EU2, in each dataset one data center provides more than 85% of the traffic. We refer to this primary data center as the *preferred* data center for that particular trace and other data centers will be labeled as *non-preferred*. At EU2, two data centers provide more than 95% of the data, one of them located inside the ISP and the other outside in the Google AS. We label the data center with the smallest RTT in EU2 as the preferred one. We give a closer look to the EU2 case in section VII-A.

Further, we notice that the data center that provides most of the traffic is also the data center with the smallest RTT for each dataset. This suggests that RTT does play a role in the selection of YouTube servers. However, we have reason to believe that RTT is not the only criteria and that the preferred data center may change over time. For example, in a more recent dataset collected in February 2011, we found that the majority of US-Campus video requests are directed to a data center with an RTT of more than 100 ms and not to the closest data center, which is around 30 ms away.

Figure 8 considers the distance (in kilometers) between users and the data centers they are mapped to. In most cases, the data centers with the smallest delay to the customers are also the physically closest ones. This is not the case for the US-Campus dataset, where the five closest data centers provide less than 2% of all the traffic. Coupled with previous observations about RTT, this is an indication that geographical proximity is not the primary criterion used in mapping user requests to data centers.

The final observation we make is that although most traffic comes from the preferred data center that is typically very close to the customers, there are some exceptions in all datasets. For the US-Campus and the EU1 datasets, between 5% and 15% of the traffic comes from the *non-preferred* data centers. However, in EU2, more than 55% of the traffic comes from *non-preferred* data centers. We now are interested to see the variation over time of the fraction of traffic coming from non-preferred data centers. One hour-long time slots are considered, and the fraction of traffic served by non-preferred data centers in each of these time slots is determined. Figure 9 plots a CDF of these fractions. The results indicate that the fraction varies across time for most datasets, the variation being most prominent for the EU2 dataset. In particular for this dataset, 50% of the samples have more than 40% of the accesses directed to the non-preferred data center.

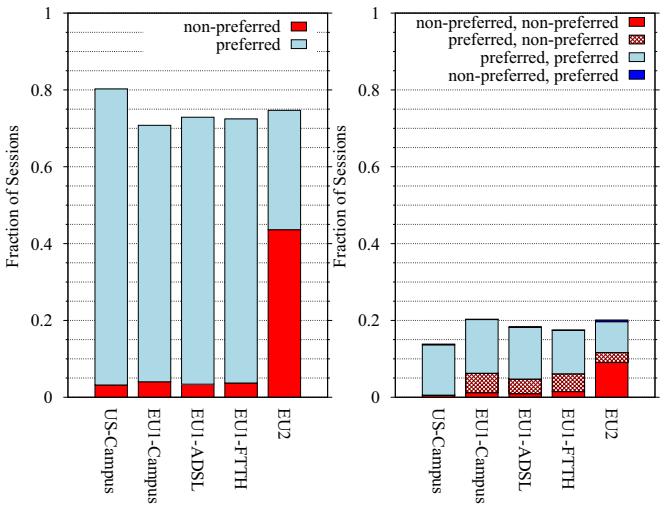
C. Mechanisms resulting in accesses to non-preferred data centers

We have seen that a non-negligible fraction of video flows are downloaded from non-preferred data centers. There are at least two possible causes for this. A first possibility is that the DNS mechanisms direct a request to the non-preferred data center. A second possibility is that the request was redirected to another data center by the preferred data center server.

To disambiguate the two cases, we consider the video session associated with each flow, as discussed in Section VI-A. In the case that DNS mapped a request to a non-preferred data center, the video session must consist of a single video flow to a non-preferred data center, or must begin with a control flow to the non-preferred data center. In the other scenario, the session must begin with a control flow to the preferred data center (indicating the DNS mapping was as expected), but subsequent flows in the session must be to non-preferred data centers.

