Network tomography from an operator perspective

Louis Plissonneau

ParisTech
INSTITUT DES SCIENCES ET TECHNOLOGIES
PARIS INSTITUTE OF TECHNOLOGY

TELECOM ParisTech

2012-ENST-033

EDITE - ED 130

Doctorat ParisTech

T H È S E

pour obtenir le grade de docteur délivré par

TELECOM ParisTech

Spécialité “INFORMATIQUE et RESEAUX”

présentée et soutenue publiquement par

Louis PLISSONNEAU

le 9 juillet 2012

Network Tomography
from an Operator Perspective

THÈSE

Directeur de thèse: **Prof. Ernst BIERSACK**

Jury

M. Chadi BARAKAT, INRIA, Sophia Antipolis - France	Rapporteur
M. Refik MOLVA, EURECOM, Sophia Antipolis - France	Examinateur et Président
M. Matti Mikael SIEKKINEN, AALTO University, Aalto - Finlande	Examinateur
M. Guillaume URVOY-KELLER, Laboratoire I3S, Sophia Antipolis - France	Rapporteur

TELECOM ParisTech
école de l’Institut Télécom - membre de ParisTech

Abstract

Network tomography is the study of a network's traffic characteristics using measures. This subject has already been addressed by a whole community of researchers, especially to answer the need for knowledge of residential Internet traffic that ISPs have to carry. One of the main aspects of the Internet is that it evolves very quickly, so that there is a never ending need for Internet measurements. In this work, we address the issue of residential Internet measure from two different perspectives: passive measurements and active measurements.

In the first part of this thesis, we passively collect and analyse statistics of residential users' connections spanning over a whole week. We use this data to update and deepen our knowledge of Internet residential traffic. Then, we use clustering methods to form groups of users according to the application they use. This shows how the vast majority of customers are now using the Internet mainly for Web browsing and watching video Streaming. This data is also used to evaluate new opportunities for managing the traffic of a local ADSL platform. As the main part of the traffic is video streaming, we use multiple snapshots of packet captures of this traffic over a period of many years to accurately understand its evolution. Moreover we analyse and correlate its performance, defined out of quality of service indicators, to the behavior of the users of this service.

In the second part of this thesis, we take advantage of this knowledge to design a new tool for actively probing the quality of experience of video streaming sites. We have modeled the playback of streaming videos so that we are able to figure out its quality as perceived by the users. With this tool, we can understand the impact of the video server selection and the DNS servers on the user's perception of the video quality. Moreover the ability to perform the experiments on different ISPs allows us to further dig into the delivery policies of video streaming sites.

Résumé

Le domaine de la mesure des caractéristiques du trafic transitant sur un réseau a été largement traité par une vaste communauté de chercheurs, en premier lieu pour répondre aux attentes des opérateurs fournisseurs d'accès à Internet. En effet, leur première préoccupation est de savoir quel type de trafic ils doivent transporter. Une des principales caractéristiques de l'Internet est qu'il évolue très vite, de sorte que le besoin de mesures du trafic grand public ne se tarit jamais. Dans ce travail, nous abordons la question de la mesure du trafic Internet grand public par deux perspectives différentes : les mesures passives et les mesures actives.

Dans la première partie de cette thèse, nous capturons et analysons passivement les statistiques des connections d'utilisateurs d'Internet durant plus d'une semaine. Nous utilisons ces données pour réviser et approfondir notre connaissance du trafic Internet résidentiel. Ensuite, nous utilisons des méthodes de regroupement pour créer des ensembles d'utilisateurs en fonctions des applications qu'ils utilisent. Nous apprenons donc qu'une vaste majorité des clients se connectent à Internet principalement pour surfer sur le Web et regarder des vidéos en *streaming*. Ces données nous servent aussi à évaluer de nouvelles possibilités de contrôler le trafic d'une plateforme ADSL. Comme la principale partie du trafic provient du vidéo *streaming*, nous prenons plusieurs instantanés de ce trafic avec des captures paquet durant une période de plusieurs années, ceci pour comprendre précisément l'évolution de ce trafic. De plus, nous analysons et relions la performance du vidéo *streaming*, définie par des indicateurs de qualité de service, au comportement des utilisateurs de ce service.

Dans la deuxième partie de cette thèse, nous tirons parti de cette connaissance pour concevoir une sonde active capable de mesurer la qualité d'expérience des sites de vidéo *streaming*. Nous avons modélisé la lecture des vidéos *streaming* pour pouvoir déterminer leur qualité telle qu'elle est perçue par les utilisateurs. Grâce à cet outil, nous pouvons comprendre l'impact de la sélection du serveur vidéo et du serveur DNS sur la perception de la qualité vidéo par l'utilisateur. De plus, la possibilité de réaliser des mesures depuis divers opérateurs, nous permet de détailler les politiques de distribution vidéo utilisées par les sites de *streaming*.

Acknowledgements

Tuez-les tous, Dieu reconnaîtra les siens.[1]

Arnaud Amaury

The first person PhD students traditionally thank is their supervisor. I have always thought the main reason to do this is that he is the one to authorise you to defend your thesis. At least in my case, the main reason is not this one but the gratitude for giving me[2] good scientific habits, namely *explore every bit of data* and *plot readable graphs*. With these two mottoes in mind, any research work becomes much easier. So I would like to thank Ernst for accepting taking me as his student.

In the specific case of a part-time PhD, the person giving you the possibility to undertake these studies comes next in the acknowledgements. Here again, this is not the reason why I would like to thank Jean-Pierre Paris. It's because his exemplary nature and his technical expertise are the North Star I've been following.

All the managers I've had during this long lasting PhD have helped and motivated me a lot. Here, I would like to specially mention Jean-Laurent Costeux.

A lot of people have contributed directly or indirectly to this thesis. I would like to thank Marcin Pietrzyk, Guillaume Vu-Brugier, Taoufik En-Najjary, Mickaël Meulle, Parikshit Juluri, Guillaume Urvoy-Keller, Deep Medhi, Stevens Le Blond, Moritz Steiner, Simon Leinen and all the other contributors who have worked with me during this PhD. I also would like to thank all my colleagues in Orange who have helped me building a strong and useful (or pragmatic) technical knowledge in the domain of networking. Even though I was not there often, I would like to thank the staff in Eurécom, especially Gwenaëlle Le-Stir, for making the administrative tasks transparent and painless.

As usual, the last word goes to the family. My family has been very supportive, motivating and understanding. I would say that hearing your children say *"So finally, you are really going to finish your thesis!"* is the greatest scientific achievement ever.

[1] *Kill them all, God will know his own.* – my translation

[2] more precisely: forcefully instilling

Contents

List of Figures

List of Tables

CHAPTER 1

Introduction

Dessine-moi un mouton ![1]

Antoine de Saint-Exupéry,
Le Petit Prince

1.1 Network Measurement: Tomography

Internet measurement can be undertaken at different levels: from an end-user computer to a router of the core network. The amount of data collected is thus a trade-off between the precision and the storage (or analysis) capacity of the system. A very coarse view of a system can be given by the count of the total amount of bytes or packets transiting through a network interface, this is a typical setup for routers transmitting Giga-Bytes of traffic per second. The most precise measurement is packet level trace and is usually captured through dedicated software. The data measurement setup should not be determined by the capacity of the probe but by the precision of analysis required.

The methodology of capture is as important as the data collection: actively requesting a server *vs.* passively duplicating Internet packets are two completely different methods that do not share the same objectives. Active probing can be used to measure how a service is accessed or what is its performance on a specific setup. On the contrary, passive measurements are usually taken at a much larger scale, but at the cost of losing the ability to customize the requests. Passive captures are used to understand what is actually happening on the monitored network.

The purpose of network tomography is not only to collect data, but to understand it. Usually, it imply evaluating the performance of an Internet connection. Internet performance can have many definitions depending on the point of view:

- at router scale, the drop rate of packets (independently of the connection) is the main indicator;

[1] *Draw me a sheep* – my translation

- on a transit link, the load of the link is of primary interest;
- for an ISP, the global load of a local platform determines not only the satisfaction of its customers, but also the need of upgrading the hardware;
- for a Web user, the delay encountered while accessing it favorite website is the only satisfaction measure;
- whereas a P2P user shall be mainly interested in the total throughput achieved for its file transfers;
- finally, a TCP expert can define the performance of a connection as the ratio of desequenced packets without retransmission only during bulk transfer periods.

Here again the definition of performance should be taken according to the goal of the analysis and not to some pre-computed available metrics.

1.2 ISP Motivation

Internet Service Provider provide a so-called *best-effort* service: their first goal is to transmit packets between their customers and other machines on the Internet. Many factors have an impact on the customers connections:

- the access network capacity (and also collection infrastructure: ATM *vs.* GE);
- the ISP network from the access collection point towards the destination of the connection;
- the link capacity between the ISP and the next AS towards the destination;
- the routing policies between all the ASes through which the packets will transit until the destination.

Only some of these factors can be controlled by ISPs. Nevertheless, the main protocol used to transmit packets over the Internet is TCP which is an end-to-end protocol. This means that the packet analysis of a connection (at any point of measure) can give useful information on the path capacity and the resulting performance from the end-user point of view.

The motivation of an ISP is to give the best performance to its customers at a given cost (both for its own infrastructure and the peering agreements with other ASes). The use of different methods of measurement and analysis, as presented in this thesis, is thus of primary interest for ISPs. This can lead to new ways of managing the traffic ranging from local platform load management to TCP configuration according to the service accessed.

1.3 Organisation of the Thesis

The two main parts of the thesis are based on the choice of the measurement method: passive in Part I *vs.* active in Part II.

We first give the context and related work on passive measuring the Internet in Chap. 2. The passive measurement studies in Part I benefits from data collected from many different sources and at very different scale. For Chap. 3, we have analysed connection statistics over more than a week for 3 local ADSL platforms. This gives us many insights on the applications used[2] and the performance of 4,000 different users. We also use this information to evaluate innovative ways of managing a local platform. In Chap. 4, we use multiple packet level traces of all users of a local ADSL platform during short time spans (1 hour) to focus on HTTP Streaming performance. This data has been collected over a period of three years. We show how the network conditions influence the behavior of users watching streaming videos. We conclude Part I in Chap. 5.

The challenges and related work on how to actively measure the Internet are given in Chap. 6. The active measurements presented in Chap. 7 have been collected by a new tool measuring the quality of experience of YouTube videos. We have collected data from many volunteers around the world, and also analysed data from a laboratory connected to the Internet through multiple ISPs. From this data, we figure out the main causes of perturbations of the end-user perceived quality. The main message is that link cost and ISP dependent policies have much more impact on the quality than usual quality of service metrics. Moreover, the access capacity on ADSL (and even more on FTTH) is no more a bottleneck in the access of video streaming service. We conclude Part II in Chap. 8.

Finally a conclusion of the thesis is given in Chap. 9.

[2] and which applications are used in parallel

Part I

Passive Measurements

CHAPTER 2

Passive Measurements Context and Methods

We don't see things as they are, we see them as we are.

Anaïs Nin

Monitoring what is happening in a network (or on a link) without perturbing the traffic is of primary interest for an ISP. It gives the opportunity to understand and plan the development of the traffic transiting through its network. Nevertheless, if an operator would like to have a complete view of all its customers, the amount of data to capture can quickly become huge. Thus methods are used to reduce the data necessary to fit in the evolution of traffic. In this part, we focus on data collected from an ISP at a local platform level. We shall study in detail connection level statistics over a timescale of a week in Chap. 3. This will give us an updated view of the actual traffic generated by residential users. We shall learn how streaming traffic (and especially video clips) is nowadays the main application in terms of downstream volume. In Chap. 4, we use packet level traces during short timescale to precisely measure what is the performance of video streaming traffic and to determine the impact of the quality of service on the usage of video streaming.

In this chapter, we first recall main methods and tools used to passively capture Internet traffic in Sect. 2.1. In Sect. 2.2, we expose a summary of the main results of Part I and position our work in the passive measurements area. Finally, in Sect. 2.3, we review relevant related work focusing particularly on video streaming as it is one of the main focus of this thesis.

2.1 Methods and Tools for Passive Measurements

The first method to monitor Internet traffic is to collect statistics on the border routers of the entity that would like to monitor its network. This is usually done through inquiry on SNMP counters of routers or switches. These measures are used to get a broad view

and no precise information can be expected from this data. Indeed, precise evaluation of the Internet traffic needs the concept of a **connection** (also called **flow**). Traffic monitoring literature defines a connection as an aggregation of packets identified by the same **five tuple** consisting of *source and destination IP addresses, IP protocol, source and destination port numbers (for TCP and UDP only)*.

The next step towards precise measure of the traffic transiting through large routers is arguably Netflow records. Even though Netflow is originally a Cisco product, it is now recognised as a standard for traffic monitoring. Netflow records identify a connection as the standard five tuple plus the ingress interface and the IP type of service: this leads to unidirectional connections. Large routers give processing priority to the packet routing over the collection of Netflow records, thus random packet sampling was introduced in the collection process: only 1 packet out of n (usually $n = 1000$) is recorded. The accuracy and implication of this method has been studied in a large number of studies [31, 16, 11, 21, 7]... The main impact of this sampling is that it gives more importance to large flows (more likely to be caught by sampling) over small ones [12]. The large connections have been called elephants and small ones mice [6], and a whole taxonomy of Internet flows has thus been derived [9, 5].

To overcome the limitation of packet sampling, many researchers have developed dedicated probe to capture the Internet traffic The most popular packet capture softwares are tcpdump [52] and Wireshark [56]. They are based on the libpcap [32] C library, which is also the basis of many dedicated capture softwares (such as those developed by ISPs). Even if in this thesis our passive capture tool is a private one, many other good software to passively capture Internet packet are freely available [54, 50]. Dedicated hardware (such as Endace Dag cards [17]) can be used to cope with large amount of packets arriving at an interface, they also improve the precision of packet timestamping (which is useful to compute accurate performance indicators).

