Pytomo: A Tool for Analyzing Playback Quality of YouTube Videos

Parikshit Juluri*†, Louis Plissonneau†, and Deep Medhi*
*{pjuluri,dmedhi}@umkc.edu – CSEE – University of Missouri – Kansas City, USA
†louis.plissonneau@orange-ftgroup.com – Orange Labs, France

***Abstract*—Online video services account for a major part of broadband traffic with streaming videos being one of the most popular video services. We focus on the user perceived quality of YouTube videos as it can serve as a general index for customer satisfaction. Our tool, `Pytomo` [1], is a tomography tool that is designed to measure the playback quality of videos as if they are being viewed by a user. We model the YouTube video player to estimate the playback interruptions as experienced by a user watching a YouTube video. We also examine topology and download statistics such as delay towards the server, download rates, and buffering duration.**

We aim to analyze different DNS resolvers to obtain the IP address of the video server. We study how the DNS resolution impacts the performance of the video download, and thus, the video playback quality. As the tool is intended to run on multiple ISPs, we have discovered some interesting results in YouTube distribution policies. These results can be applied to any content-delivery networks (CDN) architecture and should help users to better understand what are the key performance factors of video streaming.

***Index Terms*—HTTP Streaming, Performance, QoE, DNS.**

I. Introduction

Currently, web-driven contents represent about half of the Internet traffic due to the decrease of P2P and the surge of video sharing sites [2], [3], [4], with YouTube being the most popular. Among the different online video services, streaming videos and flash videos are the most popular ones. Services such as blogs and social networks are also enabling users to embed personal videos, and thus, expand video sharing circles.

In this paper, we present our tool, `Pytomo`, to analyze the user experience while watching a YouTube video by using active download analysis. Most of the previous work so far has usually studied either the characterization of YouTube videos, or the YouTube CDN architecture.

YouTube videos characteristics' analyses mainly focus on meta-data. Each crawler fetches the properties of the video (*duration, category*. . .) to draw interesting results on caching and distribution policy evaluations [5], on comparison with "classical" web workloads [6], or on graph relations between videos [7]. Some authors complement their study with passive packet captures to analyse the streaming video sessions [8], or the behavior of users in terms of switches and jumps inside videos [9]. Previous studies provide us information about the video characteristics; however, they do not analyze the video playback quality.

The YouTube CDN architecture has also been studied in [10] with NetFlow records in order to determine traffic dynamics outside the ISP network. The YouTube server selection policy can be explained either using active measurements (on PlanetLab nodes) [11], [12], or using passive captures [13]. These measures allow us to better understand the distribution choices of YouTube videos, but they mainly focus on delay and geographical distribution of servers. In [13], the server strategy selection is also evaluated. The impact of DNS resolvers have been compared in terms of latency and caching [14].

Our work differentiates from others since we are interested not only in the delay to access the YouTube video streaming servers, but also in the perceived video playback quality. Moreover, the impact of the DNS resolver on the video playback quality has not been studied yet.

We explain the methodology and the evaluation of our tool, `Pytomo`, in Section II. We briefly present preliminary results in Section III, and state the next steps of our work in Section IV.

II. Methodology

In this section, we present the methodology behind the tool. `Pytomo` [1] is a platform independent open-source automated analysis tool written in Python. The aim is to measure the download and to analyze the playback of YouTube videos, and thus, emulate the user's watching experience. Interruptions and buffering of online streaming videos occur when the download throughput is lower than the encoding rate of the video. Thus, we choose the number of interruptions during the playback and the total buffering duration as the main *playback quality* indicators; see Section II-B on how we infer such events.

A. Tool Description

`Pytomo` performs an analysis of YouTube video downloads and helps us evaluate user experience. `Pytomo` emulates the user behavior by downloading a YouTube video, and then selects a number of random related links for downloading.

For each video, the download statistics are collected, calculated, and stored in a database. In order to perform the download, we first resolve the IP address of the content server and then use this IP address to perform the analysis. By doing so, we ensure that the analysis and video download are being done on the same server. We also take care of HTTP Redirect messages obtained from video servers.

a) DNS Resolution: It is possible to use a number of DNS resolvers. In our case, we use three DNS resolvers: a default ISP resolver, Google's Public DNS (`code.google.`

978-0-9836283-0-9 © 2011 ITC

This paper was peer reviewed by subject matter experts for publication in the Proceedings of ITC 2011

com/speed/public-dns/docs/intro.html), and the Open DNS (www.opendns.com) resolver.