To better understand the effectiveness of DNS in mapping requests to the preferred data center, consider Figure 10(a). Each bar in the figure shows the fraction of sessions that involve only one flow. Further, each bar shows a break down of the requests sent to the preferred and non-preferred data centers. For instance, for US-Campus, 80% of the sessions involve a single flow; 75% are then served by the preferred data center while 5% of sessions are directly going to the non-preferred data center. Interestingly, about 5% of the single-flow sessions are directly served by the non-preferred data center for EU1 datasets too. For EU2 however, over 40% of the single flow sessions are served by the non-preferred data center. Overall, these results show that DNS is in general effective in mapping requests to the preferred data center. Still DNS mapping mechanisms do account for a significant fraction of video flow accesses to non-preferred data centers.

We next try to understand the extent to which users downloaded video from a non-preferred data center, even though



(a) 1 flow sessions (b) 2 flows sessions

Fig. 10. Breakdown of sessions based on whether flows of the session are sent to preferred data center.

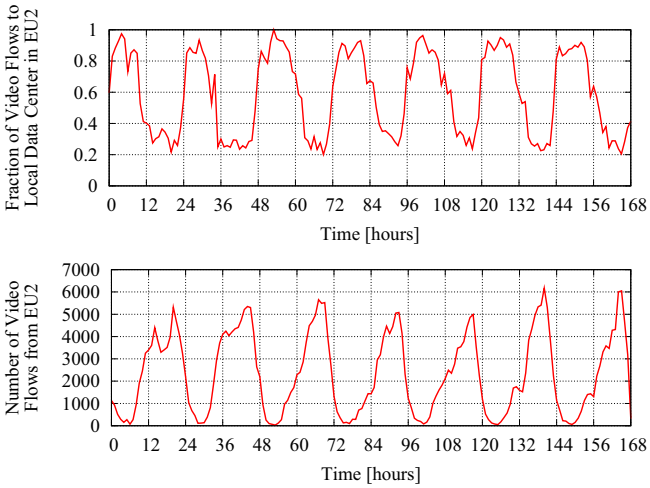


Fig. 11. Fraction of the total YouTube video traffic served by the preferred data center (top graph) and total number of video flows (bottom graph) as a function of time for the EU2 dataset.

they were directed by DNS to the preferred data center. Figure 10(b) presents the breakdown of sessions involving 2 flows. These sessions group a control flow followed by a video flow. Based on whether each flow involves the preferred or non-preferred data center, we have four possible cases: (i) both preferred; (ii) both non-preferred; (iii) the first preferred and the second non-preferred; and (iv) the first non-preferred and the second preferred. Each bar in Figure 10(b) presents the breakdown among these patterns. For all the EU1 datasets, we see a significant fraction of cases where the DNS did map requests to the preferred data center, but application-layer redirection mechanisms resulted in the user receiving video from a server in a non-preferred data center. For the EU2 dataset, we note that a larger fraction of sessions has both flows going to the non-preferred data center, meaning that the DNS is still the primary cause for the user downloading videos from non-preferred data centers. We have also considered sessions with more than 2 flows. They account for $5.18 - 10\%$ of the total number of sessions, and they show similar trends to 2-flow sessions. For instance, for all EU1 datasets, a significant fraction of such sessions involve their first access to the preferred data center, and subsequent accesses to non-preferred data centers. We omit further results for lack of space.

VII. Causes underlying non-preferred data center accesses

In this section, we investigate why accesses to non-preferred data centers occur.

A. DNS-level load balancing

As shown in the previous section, the EU2 dataset exhibits very different behavior compared to other datasets. Over 55% of the video traffic is received from the non-preferred data center, and a vast majority of accesses to non-preferred data centers is due to the DNS mapping mechanisms.