Once the question of how to capture has been resolved, the next question is: “Where to capture?” This is a crucial issue as the results drawn out of the measurements will highly depend on it. Many measurement studies are based on PlanetLab [38] or on Universities campuses (mainly in the US.). Even if modeling Internet connections can be done through this kind of measures, the lack of some applications (*e.g.* P2P, enterprise specific...) induce a large bias in the results. For ISPs, it’s even more important to have data from residential customers, and this data has to come from a similar country (from geographic and linguistic point of view) to be transposable.

The last question to address is: “What to capture?” Here again a trade-off has to be chosen between capturing more data for a shorter period of time, or having a long term analysis but reducing the scale of data captured.

Once we have collected data, the question of privacy arise. In the case of ISPs, we do not want (nor have the right to) divulge the contents of the packets transmitted. Nevertheless, if we take the analogy between a packet and a post letter, we focus only on the details given on the envelop (including TCP sequence number if needed) and not on the contents (the payload of a data packet). In the same problematic, anonymisation concerns have to be taken into account.

2.2 Contributions

In Part I of the thesis, we use passive captures from residential customers in France. The data has been collected on a local platform aggregation point, namely at BAS (Broadband Access Server) level. We have used internally developed dedicated probes geographically distributed over France, and we have seen that the data is coherent between probes. This allows us to focus on a small number of different probes (three in Chap. 3, and two in Chap. 4). We analyse a week long of connection statistics in Chap. 3. This data offers a fresh view on what are the components of residential traffic nowadays. Whereas in Chap. 4, we take into account only Streaming traffic at packet level during snapshots of one hour. Nevertheless, we have performed multiple measurements campaigns that allows us to follow the evolution of Streaming traffic over a time span of three years.

2.2.1 Analysis of one week of ADSL connections

In Chap. 3, we analyse the TCP traffic during one week of 3 ADSL platforms each connecting more than 1,200 users to the Internet. We use connection statistics enhanced with a Deep Packet Inspection (DPI) tool to recognise the application. Top applications (Streaming, Web, Download and P2P) have the same volume and the same rank over days and probes in our data. The detail for each application is given, this gives us insights on what sub-class of application carries most of the bytes or the type of traffic generated. The ability to identify users allows us to follow their behavior independently of IP address churn, and the performance indicators helps us to better understand how the applications behave. The difference of traffic patterns between working days and week-end days is also studied.

The clustering analysis allows us to understand the application mix of users: the surge of plenty of customers using only Web and Streaming is quantified. Moreover, we explore the possibility to change the timescale of analysis. Our results shows that, if well chosen (namely during busy periods), a snapshot of one hour of traffic can be as representative as a whole week. Top 20 users (in terms of volume) have quite a specialized application mix, and their share of the platform load is about 10 times more than the average.

We also address the question of local platform dimensioning. We perform some simulation to show how a well chosen rate limit policy could reduce peak rate at a very moderate impact for the users.

2.2.2 HTTP Video Streaming Performance

Chapter 4 investigates HTTP streaming traffic from an ISP perspective. As streaming traffic now represents nearly half of the residential Internet traffic, understanding its characteristics is important. We focus on two major video sharing sites, YouTube and DailyMotion.

We use eight packet traces from a residential ISP network, four for ADSL and four for FTTH customers, captured between 2008 and 2010. Covering a time span of three years allows us to identify changes in the service infrastructure of some providers.

From the packet traces, we infer for each streaming flow the video characteristics, such as duration and encoding rate, as well as TCP flow characteristics: minimum RTT, mean and peak download rates, and mean loss rate. Using additional information from the BGP routing tables allows us to identify the originating Autonomous System (AS). With this data, we can uncover: the server side distribution policy (*e.g.* mean or peak rate limitations), the impact of the serving AS on the flow characteristics and the impact of the reception quality on user behavior.

A unique aspect of our work is how to measure the reception quality of the video and its impact on the viewing behavior. We see that not even half of the videos are fully downloaded. For short videos of 3 minutes or less, users stop downloading at any point, while for videos longer than 3 minutes, users either stop downloading early on or fully download the video. When the reception quality deteriorates, fewer videos are fully downloaded, and the decision to interrupt download is taken earlier.

We conclude that

(i) the video sharing sites have a major control over the delivery of the video and its reception quality through DNS resolution and server side streaming policy,

(ii) that the server chosen to stream the video is often not the one that assures the best video reception quality.

2.3 Related Work

After a very brief review of passive measurements works, we focus on HTTP Streaming studies as most of our contributions focus on this traffic.

2.3.1 Network Tomography

Network tomography is a large domain and the relevant publications are numerous. The most authoritative reference is the “Internet Measurement” book [15]. Here are a very small number of works that have inspired us in the field of passive measurements: [35, 47, 33].

2.3.2 Video Streaming Studies

Most related work on video sharing sites focuses on YouTube, which is the most prominent video sharing site. There is no previous work to compare YouTube with its competitors such as DailyMotion.

2.3.2.1 Characterisation of YouTube Videos

Many studies have tried to find out the characteristics of YouTube videos compared to *e.g.* Web traffic or traditional streaming video sites (real time over UDP and not PDL[1]). In [10], the authors crawled the YouTube video meta-information to derive many characteristics on the video contents and its evolution with video age (*e.g.* popularity). This information is used to evaluate opportunities of P2P distribution and caching.

In [13], the authors use a long term crawl of the YouTube site to derive global characteristics of YouTube videos such as that the links between *related* YouTube videos form a small-world network. Using the properties of this graph and the video size distribution, they show that P2P distribution needs to be specifically adapted to distribute YouTube videos.

In [25], the authors use university campus traffic to gather information on YouTube video characteristics and complement their data with a crawl of most popular files on YouTube. Temporal locality of videos and transfer characteristics are analyzed, and the opportunities for network providers and for service providers are studied. Another work of same authors [24] uses the same campus traces to characterize user sessions on YouTube showing that the think time and data transfered by YouTube users are actually longer than for Web traffic.

2.3.2.2 YouTube CDN Architecture

Some recent papers study the global architecture of the YouTube CDN[2]. In [2], the authors explain with Tier-1 NetFlow statistics some of the load-balancing policies used by YouTube and use these measurements to figure out traffic dynamics outside the ISP network. This method is used to evaluate different load-balancing and routing policies. Even if the methodology still holds, the data collected for this work was taken before heavy changes in YouTube infrastructure in the second half of 2008 (two years after Google bought YouTube).

The same authors study the YouTube server selection strategy [3]. Using PlanetLab nodes to probe and measure YouTube video transfers, this analysis shows that YouTube is using many different cache servers hosted inside their network or by other ISPs. Some of the load-balancing techniques used by YouTube are also revealed in this paper.

In the same vein, the authors of [53] use recent traces from different countries and access type (university campus *vs.* ADSL and FTTH on an ISP networks) to analyse the YouTube service policy. In most cases, YouTube selects a geographically close server except when these servers are heavily loaded.

The details of YouTube video streams at TCP level have been studied in [45]. This analysis of residential ISP datasets shows that the bursty nature of the YouTube video flow is responsible for most of the loss events seen. In [44], the interaction of the type of application and the type of video playback strategy with TCP is studied on Netflix and YouTube records. This shows how ON-OFF cycles can occur in video streaming trans-

[1] PDL: Progressive DownLoad
[2] CDN: Content Delivery Network

fers. Moreover a model of these behaviors is used to forecast the impact of expected changes (more mobile traffic, use of HTML5. . .) on the network.

We also would like to mention this work on load-balancing in CDNs [42]. Here the answers of DNS queries towards CDNs are stored at ISP level in order to bypass recursive DNS resolution by the CDN. This allows to directly answer to the customers with an IP address chosen by the *ISP instead of the CDN*. The evaluation of this mechanism shows an improved performance, *e.g.* download time are reduced by up to a factor of four. This work shows that a cooperation between the CDN operators and the ISPs could not only be beneficial to these actors but also to the users. In a similar vein, the study of YouTube [53] also illustrates the importance of DNS resolution in server selection and how video sharing sites (and more generally CDNs) use it to apply complex load-balancing strategies. Also the influence of traffic management between ISPs and main CDNs is underlined in [19].

CHAPTER 3

Analysis of one Week of ADSL Traffic

Vous arrivez devant la nature avec des théories, la nature flanque tout par terre.[1]

Pierre-Auguste Renoir

The Internet is a very dynamic environment: new services and usages are invented every day. The attempt to measure it is thus an endless challenge. Nevertheless, regular measures are needed to quantify its evolution. The diversity of the Internet also resides in the ways to access it: of the many access types (from work, Universities, or home), residential access is the most free. Indeed no traffic regulation apply, and only few studies have revealed the residential usage of the Internet [55, 36, 33].

In this chapter, we perform a large scale analysis of 3 different local ADSL platforms each connecting more than 1,200 users during one week of July 2011. We use an internal deep packet inspection (DPI) tool to recognise the application used by customers. With this information and connection summaries, we first study the characteristics of the residential traffic, and also focus on some specific popular services such as Facebook or YouTube. We then use clustering techniques to group customers according to their application mix. Finally, we simulate some traffic shaping techniques to evaluate their impact on platform dimensioning.

3.1 Data Collection

We have collected statistical information on 3 ADSL probes in France (located in Lyon, Montsouris and Rennes) over a period of one week of 2011: from Tuesday the 5th July to Tuesday the 12th July included. The data comprises information summary of the

[1] *You come in front of nature full of theories, then nature messes everything.* – my translation

Table 3.1: Summary of Trace Details

Trace	Nb of Clients Total	Nb of Clients Avg per Day	Nb of Cnx Total	Nb of Cnx Removed†	Nb of Cnx Avg per Day
Lyon252	1354	1284	66,231,068	86,576	7,788,835
Mont151	1009	951	50,008,566	59,393	6,251,070
Renn257	1139	1099	41,320,018	35,847	5,165,002

† data cleaning as explained in Sect. 3.1

$name$	$nr.$
1	159
2	159

Table 3.2: .....

TCP connections of all customers for each day of capture[2]. The following indicators have been computed for our analysis:

Cnx Id the source and destination IP addresses and Ports, and the time of start and stop of connection;

Application determined out of an internally developed DPI tool, we have access to the application, and also web-apps (such as Facebook) and a part of encrypted eMule and BitTorrent, still we shall mainly refer only to the *class* of the application (namely P2P, Streaming. . .);

Volumes the number of Bytes (with IP headers and also difference between last and first TCP sequence number) and non-empty Packets for each direction of the connection, we will also use the maximum volume per period of 20 seconds;

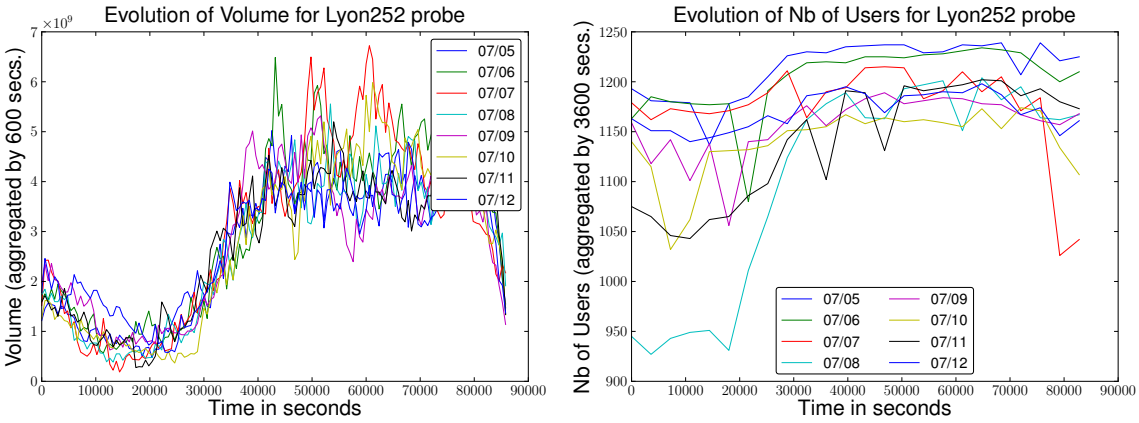
TCP Performance we define the expected sequence number as the maximum TCP sequence number seen plus the size of this packet, then a packet with a sequence number higher than the expected one is counted as a **loss**, whereas if it's lower, it's counted as a **repetition**; we also have an evaluation of RTT. All these indicators are computed for each direction of the connection.

This data is complemented with specific HTTP streaming indicators with the URL and URI of the media, and a classification according to URL of well known sites (for advertisement, video clips, porn sites. . .).

As this data is collected on the fly on probes connected to a switch after the BAS, we may have some incorrect records. Thus, we filter out the connections with incorrect statistics. The number of connections removed is included in Tab. 3.1.

In all the rest of the Chapter (and of the thesis), we call **downstream** traffic the packets coming from the Internet to the customers, whereas upstream traffic denotes packets going from the customers to the Internet.