Our tool allows us to evaluate the *playback quality* of the Youtube cache servers according to the IP address resolved by different DNS resolvers. Thus our primary interest is the download throughput achieved from this server (IP address) and not the delay towards the video server, which is the classical evaluation method. Moreover, we take a novel approach by relating this selection to the requesting DNS.

b) Crawl: The process of video crawling can be summarized as follows: *(i)* We begin by selecting an initial set of YouTube videos that are used as the seed (by default the most popular videos of the week); *(ii)* For each selected video, we obtain the cache-URL of the video server hosting the video file[1]; *(iii)* Next we obtain the (possibly different) IP addresses of the video server by querying the three different DNS servers; *(iv)* We collect the ping statistics for each resolved IP address of the video servers; *(v)* We download from each resolved IP address a sample of the video for 30 seconds with the default video format (640×390); *(vi)* We continue crawling from step *(ii)* with the next related video.

c) Statistics: Our tool collects the following statistics: ping statistics; video information; download statistics; playback statistics such as initial buffer, interruptions, total buffering duration, buffer duration at the end of the download (see [1] for a detailed description).

B. Tool Evaluation

One of the main aspects of our work is the emulation of the YouTube video player to evaluate end-user perceived quality of video playbacks. We model the video player with an *initial buffer* used before the video starts; the initial buffer is obtained by parsing the FLV tags of the video to determine the precise amount of bytes needed to start playing the video, we then maintain two timescales: one for the watched video and another for the downloaded video. As soon as the difference between the two time metrics is lower than the *minimal playback buffer*, the video stops. Once the downloaded video time is larger than the *minimal restart buffer*, the video playback resumes.

To evaluate and calibrate the parameters of our model, we use a local web server on a test machine so that we can control the entire video distribution. To do this, we watch a video with the YouTube Shockwave Flash (swf) player and intercept the video requests with a dedicated proxy[2] so that it is served by our monitored server instead of the YouTube video servers.

This setup allows us to determine the main parameters used by the player to switch from the buffer state to the playing state and vice-versa. Indeed, we manually record each start/stop event on the playback to infer the characteristics of the YouTube swf player.

[1] YouTube has two types of servers: YouTube front-end web servers that host the video web pages and the video servers that host the actual videos [10].

[2] Because of domain security parameters in the YouTube swf player, we cannot directly query the video on localhost.

III. Preliminary Results

We have collected preliminary data by contacting volunteers to run the tool on their private Internet access in order to obtain a good representation of a real user experience. Due to lack of space, we briefly report some of the preliminary findings.

First, the IP addresses of the YouTube video servers are usually different depending on the DNS resolver; these IPs can also belong to different ASes (in this case, AS 15169 for Google and AS 43515 for YouTube EU). However, there are cases when the three DNS resolvers return the same IP address.

Secondly, the delay towards the IP addresses can have huge variations (up to twice the time). The *closest* servers are not always coming from the same DNS resolver[3], and we have not yet explored the YouTube video server selection strategy.

Lastly, the download and playback performance has a large discrepancy depending on the resolver and the ISP. The most interesting result is that with the same Internet access, the same YouTube video can have a very different playback quality depending on the DNS resolver.

IV. Future Work

The main focus of this work has been to construct a reliable tool to automatically evaluate the playback quality of YouTube videos as experienced by users. Our next objective is to collect extensive data so that we can obtain a representative evaluation of the YouTube video streaming performance key factors, and its relation to the DNS used on the client side.

References

[1] Pytomo, "http://code.google.com/p/pytomo/."

[2] G. Maier, A. Feldmann, V. Paxson, and M. Allman, "On Dominant Characteristics of Residential Broadband Internet Traffic," in *IMC*, 2009.

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

[4] M. Pietrzyk, L. Plissonneau, G. Urvoy-Keller, and T. En-Najjary, "On Profiling Residential Customers," in *International Workshop on Traffic Monitoring and Analysis (TMA)*, 2011.

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

[6] P. Gill, M. F. Arlitt, Z. Li, and A. Mahanti, "Youtube Traffic Characterization: A View From the Edge," in *IMC*, 2007.

[7] X. Cheng, C. Dale, and J. Liu, "Understanding the Characteristics of Internet Short Video Sharing: YouTube as a Case Study," in *IMC*, 2007.

[8] P. Gill, M. Arlitt, Z. Li, and A. Mahanti, "Characterizing User Sessions on YouTube," in *ACM/SPIE MMCN*, 2008.

[9] L. Plissonneau, T. En-Najjary, and G. Urvoy-Keller, "Revisiting Web Traffic from a DSL Provider Perspective: the Case of YouTube," in *ITC SS on Network Usage and Traffic*, 2008.

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

[11] V. K. Adhikari, S. Jain, and Z. L. Zhang, "Where Do You "Tube"? Uncovering YouTube Server Selection Strategy," in *IEEE ICCCN*, 2011.

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

[13] R. Torres, A. Finamore, J. Kim, M. Mellia, M. Munafo, and S. Rao, "Dissecting Video Server Selection Strategies in the YouTube CDN," in *IEEE ICDCS*, 2011.

[14] B. Ager, W. Mühlbauer, G. Smaragdakis, and S. Uhlig, "Comparing DNS resolvers in the wild," in *IMC*, 2010.

[3] As YouTube is owned by Google, we could expect a resolution towards "best" video servers on Google Public DNS, but this was not always the case.