To understand this better, consider Figure 11. The top graph presents the evolution over time of the fraction of video flows served by the preferred data center. One hour time slots are considered. The bottom graph shows the total number of video flows seen in the EU2 dataset as a function of time. Note that time 0 represents 12am on Friday. We can clearly see that there is a day/night pattern in this set of requests. During the night, when the total number of accesses from EU2 is small, the internal data center handles almost 100% of the video requests. However, during the day, when the number of requests per hour goes up to around 6000, the fraction of requests handled by the local data center is always around 30% across the whole week. Results for other datasets are not shown for the sake of brevity. Still, all datasets exhibit a clear day/night pattern in the number of requests. However, there is less variation over time of the fraction of flows served by the preferred data center, as already seen in Fig.9. Furthermore, there is much less correlation with the number of requests.

We believe the reason for this is the unique setup in the EU2 network. In this network, the data center inside the network serves as the preferred data center. While this data center located inside the ISP is the nearest to the users, it is unable to handle the entire load generated by users inside the EU2 ISP during busy periods. There is strong evidence that adaptive DNS-level load balancing mechanisms are in place, which results in a significant number of accesses to the non-preferred data centers during the high load period of traffic.

B. Variations across DNS servers in a network

Our results from the previous section indicate that for the US-Campus dataset most of the accesses to the non-preferred data center are caused by DNS. Deeper investigation indicates that most of these accesses may be attributed to clients from a specific internal subnet within the US-Campus network. Those clients indeed request significantly higher fraction of videos from non-preferred data centers than clients in other subnets. To see this, consider Figure 12. Each set of bars corresponds to

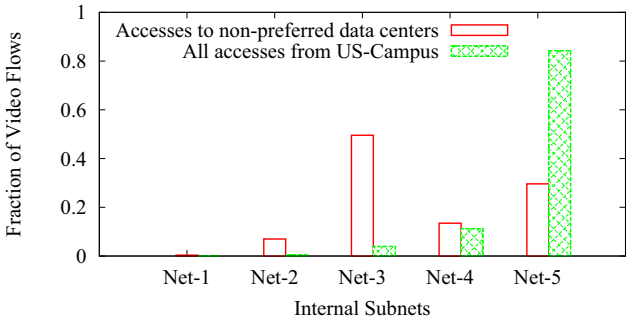


Fig. 12. Fraction of all video flows, and video flows to non-preferred data centers for each internal subnet of the US-Campus dataset.

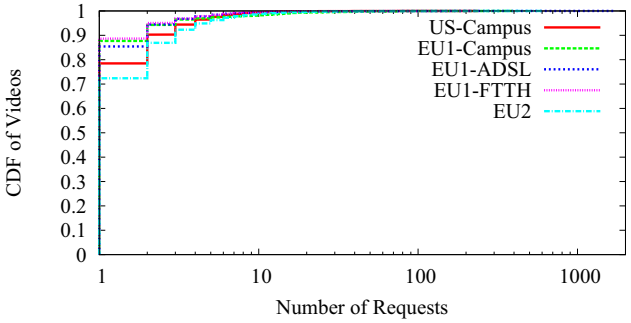


Fig. 13. Number of requests for a video to the non-preferred data centers.

an internal subnet at US-Campus. The bars on the left and right respectively show the fraction of accesses to non-preferred data centers, and the fraction of all accesses, which may be attributed to the subnet. Net-3 shows a clear bias: though this subnet only accounts for around 4% of the total video flows in the dataset, it accounts for almost 50% of all the flows served by non-preferred data centers.

Further investigation shows that hosts in the Net-3 subnet use different DNS servers that map YouTube server names to a different preferred data center. In other words, when the authoritative DNS servers for the YouTube domain are queried by the local DNS servers in Net-3, the mapping provided is to a different preferred data center than the other subnets on US-Campus. We believe this behavior is not a misconfiguration in the YouTube servers or the Net-3 servers, but we rather hypothesize that this is the result of a DNS-level assignment policy employed by YouTube, probably for load balancing purposes, which can vary between DNS servers and thus subnets that belong to the same campus or ISP network.