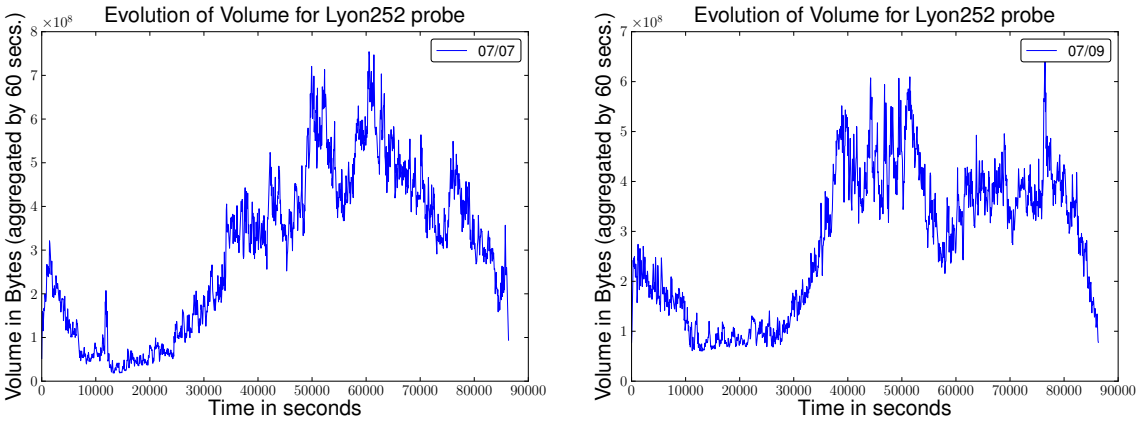
[2] the connections spanning over 2 days are split



(a) Volume (aggregated by 10 minutes)

(b) Nb. of Users (aggregated by 1 hour)

Figure 3.1: Evolution of stats captured on the probe on Lyon's probe for the whole week



(a) Thursday, 7th

(b) Sunday, 9th

Figure 3.2: Evolution of TCP Volume captured on the probe (aggregated by minute) on Lyon's probe on 5th July

3.2 Basic Characteristics of the Residential Traffic

To give an overview of the data, we start by showing the evolution of volume and number of users over the week, we plot each days on Fig. 3.1. The curve of the volume (Fig. 3.1a) shows a very similar pattern over the week. Thursday the 7th has the highest volume whereas Sunday the 9th has the lowest. As for the number of users (Fig. 3.1b), there are some differences between the days: for example, Saturday night has the least amount of users.

In Fig. 3.2, we trace the evolution of volume (TCP) over two different days of the week. On a week day (Fig. 3.2a), we observe an unsurprising camel curve with 2 large peaks around the mid-day break. Whereas on Sunday (Fig. 3.2b), we have a large plateau from late morning to early afternoon.

Table 3.3: Distribution of Application Classes according to Downstream Volume per Day and per Probe

(a) Lyon

	Top Applications (fraction of total downstream volume)				
Date	1	2	3	4	5
05/07/2011	Streaming (47.69 %)	Web (18.75 %)	Download (18.13 %)	P2P (8.49 %)	Games (2.45 %)
06/07/2011	Streaming (47.95 %)	Web (19.56 %)	Download (17.29 %)	P2P (9.25 %)	Games (2.78 %)
07/07/2011	Streaming (47.79 %)	Download (19.53 %)	Web (18.22 %)	P2P (10.26 %)	Mail (1.66 %)
08/07/2011	Streaming (44.73 %)	Download (21.40 %)	Web (18.66 %)	P2P (6.98 %)	Games (3.48 %)
09/07/2011	Streaming (48.82 %)	Download (21.67 %)	Web (15.93 %)	P2P (10.31 %)	Unknown (1.60 %)
10/07/2011	Streaming (53.38 %)	Download (17.90 %)	Web (17.24 %)	P2P (8.46 %)	News (1.02 %)
11/07/2011	Streaming (49.01 %)	Web (20.52 %)	Download (15.93 %)	P2P (9.52 %)	Unknown (1.97 %)
12/07/2011	Streaming (51.64 %)	Web (19.19 %)	Download (14.29 %)	P2P (9.78 %)	Unknown (2.62 %)

(b) Montsouris

	Top Applications (fraction of total downstream volume)				
Date	1	2	3	4	5
05/07/2011	Streaming (38.86 %)	Web (25.47 %)	Download (21.37 %)	P2P (7.81 %)	Mail (3.18 %)
06/07/2011	Streaming (44.78 %)	Web (22.48 %)	Download (17.19 %)	P2P (7.64 %)	Mail (4.17 %)
07/07/2011	Streaming (43.26 %)	Web (23.62 %)	Download (18.67 %)	P2P (6.28 %)	Mail (3.84 %)
08/07/2011	Streaming (44.94 %)	Web (22.99 %)	Download (17.42 %)	P2P (5.38 %)	Mail (4.67 %)
09/07/2011	Streaming (48.70 %)	Web (21.94 %)	Download (15.70 %)	P2P (7.42 %)	Unknown (2.94 %)
10/07/2011	Streaming (48.21 %)	Web (17.00 %)	Download (16.42 %)	P2P (13.64 %)	Unknown (2.12 %)
11/07/2011	Streaming (42.76 %)	Web (23.87 %)	Download (20.79 %)	P2P (5.65 %)	Mail (4.19 %)
12/07/2011	Streaming (39.86 %)	Download (24.96 %)	Web (21.23 %)	P2P (7.25 %)	Mail (3.72 %)

(c) Rennes

	Top Applications (fraction of total downstream volume)				
Date	1	2	3	4	5
05/07/2011	Streaming (47.23 %)	Download (24.07 %)	Web (16.12 %)	P2P (5.38 %)	News (3.19 %)
06/07/2011	Streaming (46.35 %)	Download (23.55 %)	Web (15.93 %)	P2P (7.74 %)	Games (2.40 %)
07/07/2011	Streaming (47.34 %)	Download (23.48 %)	Web (16.43 %)	P2P (7.80 %)	Mail (1.73 %)
08/07/2011	Streaming (43.81 %)	Download (26.73 %)	Web (16.25 %)	P2P (6.09 %)	Enterprise (3.41 %)
09/07/2011	Streaming (44.21 %)	Download (25.54 %)	Web (15.53 %)	P2P (8.56 %)	Enterprise (3.19 %)
10/07/2011	Streaming (41.58 %)	Download (22.86 %)	Web (19.06 %)	P2P (11.12 %)	Games (2.60 %)
11/07/2011	Streaming (36.92 %)	Download (19.52 %)	Web (15.81 %)	P2P (11.52 %)	Unknown (6.29 %)
12/07/2011	Streaming (40.15 %)	Download (19.92 %)	Web (16.78 %)	P2P (10.66 %)	Unknown (5.03 %)

3.2.1 Application Share

We summarize in Tab. 3.3 the distribution of applications per number of connections and per volumes. Note that we consider only application classes.

In Tab. 3.3 Streaming is by far the most used application in downstream volume. The next two application classes are Web and Download with very similar share of downstream volume. The 4^{th} most popular application is P2P. The order is quite stable over the days or over the different locations. The downstream volume generated by all other application is very low (less than 10%) compared to the one of the top 4 application classes.

3.2.2 Refined Application Distribution

For the most popular application classes, we detail the repartition of applications over the week in Tab. 3.4. We give the percentage of users, of flows, of volumes (down and up) for each application, and also the mean volumes (down and up) per flow. We have also computed the same statistics on application distribution over each day. As there is no notable difference in the weekly stats *vs.* daily stats, we do not include the daily data.

Table 3.4 gives us a finer view of the key components of the traffic. We first focus on the Streaming class. Here is the detail of Streaming applications in the table:

- HTTP-FLV and HTTP-MP4 are the main videos formats used by popular video streaming sharing sites (like YouTube);
- HTTP-STREAMING regroups other formats of videos (mainly used for small embedded advertisements);
- RTMP related protocols are usually used to deliver on demand video streaming (note that RTMPE is only used by a specific popular TV channel for its replay service: M6Replay).

The HTTP-STREAMING class appears for almost all users, and represents $3/4$ of flows. The mean downstream volume is very low as the durations of these ads videos are very small (a few seconds). In upstream, as the mean volume is almost the same for the main Streaming applications, the fraction of upstream volume generated by HTTP-STREAMING flows represents a large part of the total due to its huge number of flows. This table allows us to quantify that FLV is used about 4 times more than MP4. Indeed, this is the default format for YouTube which is the most popular video streaming site.

As for Web traffic which is used by almost all users, the images on the Web sites carry most of the bytes of this class. We can note that secured Web transfers (with TLS or on 443 TCP port) are used by $4/5$ of users, but it represents only 5% of flows. Finally, even if the fraction of flow and volume on Facebook is very low (less than 1% thus not in the table), two third of the clients use it.

The Download class is used by $9/10$ users because of Web Downloads. Even though HTTP File Sharing is used by a small fraction of users (7%), it represents almost half of downstream volume. This is due to the large volume per flow (16 MBytes), and the most popular file sharing site at this time was MegaUpload.

P2P is used by about 10% of users and represents 10% of the total traffic (Tab. 3.3). The information in Tab. 3.4 is very interesting to understand new trends in P2P networks:

- BitTorrent is the most popular P2P application, and even if its encrypted version is used by half of BitTorrent users, the total downstream volume that is encrypted is very low compared to non-encrypted one;
- eMule/eDonkey is the second most popular P2P application, but in this case the encrypted protocol is the most popular (both in terms of users and bytes);

Table 3.4: Distribution of Applications ordered by decreasing volume down for each class for Lyon's probe over the week (from 05 to 12 July) only applications with more than 1% vol down

App. Class	App.	Nb. Users	Nb. Flows	Vol. Down	Vol. Up	Mean per Flow Vol. Down	Mean per Flow Vol. Up
Streaming	HTTP-FLV	59.22 %	16.99 %	51.64 %	12.80 %	5 695 173,3	1 704,1
Streaming	HTTP-STREAMING	93.42 %	74.50 %	32.71 %	60.16 %	822 918,7	1 827,7
Streaming	RTMPE	4.57 %	0.17 %	6.05 %	0.34 %	68 066 027,7	4 588,0
Streaming	RTMP-Data	20.12 %	0.94 %	3.49 %	10.22 %	6 951 763,4	24 538,5
Streaming	HTTP-MP4	17.26 %	3.89 %	2.85 %	1.56 %	1 372 468,9	905,1
Streaming	RTMP	13.10 %	0.50 %	1.17 %	4.07 %	4 394 348,0	18 427,4
Web	Images Web	81.08 %	29.61 %	52.48 %	29.52 %	28 561,8	2 267,9
Web	Default http 80	86.73 %	34.37 %	25.50 %	35.24 %	11 954,8	2 332,4
Web	TLS	80.68 %	5.10 %	8.49 %	14.86 %	26 849,9	6 631,5
Web	http	80.57 %	7.85 %	7.14 %	8.25 %	14 671,0	2 390,3
Web	Unknown	99.89 %	20.09 %	3.42 %	6.76 %	2 742,6	765,5
Web	Other443	49.64 %	1.04 %	1.81 %	0.25 %	28 122,5	550,1
Download	DownloadWeb	88.20 %	52.34 %	46.57 %	9.44 %	558 654,8	1 796,6
Download	HTTP File Sharing	6.95 %	1.81 %	45.53 %	1.00 %	15 785 114,3	5 485,5
Download	AppStore	2.98 %	0.17 %	4.01 %	0.01 %	14 790 512,3	859,7
Download	Encrypted FTP	2.82 %	0.20 %	1.98 %	7.84 %	6 262 936,5	393 622,5
Download	FTP-Data-Passive	7.72 %	18.24 %	1.37 %	41.46 %	47 243,0	22 640,9
P2P	Bittorrent	7.24 %	59.65 %	45.79 %	25.56 %	30 506,6	8 995,1
P2P	eMuleEncrypted	4.68 %	3.11 %	28.53 %	43.74 %	364 595,6	295 240,0
P2P	eDonkey	2.79 %	7.67 %	16.23 %	18.46 %	84 070,5	50 512,8
P2P	BitTorrentEncrypted	4.34 %	0.60 %	8.33 %	9.58 %	552 204,8	335 471,8

- the ratio of downstream to upstream volume is 3 times higher for BitTorrent than for eMule/eDonkey (we cannot compute it in the table as the fraction of volumes are separated by direction and thus are not comparable).

3.2.3 Streaming Analysis

We focus in this section more closely on the composition of streaming traffic in Tab. 3.5. The streaming traffic consists mainly of Clips if we consider downstream volume, see Tab. 3.5a. But Advertisement and Unknown (most probably advertisements) represent the majority of flows. The categories are obtained through pattern matching on URLs with well-known services.

We can rank streaming sites in Tab. 3.5b according to their share of downstream volume. Note that the upstream volume is very low for this application class. YouTube represents more than $1/5$ of total downstream volume, and its next competitor generates only half of its traffc (10%). Then comes porn sites and TV replay sites. Note that the most popular music streaming site represents 5% of flows (but a lower share of volume).

3.2.4 Facebook

We have seen in Sect. 3.2.2 that the fraction of users on Facebook is about 2/3 of users over the week. In Tab. 3.6, we compute for each Facebook user how many days he has been connecting to the service. We learn that *most users connect every day* or all days except during the week-end.

Table 3.5: Composition of Streaming traffic over the week for Lyon's probe (tables are ranked according to decreasing downstream volume)

(a) Distrib. of type of Streaming traffic

Service	Flows	Volume Down
Clip	30.36 %	77.83 %
CatchUp TV	0.22 %	6.71 %
RadioLive	0.97 %	4.90 %
Unknown	39.23 %	4.86 %
TVLive	0.66 %	3.84 %
Advertisement	26.54 %	1.56 %
Chat	0.10 %	0.18 %
Games	1.92 %	0.12 %

(b) Popularity of Streaming sites

Service	Flows	Volume Down
Unknown	68.15 %	27.83 %
YouTube	4.27 %	22.02 %
MegaVideo	1.42 %	11.26 %
DiversX	4.88 %	9.71 %
Orange	2.06 %	6.80 %
M6	0.08 %	3.94 %
DailyMotion	1.10 %	3.67 %
AutresRadio	0.16 %	3.16 %
FranceTelevision	0.49 %	1.97 %
Deezer	4.99 %	1.70 %
Apple	0.11 %	1.53 %
Canal+	0.21 %	1.20 %
RadioFrance	0.75 %	1.08 %

Table 3.6: Frequency of Connection for Facebook Users over the week for Lyon's probe

Nb. of Days†	Nb. of Users
8	342
7	158
6	175
5	115
4	119
3	103
2	104
1	91

† Nb. of days where the user has at least one Facebook flow.

