

# Pytomo: YouTube Crawler

Louis Plissonneau

Orange Labs

May 2011



# 1 - Introduction

## Outline

**Goal:** Collecting statistics on the actual video download from YouTube

### Context

What to measure?

How to measure?

### Contributing

Why helping?

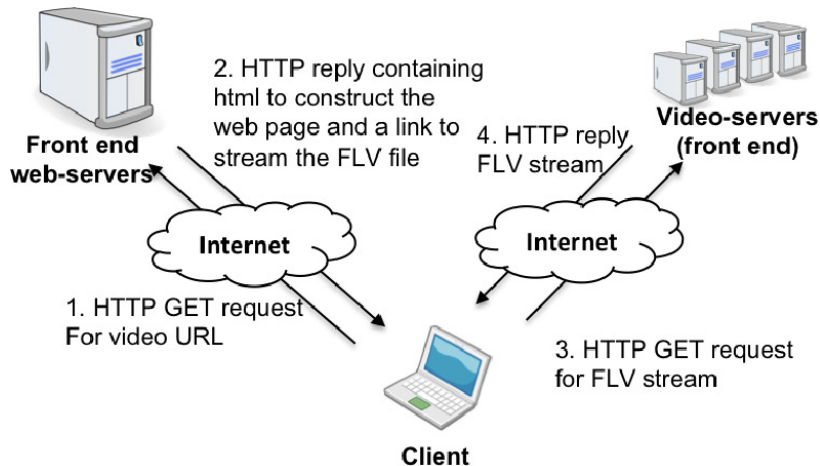
How to help?

Example of results

## 2 - Context



## 2 - Context



**Figure:** Explanation of YouTube video retrieval (from “YouTube Traffic Dynamics and Its Interplay with a Tier-1 ISP: An ISP Perspective”, V. K. Adhikari, S. Jain and Z. Zhang IMC 2010)

## 2 - Context

What to measure?

### Video Download Statistics

- ▶ Not global video statistics
  - ▶ Already done
  - ▶ Not indicating download performance  $\Rightarrow$  user experience
- ▶ Really downloads videos from YouTube: 30s at the beginning of the video
- ▶ Links collection:
  - ▶ Start with the most popular videos of the week (configurable to month, all time. . .)
  - ▶ Use a random number of related videos for next links

## 2 - Context

### What to measure?

### DNS load-balancing

- ▶ Common technique to send requests to different data-centers/servers
- ▶ Done through DNS resolution

Use different DNS in order to map the performance to the resolver used:

- ▶ ISP default DNS
- ▶ Google Public DNS
- ▶ Open DNS

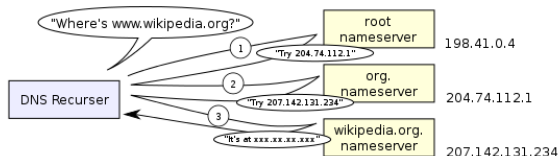


Figure: DNS Recursive Resolution (from Wikipedia)

## 2 - Context

### How to measure?

#### Metric used

- ▶ Ping times: approximation of the distance to the data-center
- ▶ HTTP redirection
- ▶ Download stats:
  - ▶ Downloaded duration
  - ▶ Downloaded bytes
  - ▶ Max instant throughput
  - ▶ Buffer duration at the end of download
  - ▶ Number of interruptions in playback
- ▶ Additional information: video duration, video size, encoding rate

## 2 - Context

How to measure?

### Modeling of video playback

- ▶ Out of average encoding rate and bytes received by the application<sup>1</sup>  $\Rightarrow$  evaluation of *downloaded video time*
- ▶ Count of *current video time* after a buffering period at the start of video (default 3 seconds)
- ▶ Each time the current video time reaches the downloaded time, we count an interruption

---

<sup>1</sup>no data is stored locally

# 3 - Contributing



# 3 - Contributing

## Why helping?

Many measures are needed

- ▶ Need of **multiple points of measure** so that we have a more global view of the performance
- ▶ Need of **multiple ISPs** to figure out different distribution policies according to ISP

This will help all ISPs to understand how to get better user performance on YouTube

## 3 - Contributing

### How to help?

#### Run the crawler: Pytomo

- ▶ Open source crawler written completely in Python with no external dependencies
- ▶ Works on Linux, Windows (standalone executable), Mac

#### Websites:

- ▶ <http://code.google.com/p/pytomo/> (download of all version and browse the code)
- ▶ <http://pypi.python.org/pypi/Pytomo> (Python Package Index)

# 3 - Contributing

## How to help?

Send the results (database and log files):

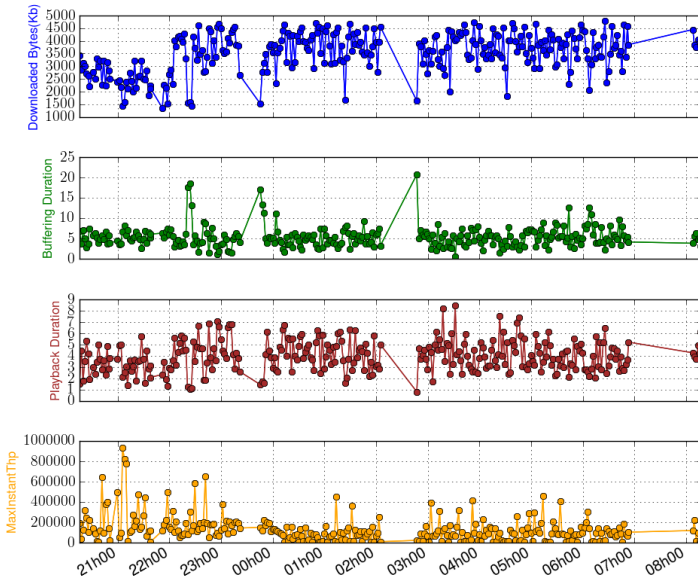
- ▶ to me: `louis.plissonneau@gmail.com`
- ▶ to Ernst Biersack: `erbi@eurecom.fr`

## Notes

- ▶ You can run it multiple times (it will give more data to analyse)
- ▶ Do NOT run multiple instances in parallel!
- ▶ You can let it run all night long (all week long...)
- ▶ Don't hesitate to advertise it (friends, family...)

# 3 - Contributing

## Example of results



## 4 - Conclusion

Your help is needed:

to **improve YouTube performance**, we need measures

You can easily run this tool to allow ISPs to figure out how to manage YouTube traffic.

# Pytomo: YouTube Crawler

Louis Plissonneau

Thank you!