C. Investigating redirection at the application layer

We now consider cases where users download video from non-preferred data centers, even though DNS mapped them to the preferred data center.

To get more insights into this, consider Figure 13 which reports the CDF of the number of times a video is downloaded from a non-preferred data center. Only videos that are downloaded at least once from a non-preferred data center are considered. The results show two trends. First, a large

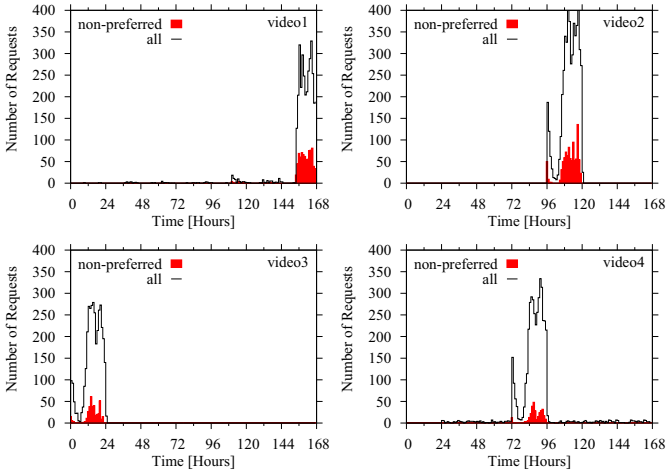


Fig. 14. Load related to the top 4 videos with the highest number of accesses to the non-preferred data centers for the EU1-ADSL dataset.

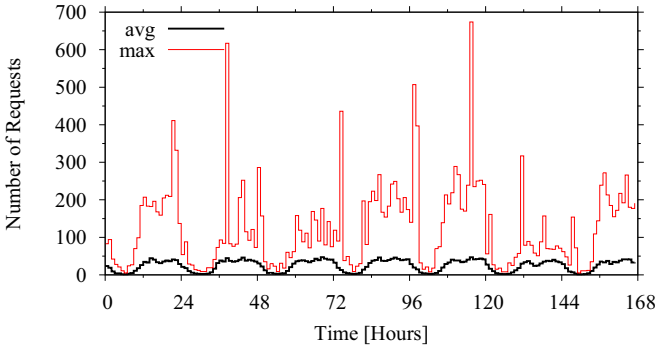


Fig. 15. Average and maximum number of requests per server in the preferred data center of EU1-ADSL dataset.

fraction of videos are downloaded exactly once from the non-preferred data center. For example, for the EU1-Campus dataset, around 85% of the videos are downloaded only once from the non-preferred data center. Second, there is a long tail in the distributions. In fact, some videos are downloaded more than 1000 times from non-preferred data centers. We consider the impact of popular and unpopular videos on server selection in the next few paragraphs.

• **Alleviating hot-spots due to popular videos:** Let us focus first on the tail in Figure 13. Figure 14 considers the four videos with the highest number of accesses to the non-preferred data centers for the EU1-ADSL dataset. Each graph corresponds to one of the videos, and shows (i) the total number of accesses to that video; and (ii) the number of times the video is downloaded from the non-preferred data center, as a function of time. We see that there are spikes indicating that some videos are more popular during certain limited periods of time. Most accesses to non-preferred data centers occur during these periods. In particular, all these videos were played by default when accessing the www.youtube.com web page for exactly 24 hours, i.e., they are the “video of the day”.