Table 3.7: Usage of Facebook (FB) for Lyon's probe

Day	FB Users	Total Users	Nb. FB Flows	Vol. Down FB
05/07/2011	871	1 306	63 669	465 919 106
06/07/2011	850	1 299	65 676	494 216 107
07/07/2011	887	1 311	56 568	390 515 974
08/07/2011	851	1 290	58 779	377 824 828
09/07/2011	713	1 250	46 566	324 139 375
10/07/2011	703	1 255	47 738	373 960 414
11/07/2011	837	1 268	53 122	403 244 646
12/07/2011	839	1 267	55 762	418 309 987

† 9^{th} and 10^{th} July were Saturday and Sunday.

In Tab. 3.7, we compute per day the number users, the number of flows and the downstream volume of Facebook. We have a very stable number of users per day except during the week-end when Facebook (as well as Internet in general) is less used by residential customers.

In Fig. 3.3, the evolution of the number of users over the week is computed over periods of 600 seconds. We have a clear daily pattern with very low night traffic and a small increase around 8 pm. Note there are very few background users. If we focus on each day separately in Fig. 3.3b, Sunday has the least amount of traffic. Also note the traffic is stable during the day with a very low decrease around 3 pm.

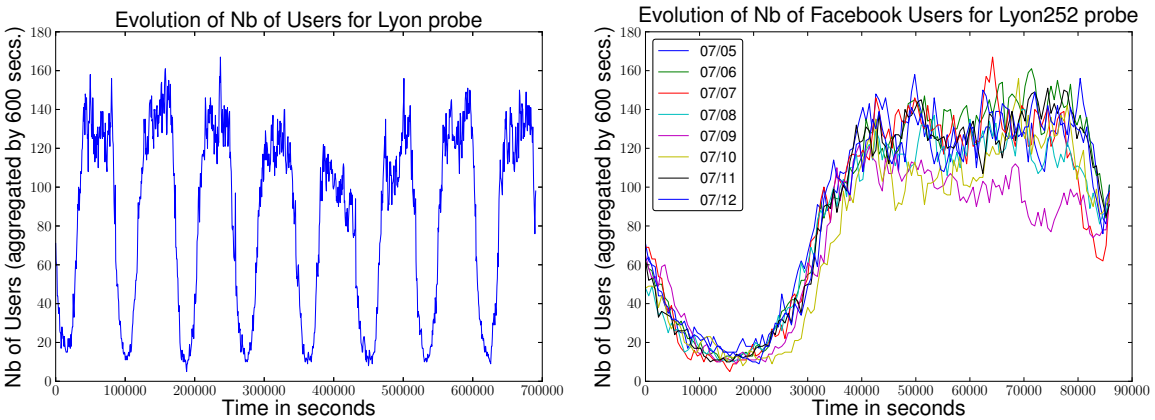
In Fig. 3.4, we plot the CDF of upstream RTT for Facebook connections. In Fig. 3.4a, we clearly have steps that are different from multiple order of magnitude. This is the same phenomenon as in [22]. If we detail per /24 prefix in Fig. 3.4b, we have a very homogeneous distribution per prefix. The prefixes are thus not shared between datacenters, and the absence of variance in RTT shows that the datacenters are well provisionned (access as well as machines). As a comparison, the YouTube datacenters studied in Sect. 7.4 can have dramatic RTT variance even at European distance.

3.2.5 YouTube

We now focus more precisely on YouTube traffic. In Fig. 3.5, we plot the CDF of average throughput per YouTube connection. We have filtered out connections smaller than 400 kBytes to remove the connections comprising of flash player download (see Sect. 4.3 for the details of YouTube functioning). Most connections (more than 95%) achieve a rate above the median encoding rate. This is more than what we shall see in Sect. 4.4.3.2 with older traffic traces. Also, even if this is not very precise, from Sect. 7.3.3 we have that most of these connections should have a good playback quality. Indeed at the time of capture, a dedicated AS was used to deliver YouTube videos and the links towards this AS were moderately loaded.

We try to find a specific daily pattern for YouTube. We thus plot the evolution of the volume and of the number of users in Fig. 3.6.

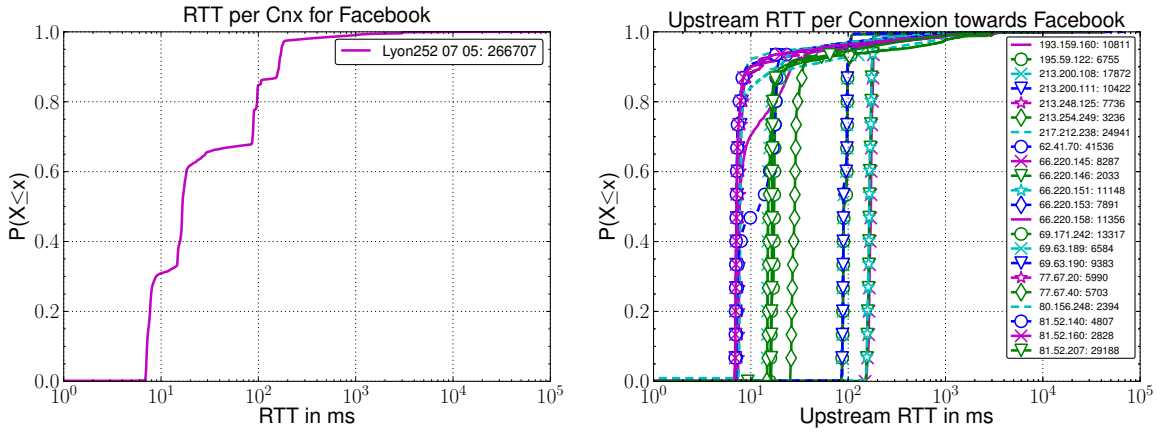
We observe a usual daily pattern with peaks during day time (Fig. 3.6a). As for the number of users (Fig. 3.6b), there is a small drop of the number of users on the weekend (9^{th} and 10^{th} July).



(a) Whole week

(b) Each day separately

Figure 3.3: Evolution of Nb. of Users of Facebook over the week for Lyon's Probe



(a) All Cnx **(b)** Per Prefix (more than 2000 cnx per prefix)

Figure 3.4: RTT from BAS towards Facebook servers

We plot the daily volume for each day of the week in Fig. 3.7a. We have very few variation in the pattern. The only remarkable point is important small peaks can be observed in the volume aggregated by 10 minutes. The number of users (Fig. 3.7b) clearly has a week *vs.* week-end days pattern with less users on the week-ends. The highest number of users is found on Wednesday the 6th July (especially in the afternoon).

3.2.6 Volumes

The CDF of global downstream volumes per application is very stable over the days of the week and also over the different probes. The only notable (but expected) point is that P2P CDF has a very large amount of small connections. We do not include this graph for brevity.

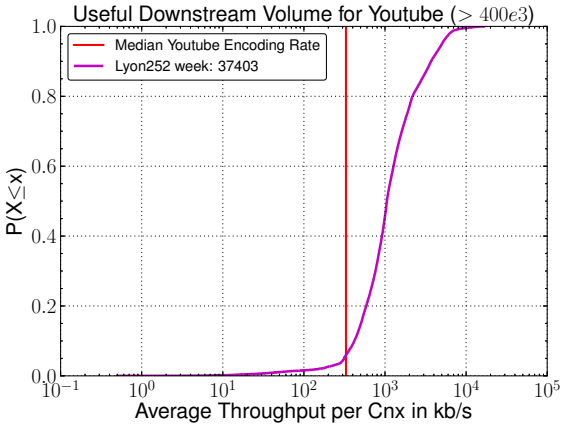
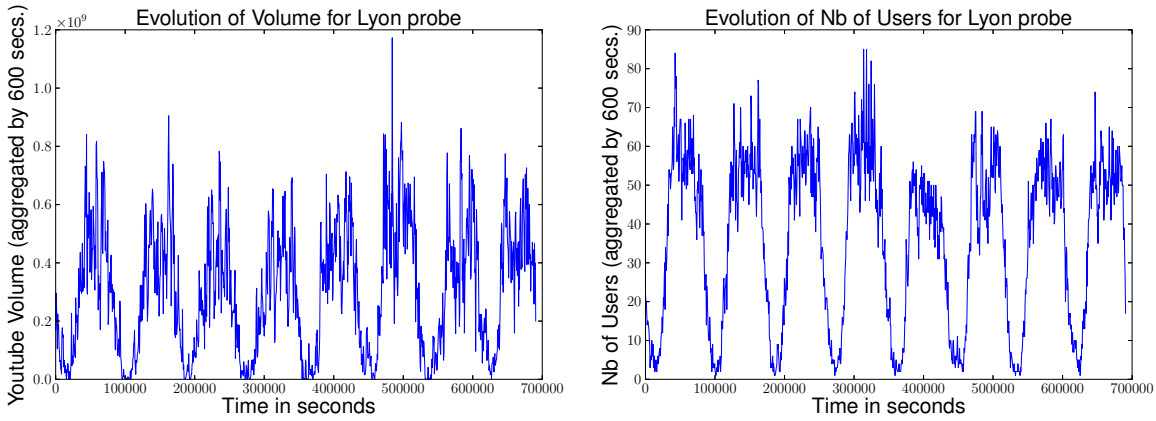
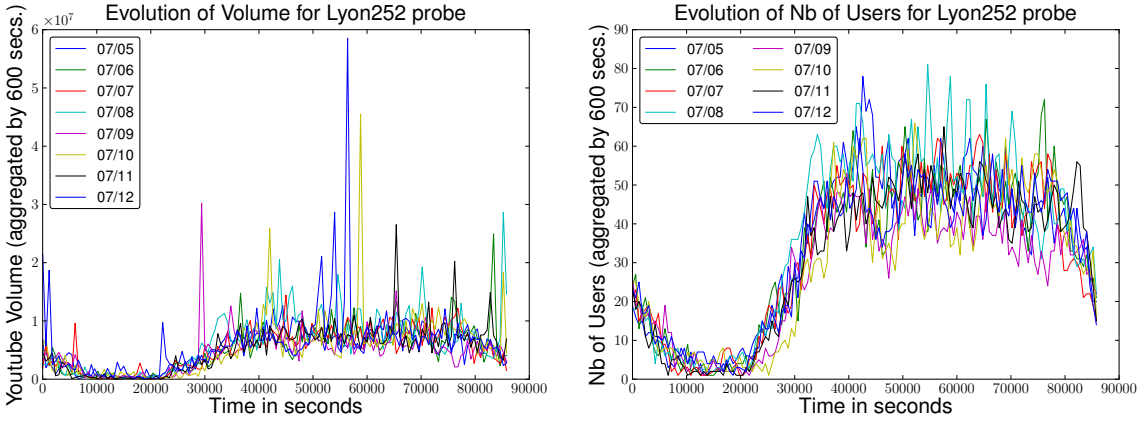


Figure 3.5: CDF of Throughput for YouTube Connections (>400 kB)



(a) Volume **(b)** Nb. of Users

Figure 3.6: Evolution of YouTube traffic over the week for Lyon's probe



(a) Volume **(b)** Users

Figure 3.7: Evolution of YouTube traffic per day

3.2.6.1 Distribution of Volume per Day

In Tab. 3.8, we sum up the total volumes (down and up), number of users and of connections for each day of capture. The average downstream rate per customer are quite low: around 5 kb/s if we consider 400 GB shared among 1000 users for 8 days. This mean that we shall focus on users generating most of the bytes (heavy hitters), or busy periods, in order to draw the characteristics of the platform.

3.2.6.2 Useful Connections

We define a useful connection as a connection with at least 1 kByte of downstream volume, and give in Tab. 3.9 the same figures as in Tab. 3.8 but considering only useful connections. The difference between Tab. 3.8 and 3.9 is mainly seen in number of connections and in upstream volume. The number of connections is approximately

Table 3.8: Distribution of Volume per Day (for Lyon probe)

Day	Nb Customers	Nb. Cnx	Vol. Down	Vol. Up
Lyon252/07/05/	1 306	8 453 211	409 GB	46 GB
Lyon252/07/06/	1 299	8 336 392	426 GB	38 GB
Lyon252/07/07/	1 311	8 656 706	430 GB	38 GB
Lyon252/07/08/	1 290	7 450 024	382 GB	32 GB
Lyon252/07/09/	1 250	6 680 858	394 GB	41 GB
Lyon252/07/10/	1 255	6 541 489	396 GB	32 GB
Lyon252/07/11/	1 268	7 811 004	395 GB	48 GB
Lyon252/07/12/	1 267	8 381 002	392 GB	42 GB

Note the 10th July was a Sunday.

Table 3.9: Distribution of Volume per Day (for Lyon probe) only Connections larger than 1 kB in downstream

Day	Nb Customers	Nb. Cnx	Vol. Down	Vol. Up
Lyon252/07/05/	1 184	2 677 139	408 GB	39 GB
Lyon252/07/06/	1 182	2 831 656	425 GB	31 GB
Lyon252/07/07/	1 197	2 680 339	429 GB	32 GB
Lyon252/07/08/	1 154	2 553 930	381 GB	26 GB
Lyon252/07/09/	1 056	1 955 062	393 GB	36 GB
Lyon252/07/10/	1 033	2 132 712	395 GB	27 GB
Lyon252/07/11/	1 154	2 586 691	394 GB	38 GB
Lyon252/07/12/	1 143	2 617 861	390 GB	33 GB

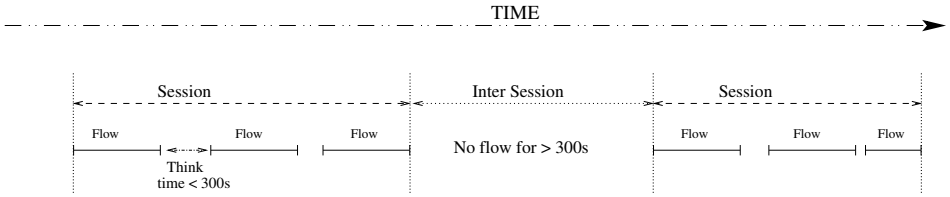


Figure 3.8: Schema of session construction

divided by 4, whereas the upstream volume is decreased by about 20%. Downstream volume is almost unchanged. This is mainly due to P2P applications that generate a lot of very small connections.