Those are therefore very popular videos, which possibly generate a workload that can exceed the preferred data center capacity. Therefore, application-layer redirection is used to

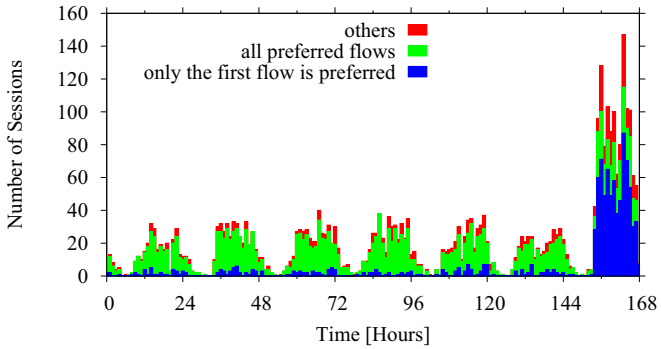


Fig. 16. Number of video sessions per hour seen by the server handling *video1* in the preferred data center of the EU1-ADSL dataset. A breakdown of sessions based on whether flows are directed to preferred data centers is also shown.

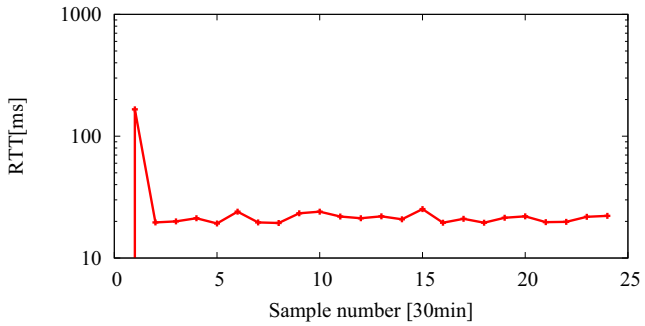


Fig. 17. Variation over time of the RTT between a PlanetLab node and the content servers for requests of the same test video.

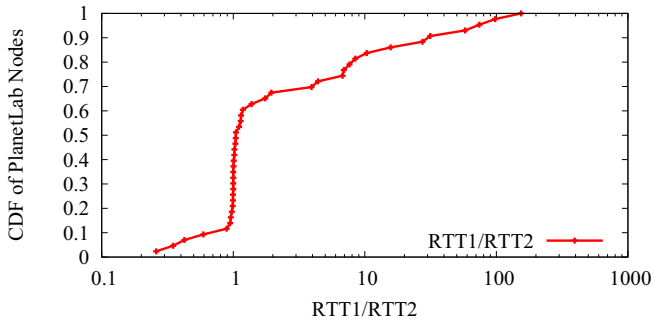


Fig. 18. Reduction in RTT from PlanetLab nodes to the content servers when a test video is downloaded twice. The first access may incur a higher RTT due to unavailability of content in the preferred data center.

handle the peaks. As further evidence, Figure 15 shows the average and the maximum number of requests served by each server (identified by its IP address) in the preferred data center as a function of time. The figure shows that at several times, the maximum number of requests a single server has to handle is by far larger than the average load. For instance at time 115, the average load is about 50 video flows, but there is one server that answers more than 650 requests. Interestingly, we note that the servers suffering the peak loads are those serving the majority of the top videos of Figure 14.

Further investigation reveals that DNS correctly forwards the request to a server in the preferred data center, but since its load is too high, the server redirects part of the requests to another server in a non-preferred data center. Consider Figure 16, which shows the load in terms of sessions, handled by the server receiving the requests for *video1* for the EU1-ADSL dataset. Different colors are used to show the breakdown of the total number of sessions according to the preferred/non-preferred patterns. For example, we can see that in the first 6 days, the majority of the sessions involves only flows served by the preferred data center. On the last day however, a larger number of requests is received, which leads to an increase in application-layer redirections to a non-preferred data center. Overall, these results show that local and possibly persistent overload situations are handled by the YouTube CDN via application-layer redirection mechanisms.

• **Availability of unpopular videos:** Consider again Figure 13. Let us now focus on the observation that several videos are downloaded exactly once from the non-preferred data center. Further analysis indicated that for most datasets, over 99% of these videos were accessed exactly once in the entire dataset, with this access being to non-preferred data centers. However, when the videos were accessed more than once, only the first access was redirected to a non-preferred data center.