3.2.7 Users' Sessions

To figure out how the users behave we construct sessions as aggregation of connections. We explain how we have constructed the sessions in Fig. 3.8. This construction intends to mimic a usual activity pattern with multiple flows following each other with periods of silence (user's think time) in between. We have chosen a threshold for inter-session of 5 minutes. We expect these sessions correctly aggregate Streaming flows resulting from a continuous watch of multiple videos.

In Fig. 3.9, we plot the CDF of session durations for all users. We also plot per user the median duration of its sessions. Note we consider only connections lasting more than 1 second for the session construction. The median session duration is at about

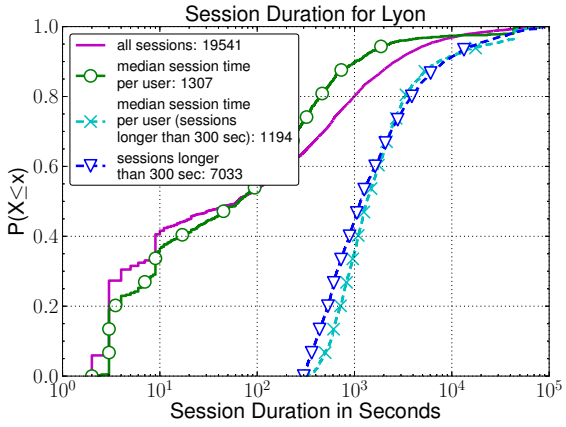
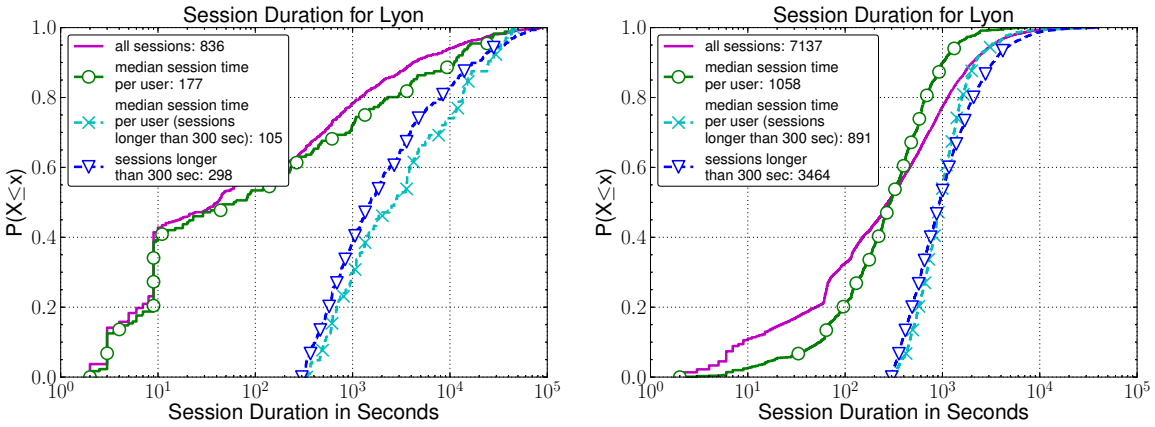


Figure 3.9: CDF of session durations for Lyon probe on 05/07 (only cnx $>$ 1 sec)



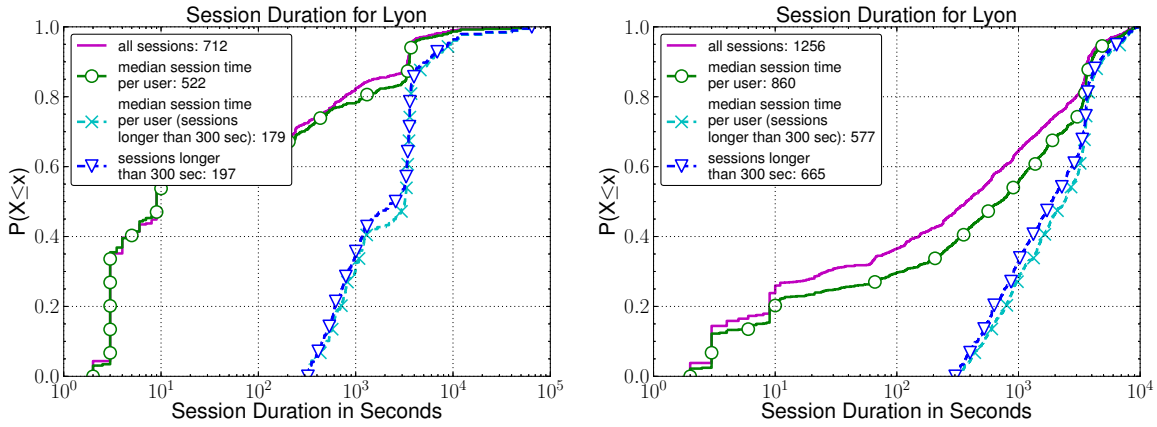
(a) P2P flows only **(b)** Streaming flows only

Figure 3.10: CDF of session durations for P2P and Streaming for Lyon probe on 05/07 (only cnx $>$ 1 sec)

2 minutes globally: this seems quite low for a real user session. Thus, we also plot in this figure the same CDFs but with only sessions longer than 5 minutes. For these longer connections, the median is of about 15 minutes (1000 seconds) which seems more reasonable for a user's session.

3.2.7.1 Sessions discriminated per Application

We have also conducted a session study based on the application used in Fig. 3.10. We focus on P2P and Streaming as their usage is quite different: background traffic for P2P *vs.* interactive usage for Streaming. Indeed we have much more short sessions for P2P than Streaming: 80% of Streaming sessions last more than 100 seconds whereas it's only 50% of P2P ones. Focusing on sessions larger than 5 minutes, the distribution is similar between the two application classes.



(a) 4 am **(b)** 9 am

Figure 3.11: CDF of session durations per hour for Lyon probe on 05/07 (only cnx $>$ 1 sec)

3.2.7.2 Sessions per hour

We study the impact of the time of the day on the session durations: as the application used change the session pattern, the time shall also have an impact. Indeed the application mix is different depending on the hour. We focus on two specific hours: 4 am and 9 pm in Fig. 3.11a and 3.11b respectively. We have a very different pattern depending on the hour:

- at 4 am, most of the connections are shorter than 10 seconds;
- whereas at 9 pm, most of connections are longer than 2 minutes.

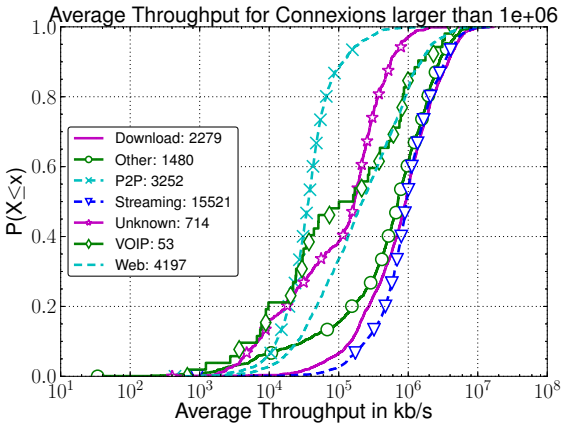
This is obviously caused by the underlying applications, but the residential usage of the Internet is the root cause: mostly interactive usage in the evening *vs.* batch usage in the middle of the night.

3.2.8 User's Level Analysis

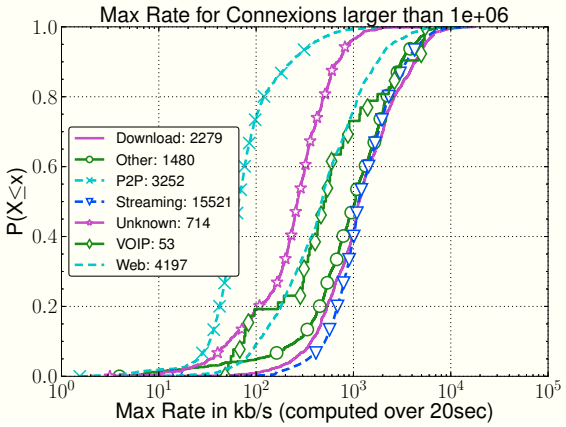
In this section, we study the usage of applications by the customers. We first look at global trends for all the platform users, and then focus on the 4 customers generating most bytes in the platform.

3.2.8.1 Parallel Connections and Aggregated Throughput

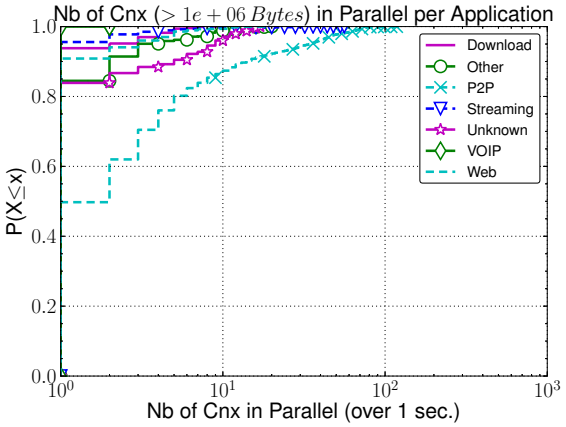
In Fig. 3.12a, we compute the CDF of the average downstream throughput per connection (only connections larger than 1 MBytes). In this graph, we treat each application independently. A global remark is that very few connections achieve average throughput close to access rate: this means that the access rate to the Internet is not at all a bottleneck for connection throughput. The main point in this graph is that P2P connections achieve a very low average throughput: 90% of P2P connections (larger than



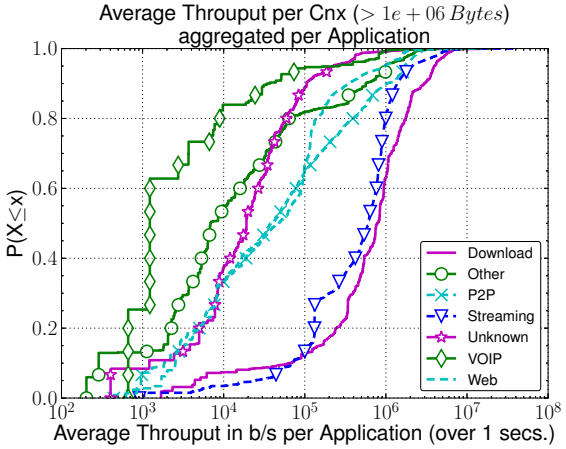
(a) CDF of Average Throughput depending on Application



(b) CDF of Max Rate depending on Application



(c) CDF of Parallel Connections



(d) CDF of Aggregated Throughput

Figure 3.12: Stats for all Users on Lyon's probe

1 MBytes) have a throughput lower than 100 kb/s. P2P applications usually limit the downstream throughput, this is done through

- the reduction of the packet size (*e.g.* for eDonkey/eMule[3]);
- or the introduction of periodic silent periods with no packets send (*e.g.* for BitTorrent).

The applications with the fastest connections are Download and Streaming.

In Fig. 3.12b, we reduce the impact of silent periods by taking the maximum rate achieved over a 20 seconds sliding window. The average throughputs for P2P are still very low: for 80% of connections it's under 100 kb/s. We can explain this because P2P applications are using a lot of parallel connections as a way to improve their download throughput. This is a method for not keeping a low individual upload speed limit.

[3]this is an efficient way to detect this application out of packet statistics (along with the TCP PUSH flag)

To check the impact of multiple connections per application, we count for each user the number of connections in parallel for each application in Fig. 3.12c. The aggregation of the number of connections is done at a specific time scale, but we have tested that a time scale between 1 and 10 seconds do not impact the results. The study of parallel connections shows that P2P applications generate a lot more parallel connections than the other applications. The P2P application used has a large impact here. We shall see it in details for some heavy hitters, but generally eDonkey/eMule generate a lot of parallel connections, and specifically much more than BitTorrent.

To be able to compare the application rates, we have aggregated per user and per application the bytes generated per seconds (by equally distributing the bytes over the duration of the connection). Then we plot the CDF of the average throughput per application aggregated over 1 second for each user in Fig. 3.12d. We can compare the achieved throughputs:

- Streaming and Download achieve the highest throughputs, with almost all connections with more than 100 kb/s and up to some Mb/s;
- P2P applications still have moderate throughputs, but the median rate is at around 100 kb/s;
- the fact that VOIP applications mainly share the same average throughput is an indication of the correctness of our data processing method.

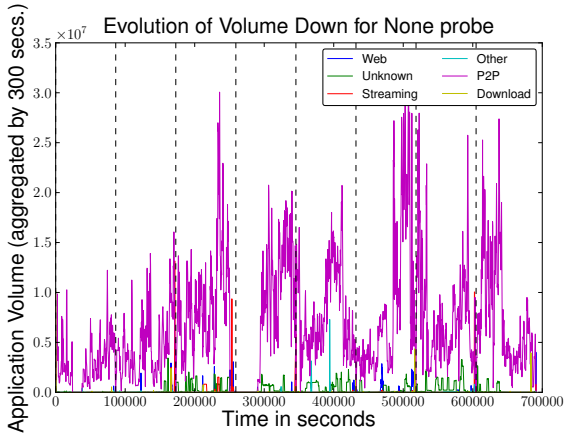
Note that the throughput limit of 1.25 Mb/s observed for YouTube in [41] can be seen in the aggregated throughput and the max rate over 20 seconds graphs as YouTube is by far the most popular streaming service (as seen in Tab. 3.5b).