This observation leads us to hypothesize that downloads from non-preferred data centers can occur because of the limited popularity of the videos. In particular, videos that are rarely accessed may be unavailable at the preferred data center, causing the user requests to be redirected to non-preferred data centers until the video is found.

Since our datasets only contain a limited view of the accesses seen by a data center, it is difficult to validate this claim using only our datasets. We therefore conducted controlled active experiments using PlanetLab nodes. In particular, we uploaded a test video to YouTube. The video was then downloaded from 45 PlanetLab nodes around the world. Nodes were carefully selected so that most of them had different preferred data centers. From each node, we also measured the RTT to the server being used to download the content. We repeated this experiment every 30 minutes for 12 hours.

Figure 17 shows an example of the variation of RTT samples considering a PlanetLab node located in California. Observe that the very first sample has an RTT of around 200 ms. In contrast, later samples exhibit RTT of about 20 ms. Further investigations showed that the first time, the video was served by a data center in the Netherlands while subsequent requests were served by a data center in California.

Figure 18 shows the CDF of the ratio of the RTT to the server that handled the first video request (RTT1) to the RTT to the server that handled the second video request (RTT2) for all the PlanetLab nodes. A ratio greater than 1 means that the video was obtained from a closer data center in the second attempt than in the first attempt. A ratio with a value close to 1 shows that the first request went to the same server or a server in the same data center as the second request. For over 40% of the PlanetLab nodes, the ratio was larger than 1, and in 20% of the cases the ratio was greater than 10. Interestingly, we have also found the RTT of subsequent samples is comparable to the

RTT of the second sample. Overall, these results indicate that the first access to an unpopular video may indeed be directed to a non-preferred data center, but subsequent accesses are typically handled from the preferred data center.

VIII. Related Work

The attention of the research community on YouTube has grown in the last few years. We can coarsely group works in two categories:

• **YouTube Infrastructure Studies:** Recently, a few works analyzed the YouTube video delivery infrastructure ([7], [8]). Both works focus on the "old" YouTube infrastructure. In [7], the authors collected traces at the backbone of a large ISP. Using DNS name resolution of servers, they discovered eight data centers around the U.S. that provided most videos to clients around the world. Further, they found that the YouTube server selection algorithm does not consider geographical location of clients and that requests are directed to data centers proportionally to the data center size. In contrast, our work focuses on the "new" YouTube infrastructure; we have evidence that requests are now redirected to servers in a preferred data center particular to each network and that RTT between data centers and clients plays a role in the server selection strategy. In [8] the authors perform PlanetLab experiments to download YouTube videos and measure user performance. The authors found that most videos are being sent from a few locations in the U.S. and that YouTube pushes popular videos to more data centers. In contrast, in our work, we study traces from several large ISP and campus networks in two continents, which capture actual user behavior; we found that most videos are being delivered from a preferred data center, typically close to the client and that, while popularity of videos may play a role on the redirection of clients to non-preferred data centers, it is not a prominent reason for it. Finally, we also differ from [7], [8] in that we analyze key factors that affect server selection in the YouTube CDN. More recently, a concurrent and preliminary work [15] has started analyzing the new YouTube infrastructure. Our work clearly differs in various aspects. In particular, we use more systematic state-of-the-art algorithms for server geolocation; we also rely on a trace-based analysis instead of active PlanetLab experiments and finally we dig deeper into identifying the various causes underlying content server redirection.

• **YouTube Videos Characterization:** Several works have focused on characterizing various aspects of videos existing in YouTube as well as usage patterns. [3] and [4] collected traces at the edge of a single campus network and characterized per video statistics such as popularity, duration, file size and playback bitrate, as well as usage pattern statistics such as day versus night accesses and volume of traffic seen from the Campus. [5] and [6] crawled the YouTube site for an extended period of time and performed video popularity and user behavior analysis. Further, [5] compares YouTube to other video providers such as Netflix and [6] investigates social networking in YouTube videos. We differ from all these works since we study the video distribution infrastructure. In particular we focus on understanding the content server selection mechanisms used by YouTube. In addition, we analyze datasets from five distinct vantage points ranging from campus networks to nationwide ISPs.