3.2.8.2 Heavy Hitters

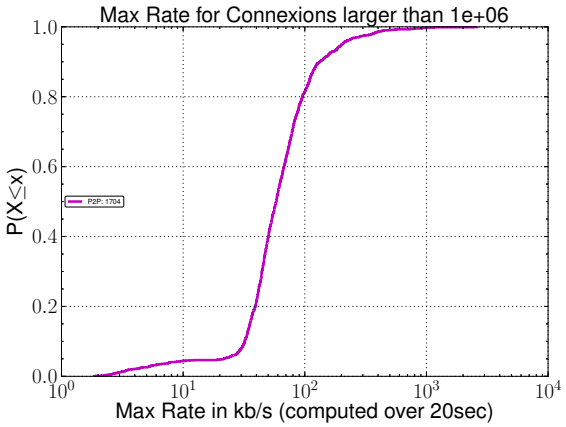
In this section, we focus on the top 4 heavy hitters to refine the discussion of the previous section. Their main characteristics are exposed in Tab. 3.10. As seen in [37], the application generating most of the volume of a heavy hitter explain a very large majority of its bytes: from 70 to 95% of its total volume. The share of downstream volume for the top 4 heavy hitters is at about 1%: with a total number of users of 1000, this means they consume 10 times more than the average. Moreover the top heavy hitter using mainly P2P in upstream direction generates as much as 15% of the upstream traffic volume of the whole platform.

Table 3.10: Top 4 users (most up+down volume) week stats

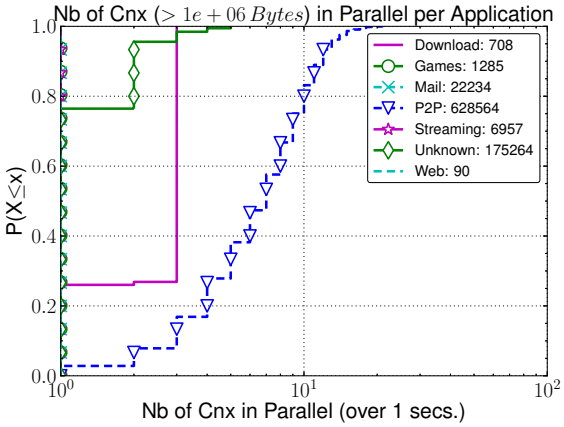
Rank	Main App.	Vol. Tot	Vol. Down	Vol. Up	Nb. Flows	$\frac{vol\_down}{tot\_down}$	$\frac{vol\_up}{tot\_up}$	$\frac{nb\_flows}{tot\_flows}$
1	P2P (95%)	63.6 GB	15.2 GB	48.5 GB	157,642	0.5%	15.1%	0.3%
2	Streaming (68%)	56.9 GB	55.0 GB	2.0 GB	828,782	1.7%	0.6%	1.3%
3	P2P (77%)	56.6 GB	49.8 GB	6.8 GB	1,355,444	1.5%	2.1%	2.2%
4	Download (94%)	51.0 GB	50.8 GB	0.2 GB	140,298	1.5%	0.1%	0.2%



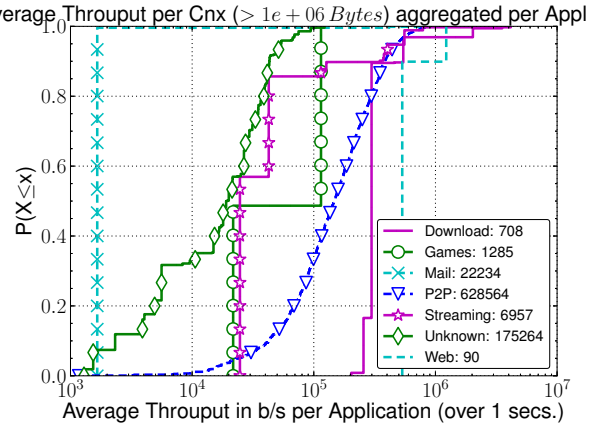
(a) Evolution of Volume over the week



(b) CDF of Max Rate depending on Application



(c) CDF of Parallel Connections



(d) CDF of Aggregated Throughput

Figure 3.13: Heavy Hitter 1

3.2.8.3 1st Heavy Hitter

This heavy hitter is mainly using P2P (95% of its bytes), and generates much more upstream traffic than other users (15% of the whole platform upstream volume). Its upstream volume is 4 times higher than its downstream one (contrarily to other heavy hitters, and to the ADSL users in general). In Fig. 3.13a, we plot the evolution of its downstream volume over time. We have that his usage of P2P is almost continuous throughout the week. This user is using mainly eMuleEncrypted protocol (for more than 90% of its volume) and then eDonkey protocol. This explains why the maximum rate over a 20 seconds sliding window is very low (around 50 kb/s) in Fig. 3.13b: with multiple connections in parallel (Fig. 3.13c), the aggregated throughput for P2P is between 100 kb/s and 1 Mb/s.

We detail in Fig. 3.14 the evolution of its upstream volume. We have that his upstream volume is very stable and is reduced around midnight every day (although this user changed its IP address only once at the end of the first day). Also note that the P2P upstream and downstream traffics are not directly correlated.

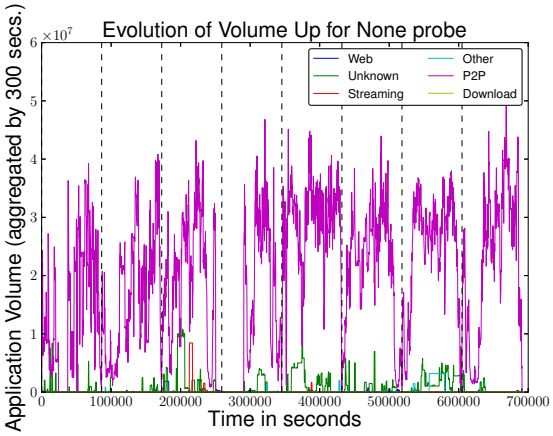
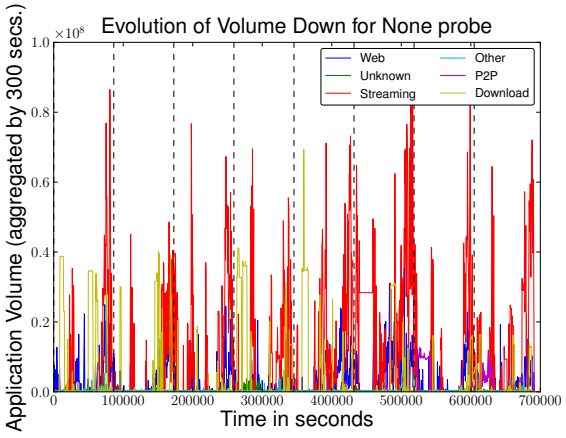
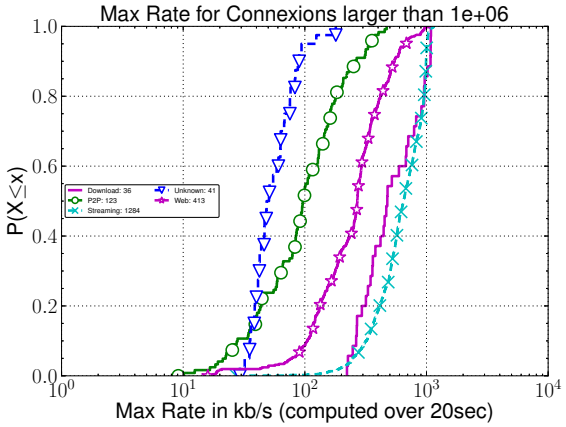


Figure 3.14: Evolution of Upstream Volume over the week for Heavy Hitter 1

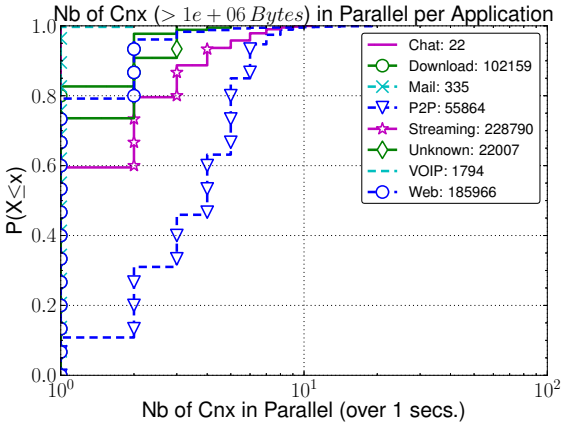
3.2.8.4 2nd Heavy Hitter



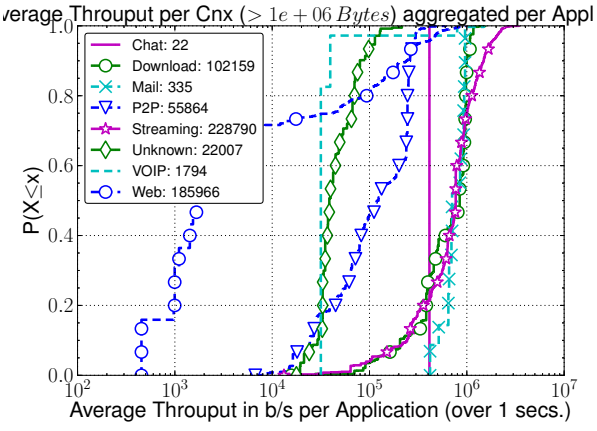
(a) Evolution of Volume over the week



(b) CDF of Max Rate depending on Application



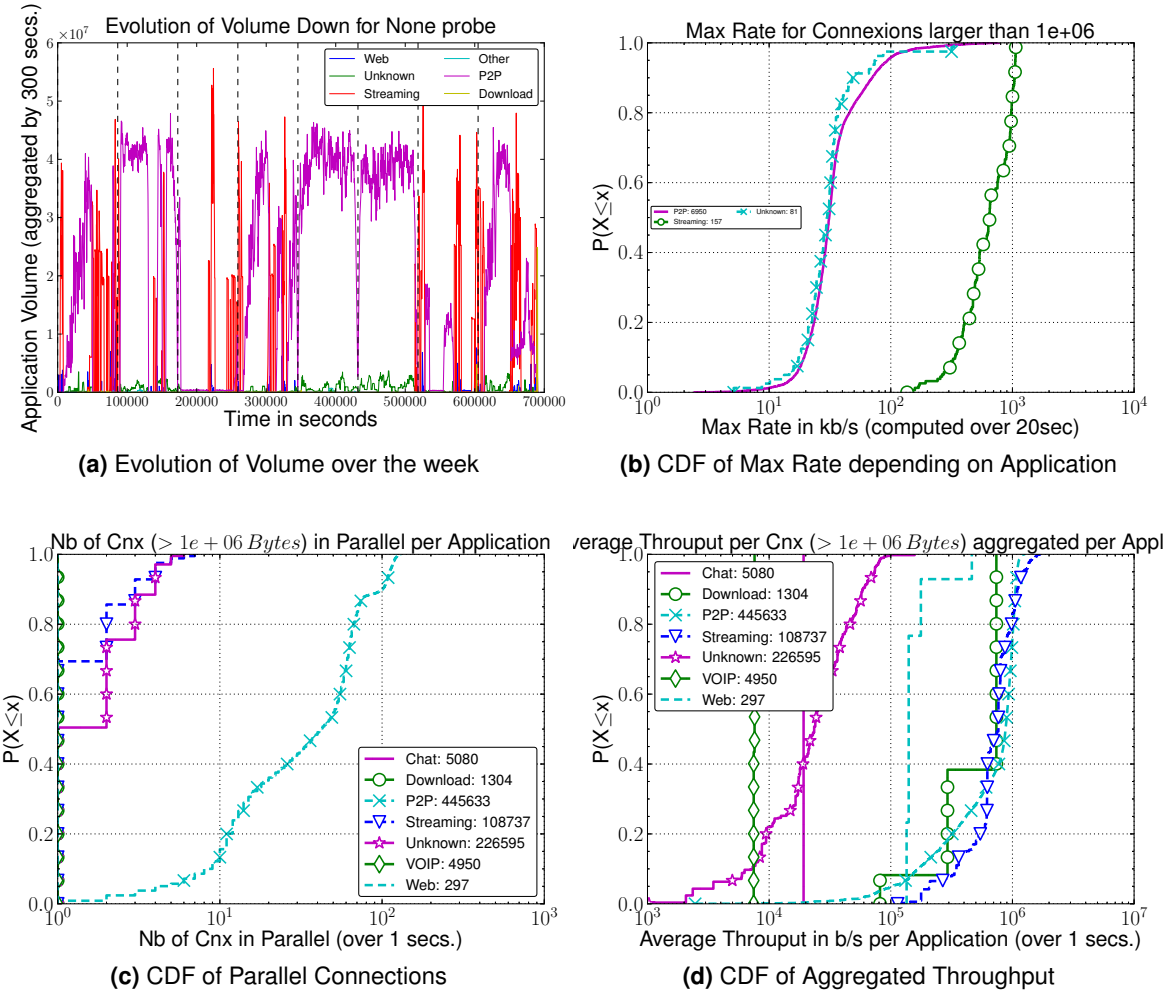
(c) CDF of Parallel Connections



(d) CDF of Aggregated Throughput

Figure 3.15: Heavy Hitter 2

This heavy hitter is using mainly Streaming (68% of its bytes), and then Download, Web and P2P. In Fig. 3.15a, we have a mix of applications, and distinct periods of activity



(a) Evolution of Volume over the week

(b) CDF of Max Rate depending on Application

(c) CDF of Parallel Connections

(d) CDF of Aggregated Throughput

Figure 3.16: Heavy Hitter 3

(but mainly on evenings). In Fig. 3.15b, the throughputs over a 20 seconds sliding window are mainly above 300 kb/s for Streaming and Download, but are very low for P2P. Then the aggregation over parallel connections increases P2P throughputs, but not Streaming or Download.

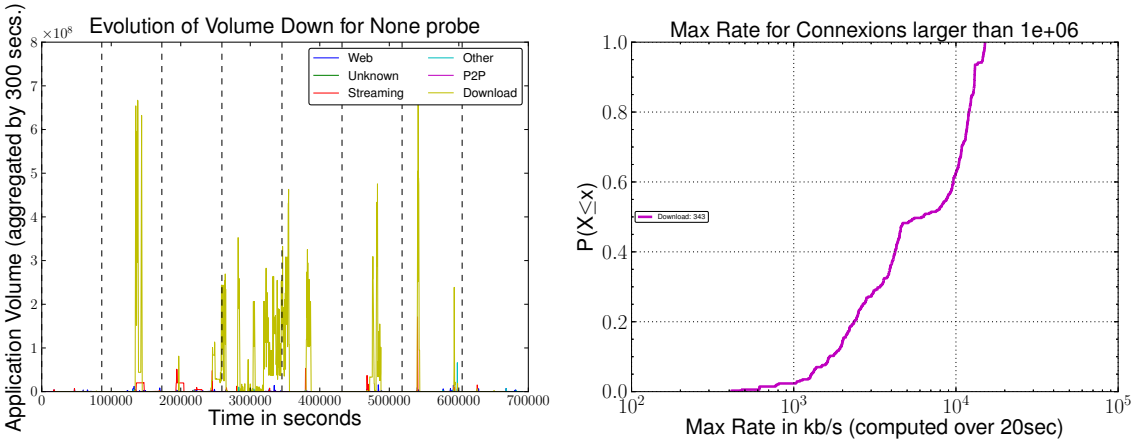
3.2.8.5 3rd Heavy Hitter

This user is the most interesting for our analysis: indeed, its daily profiles change a lot (see Fig. 3.16a). Its main applications are P2P and Streaming, but depending on the day of the week, the usage is different:

- on week-end days, almost only P2P is used (97% of volume down) and there is no daily pattern (traffic is stable throughout the day, with a large drop at midnight);
- on week days, Streaming is the dominant application (from 45 to 70%) and has a pattern with periods of high activity alternating with silent periods.

Also note that P2P and Streaming are almost never used simultaneously: on 8th July, periods of P2P and Streaming activity follow each other but never at the same time. In Fig. 3.16b, the maximum rates over a 20 seconds sliding window are one order of magnitude lower for P2P and Unknown applications than Streaming. The parallel connections for P2P is quite important with a median number of 50 parallel connection per second. Other applications have a majority of single connections at second timescale. The impact of this on the aggregated throughout is that P2P achieve similar rates than Streaming or Download, with an aggregated throughout of more than 600 kb/s in half of the cases. This means that the method used by P2P applications (in particular eMule/eDonkey) of generating a large number of parallel connections with low throughputs can effectively lead to large throughputs at user's scale.

3.2.8.6 4rd Heavy Hitter



(a) Evolution of Volume over the week

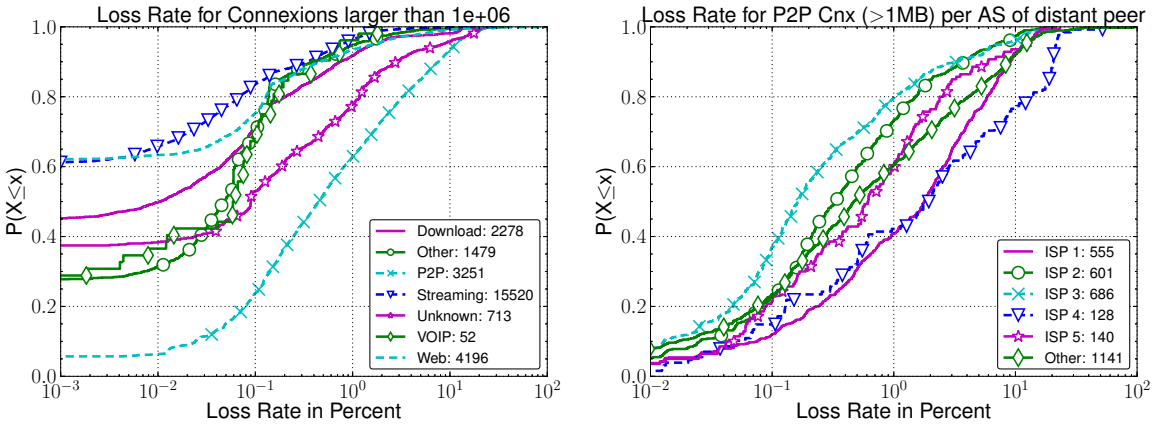
(b) CDF of Max Rate depending on Application

Figure 3.17: Heavy Hitter 4

This user almost exclusively use Download: 94% of its downstream volume. From Fig. 3.17a, we see that the activity periods are very narrow although generating a lot of bytes. This user is mainly downloading from MegaUpload. In Fig. 3.17b, the maximum rate over a 20 seconds sliding window is very high with at least 1Mb/s up to the ADSL maximum rate (18 Mb/s). This user is not using parallel connections, thus the aggregated throughputs are the same than individual analysis (graphs not shown here for brevity). This is typical of users accessing specific services (often hosting illegal contents in eastern Europe) that are achieving very high rates.

3.2.8.7 Conclusion on User Analysis

This section has allowed us to better understand the functioning of the main applications, in particular P2P. The use of multiple connections with low throughputs can lead to satisfactory download rates at user scale, but this is not a general rule: most users still have a low throughput while using P2P. Streaming and Download sites usually allow larger downstream throughputs, especially in the case of specific services leading to maximum rate of the user.



(a) CDF of Loss Rate depending on Application

(b) CDF of Loss Rate for P2P depending on ISP

Figure 3.18: Loss Rates for all Users on Lyon's probe

On the methodology plane, the focus on a few users high volume users is of great value to understand what are the usages on the platform. Moreover as these users have a large impact on the global functioning of the platform, the focus on their usage is worth it. We shall see in Sect. 3.4 that specific policies targeting these users can lead to efficient local platform management.

3.2.9 Performance Analysis

We analyse the downstream loss rate for large connections (>1 MBytes) per application in Fig. 3.18a. The loss rate for P2P and Unknown (most probably P2P) is much higher than other applications. Indeed some ISPs have specific policies for P2P traffic: we distinguish the P2P traffic according to distant peer ISP (and separate top 5 main French ISPs) in Fig. 3.18b. We clearly see the policy difference for ISP 1 and 4 with a much higher loss rate. However this does not explain completely the difference in performance between P2P and Streaming.

Figure 3.18b also indicates the locality of P2P connections: indeed most of ISPs classified as *Other* are not in France. This means that 35% percent of large P2P connections come from outside France in our data.

3.3 Clustering Analysis

Based on the work in [37], we apply the same clustering techniques to understand the application mix of the users. Moreover, we try to apply it at different timescales (namely week, day and hour) to show what is needed to carry a representative analysis.

The approach is to take into account the usage information in terms of application and not only consider the volumes. Indeed, the per applications volumes are very different – e.g., P2P applications tend to generate much more bytes than Web browsing – we miss some usage information with a purely byte-based approach. We associate to each user

Table 3.11: Ad-hoc, per application and user minimum hourly thresholds to declare application usage

Class	Volume		Number of Flows	Policy
	Down	Up		
WEB	300kB	500kB	20	All
P2P	1 MB	1 MB	10	Any
STREAMING	1 MB	1 MB	–	Any
DOWNLOAD	2 kB	1 kB	–	Any
MAIL	30kB	3 kB	–	All
GAMES	5 kB	5 kB	–	Any
VOIP	200kB	200kB	–	All
CHAT	10kB	10kB	–	Any

a binary vector, which indicates her usage of each application. We take advantage of clustering techniques to present typical application mixtures.

3.3.1 "Real" *vs.* "fake" usage

We represent each customer with a binary vector: $A = [appli_1, \cdots, appli_n]$ where n is the number of applications we consider. Each $appli_i \in \{0, 1\}$ is a indication weather the customer used application i or not. We define per application heuristics to declare that a customer actually uses a class of application. To do that, we define minimal thresholds for three metrics: bytes up, bytes down and number of flows. Depending on the application any or all of the three thresholds need to be matched. We summarize the heuristics in Tab. 3.11. The values were derived from the data as explained in [37].

Heuristics are necessary to separate real application usage from measurements artifacts (for instance misclassification due to not enough payload). For instance, large fraction of users of the platform have a single flow which is declared by the DPI tool as WEB browsing. It is hard to believe that this flow is a real web browsing activity, as current web sites tend to generate multiple connections for a single site (single search without browsing on `google.com` shows up to 7 connections). Similar problems might occur with other applications, for instance peer-to-peer user that closed his application, might still receive file requests for some time due to the delay in the P2P overlay network.

3.3.2 Choice of clustering

We have considered several popular clustering techniques to be able to understand the application mix of each user, see [27] for a complete reference on main clustering techniques. As explained in the previous paragraph, we have discretized the user's characteristics according to some heuristic threshold in order to keep only "real" application usage.

We have first tried the popular k-means clustering algorithm, and observed that the resulting clusters are difficult to match to applications. Moreover the choice of the number of clusters can dramatically change this representation.

Hierarchical clustering offers an easily interpretable technique for grouping similar users. The approach is to take all the users as tree leaves, and group leaves according to their application usage (binary values). We choose an agglomerative (or down-up) method:

1. The two closest nodes[4] in the tree are grouped together;
2. They are replaced by a new node by a process called linkage;
3. The new set of nodes is aggregated until there is only a single root for the tree.

With this clustering algorithm, the choices of metric and linkage have to be customized for our purpose.

We want to create clusters of users that are relatively close considering the applications mix they use. Among comprehensive metrics for clustering categorical attributes, the Tanimoto distance [51] achieves these requirements. It is defined as follows: $d(x,y) = 1 - \frac{x^t \cdot y}{x^t \cdot x + y^t \cdot y - x^t \cdot y}$.[5] This means that users having higher number of common applications will be close to each other. For example, consider 3 users having the following mix of applications[6]:

User	Web	Streaming	Down	P2P
A	1	1	0	0
B	1	1	1	0
C	1	1	0	1

With Tanimoto distance, users B and C will be closer to each other because they have same total number of applications even if all 3 users share same common applications.

We use a complete linkage clustering, where the distance between nodes (consisting of one or several leaves) is the maximum distance among every pair of leaves of these nodes. It is also called farthest neighbor linkage.

Due to the chosen metric, and as we choose not to prune the resulting tree, the hierarchical clustering leads to as many clusters as there are applications combinations: $\sum_{i=1}^{n} \binom{n}{i}$. In our case, we restrict the set of applications we focus only to Web, Streaming, P2P and Download.

3.3.3 Impact of Timescale on the Clustering Analysis

As we focus on top 100 users (in terms of total volume generated) in this section, we first indicate the fraction of volume they represent in Tab. 3.12. The fraction of top 50 users is highly variable throughout the day: from 50% to 82% of downstream volume in one hour. Note that the fraction for the whole day is much lower (38%) as the hourly heavy hitters may change from one hour to the other. The fraction of upstream volume is even more variable due to the first heavy hitter that is responsible for 15% of total upstream volume (see Tab. 3.10). Indeed the hourly fraction varies from 35% to 90% for the top 50 users.

[4] at first occurrence, nodes are leaves

[5] x^t stands for x transposed.

[6] 1 means application usage and 0 means no application usage.

Table 3.12: Fraction of Volume of top users

	Fraction of volume carried by top 50			top 100		
Time	Down	Up	Nb. Flows	Down	Up	Nb. Flows
07/05	38%	51%	17%	55%	64%	35%
00	79%	66%	28%	90%	76%	71%
01	78%	90%	42%	83%	99%	82%
02	82%	67%	65%	83%	70%	83%
03	81%	82%	64%	82%	83%	77%
04	79%	66%	67%	80%	67%	78%
05	69%	86%	63%	71%	87%	76%
06	81%	51%	49%	86%	65%	67%
07	64%	65%	38%	80%	81%	63%
08	62%	55%	21%	77%	73%	39%
09	61%	40%	26%	77%	68%	47%
10	55%	62%	21%	70%	76%	44%
11	58%	40%	21%	74%	65%	34%
12	58%	35%	15%	71%	70%	44%
13	50%	53%	17%	68%	64%	32%
14	57%	50%	17%	73%	69%	32%
15	58%	49%	26%	75%	70%	43%
16	59%	63%	22%	76%	75%	35%
17	58%	51%	17%	74%	71%	35%
18	58%	42%	23%	76%	65%	39%
19	58%	57%	21%	74%	80%	42%
20	66%	50%	26%	79%	77%	47%
21	60%	73%	24%	76%	87%	44%
22	69%	46%	30%	81%	72%	45%
23	64%	63%	22%	79%	85%	44%

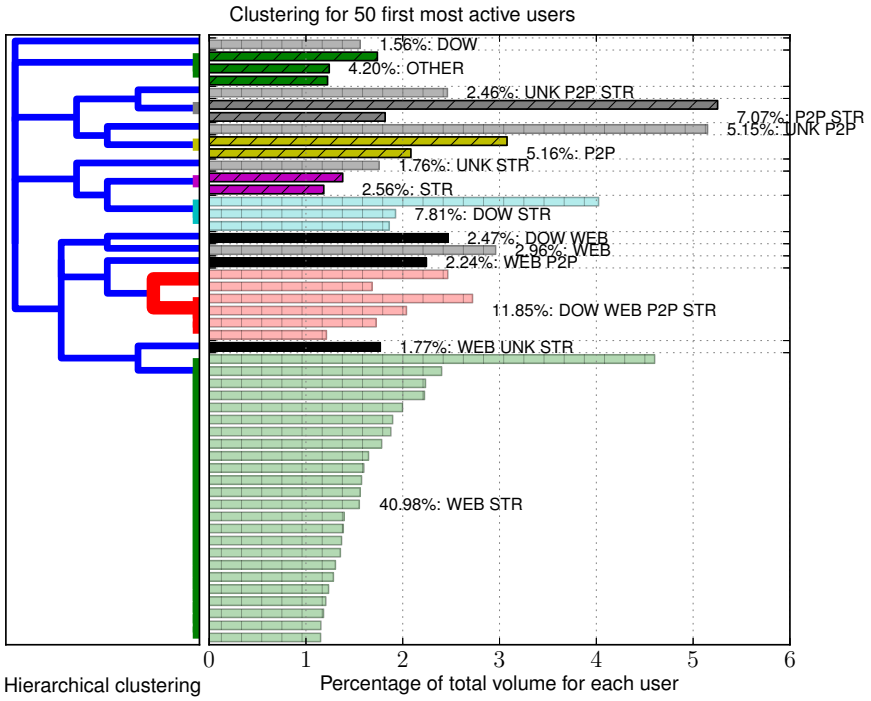
The take-away message is that the top 50 (and even more for top 100) users represent a large part of the total traffic of the platform.