IX. Conclusion

In this paper we have obtained a deeper understanding into the factors impacting how YouTube video requests are served by data centers. Our understanding has been based on week-long datasets collected from the edge of five networks including two university campuses and two national ISPs, located in three different countries. Our analysis indicates that the YouTube infrastructure has been completely redesigned compared to the one previously analyzed in the literature. In the new design, most YouTube requests are directed to a preferred data center and the RTT between users and data centers plays a role in the video server selection process. More surprisingly, however, our analysis also indicates a significant number of instances (at least 10% in all our datasets) where videos are served from non-preferred data centers. We identified a variety of causes underlying accesses to non-preferred data centers including: (i) load balancing; (ii) variations across DNS servers within a network; (iii) alleviation of load hot spots due to popular video content; and (iv) availability of unpopular video content in a given data center. Overall these results point to the complexity of factors that govern server selection in the YouTube CDN.

References

[1] C. Labovitz, S. Iekel-Johnson, D. McPherson, J. Oberheide, and F. Jahanian, "Internet Inter-domain Traffic," in *ACM SIGCOMM*, 2010.

[2] "YouTube Fact Sheet," http://www.youtube.com/t/fact_sheet.

[3] P. Gill, M. Arlitt, Z. Li, and A. Mahanti, "YouTube Traffic Characterization: A View From The Edge," in *ACM IMC*, 2007.

[4] M. Zink, K. Suh, Y. Gu, and J. Kurose, "Characteristics of YouTube Network Traffic at a Campus Network - Measurements, Models, and Implications," *Computer Networks*, vol. 53, no. 4, pp. 501–514, 2009.

[5] M. Cha, H. Kwak, P. Rodriguez, Y.-Y. Ahn, and S. Moon, "I Tube, You Tube, Everybody Tubes: Analyzing The World's Largest User Generated Content Video System," in *ACM IMC*, 2007.

[6] X. Cheng, C. Dale, and J. Liu, "Statistics and Social Network of YouTube Videos," in *IWQoS*, 2008, pp. 229–238.

[7] V. K. Adhikari, S. Jain, and Z.-L. Zhang, "YouTube Traffic Dynamics and Its Interplay With a Tier-1 ISP: an ISP Perspective," in *ACM IMC*, 2010.

[8] M. Saxena, U. Sharan, and S. Fahmy, "Analyzing Video Services in Web 2.0: a Global Perspective," in *NOSSDAV*, 2008.

[9] "Maxmind GeoLite City Database," http://www.maxmind.com/app/geolitecity.

[10] B. Gueye, A. Ziviani, M. Crovella, and S. Fdida, "Constraint-based Geolocation of Internet Hosts," *IEEE/ACM Transactions on Networking*, vol. 14, no. 6, pp. 1219–1232, 2006.

[11] M. Mellia, R. L. Cigno, and F. Neri, "Measuring IP and TCP Behavior on Edge Nodes with Tstat," *Computer Networks*, vol. 47, no. 1, pp. 1–21, 2005.

[12] M. Pietrzyk, J.-L. Costeux, G. Urvoy-Keller, and T. En-Najjary, "Challenging Statistical Classification for Operational Usage: the ADSL Case," in *ACM IMC*, 2009.

[13] "Tstat home page," http://tstat.polito.it.

[14] R. Percacci and A. Vespignani, "Scale-free behavior of the internet global performance," *Eur. Phys. J. B*, vol. 32, no. 4, pp. 411–414, 2003.

[15] V. K. Adhikari, S. Jain, G. Ranjan, and Z.-L. Zhang, "Understanding data-center driven content distribution," in *ACM Conext Student Workshop*, 2010.