3.3.3.1 Weekly Analysis

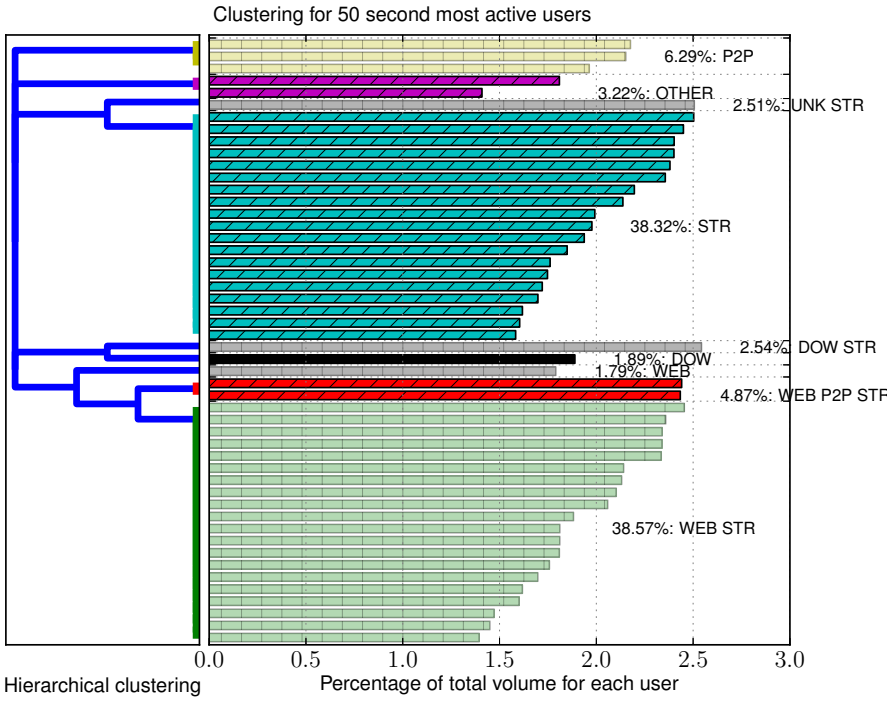
We have in Fig. 3.19a a large discrepancy in the application mix: indeed, as seen for the top 4 heavy hitters in Sect. 3.2.8.2, the Internet usage of heavy hitters is quite specific. This is mostly true for the top 25 users, even if the Web and Streaming class comprises the majority of the bytes of top 50 users. But in Fig. 3.19b, we have only two main classes of application mix: Streaming only, and Web and Streaming. This is in line with [37], and shows how usage over 6 years switched from P2P to Web and Streaming (comparing with [40]).

3.3.3.2 Daily Analysis

The daily clustering is stable throughout the week. In Fig. 3.20 and 3.21, we show a week day and a week-end day respectively. We have the same structure than for the whole week analysis: the top heavy hitters are very diverse in their application mix, whereas other heavy hitters mainly use a combination of Web and Streaming.



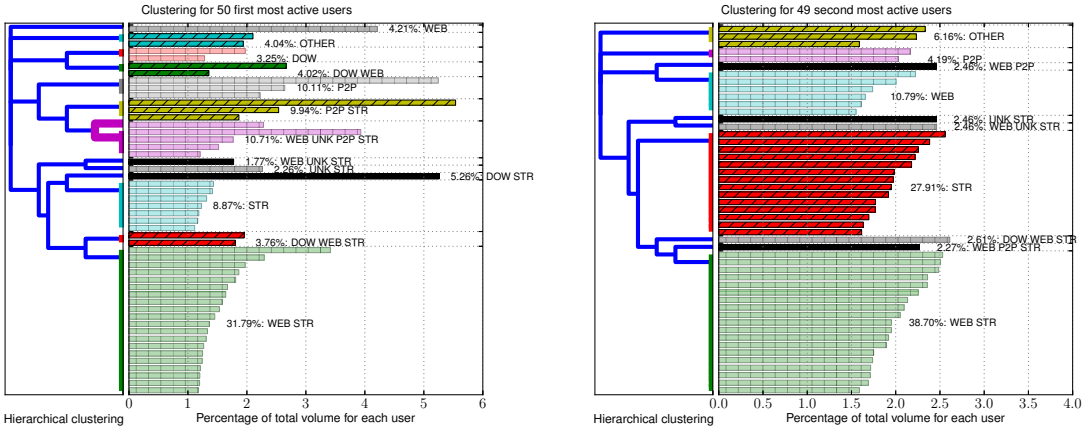
(a) First 50 heavy hitters



(b) Second 50 heavy hitters

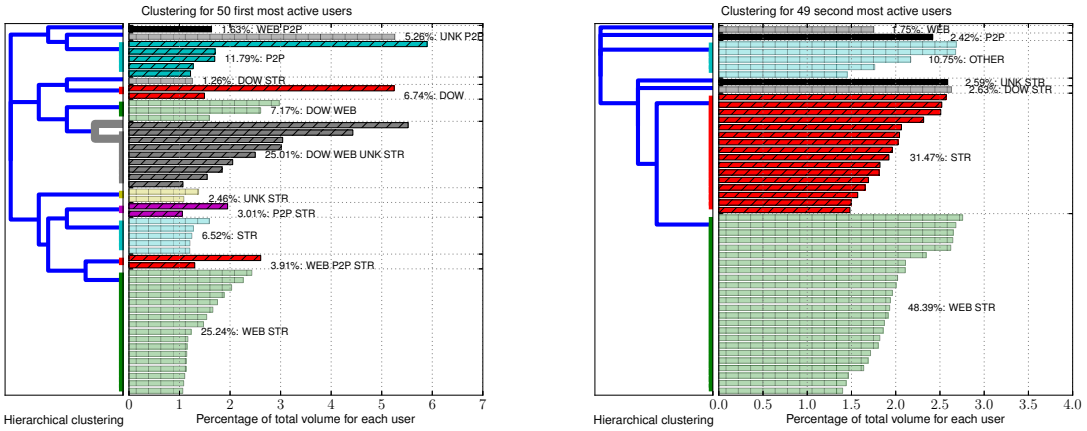
Figure 3.19: Clustering analysis for the whole week on Lyon's probe

This means that the application mix analysis done over one day or one week leads to similar results.



(a) First 50 heavy hitters **(b)** Second 50 heavy hitters

Figure 3.20: Clustering analysis for the 12th (Tuesday) July on Lyon's probe



(a) First 50 heavy hitters **(b)** Second 50 heavy hitters

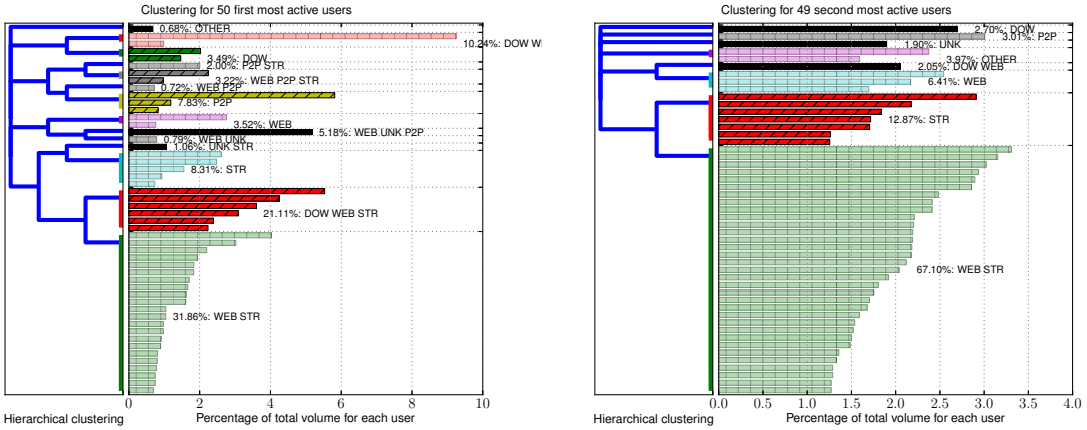
Figure 3.21: Clustering analysis for the 9th July (Saturday) on Lyon's probe

3.3.3.3 Hourly Analysis

We computed the same analysis on the 5th July at the hour scale.

The results are very similar to daily or weekly analysis from 9am to 12pm: we give the clustering results at 8pm as an example in Fig. 3.22a. This mean we have some heavy hitters using a specific mix of application, and then a large set of users with only Streaming and Web as main application. Indeed, for the 2nd set of heavy hitters, Web and Streaming represent about 90% of volume at this high usage hour (see Fig. 3.22b).

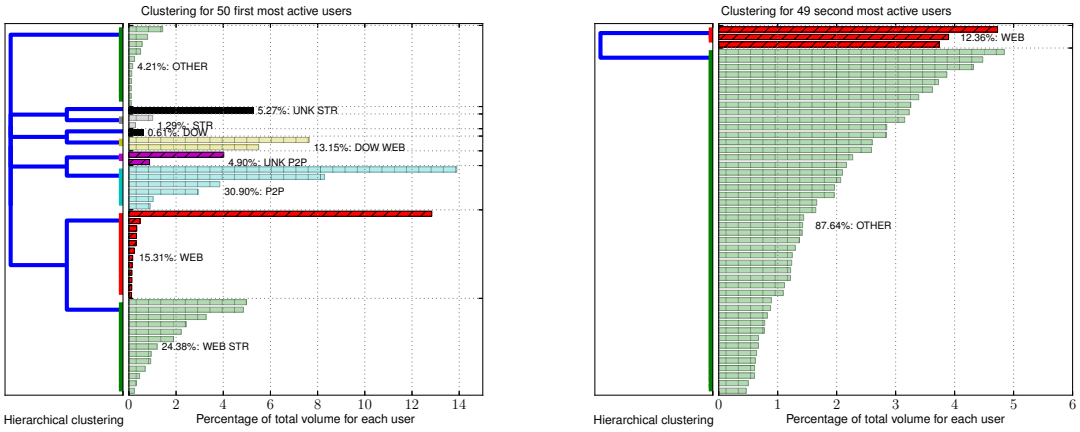
From 1am to 8am, the pattern is quite different: with only a few users generating a lot of bytes, the impact of an individual is very important. Thus the representativity of the analysis is weaker. We include the graph for 5am in Fig. 3.23a as a representative example. Users ranked after the 50th place in decreasing volume order do not use any application at this time: no real application usage threshold is triggered (see Fig. 3.23b).



(a) First 50 heavy hitters at 8pm

(b) Second 50 heavy hitters at 8pm

Figure 3.22: Clustering analysis per hour for the 5th July on Lyon's probe at 8pm



(a) First 50 heavy hitters at 5am

(b) Second 50 heavy hitters at 5pm

Figure 3.23: Clustering analysis per hour for the 5th July on Lyon's probe at 5pm

3.3.4 Conclusion of the Users Clustering Analysis

In this section, we have analysed the application mix of users of the ADSL platform. The main point is even though the first 20 top users (in terms of volume) are extensively using a specific application (P2P, Download or Streaming), the main mix is Web and Streaming. This means that users ranked 20 to 100 represent a large share of total volume, and use almost only Web and Streaming. This is interesting as ISPs should not only focus on access rate, as the main usage of the Web and Streaming is interactive. Still, global platform characteristics are strongly influenced by top 20 heavy hitters, and their continuous Internet usage.

These results are an extension of what has previously been studied in [37]. On a different perspective, we wanted to check if a change in the timescale imply a change in the results. We have shown that the daily results are coherent with weekly ones (choosing a week day is anyway more representative). A surprising point is even an

hourly analysis can lead to coherent results with the week if we choose a peak traffic hour (evenings), but also any hour between 9am and midnight.

3.4 Dimensioning

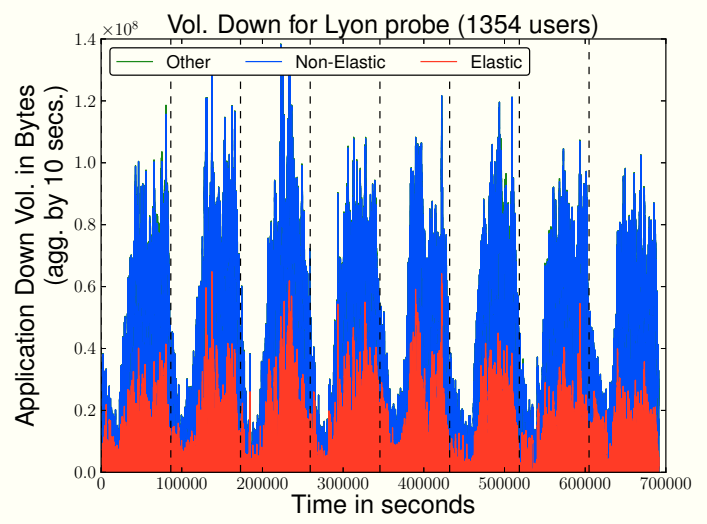


Figure 3.24: Evolution of the downstream Volume for the week on Lyon's probe per application type

In this section, we want to state some insights for local platform dimensioning. On average there is very few traffic per customer (4 kBytes/s see Tab. 3.8). Moreover the majority of the traffic volume is Streaming and Web, so we can consider it interactive. In Fig. 3.24, we aggregate downstream volume on 10 seconds slices over the week, and distinguish between different types of traffic. We call:

elastic: the traffic generated by P2P, Download and Unknown;

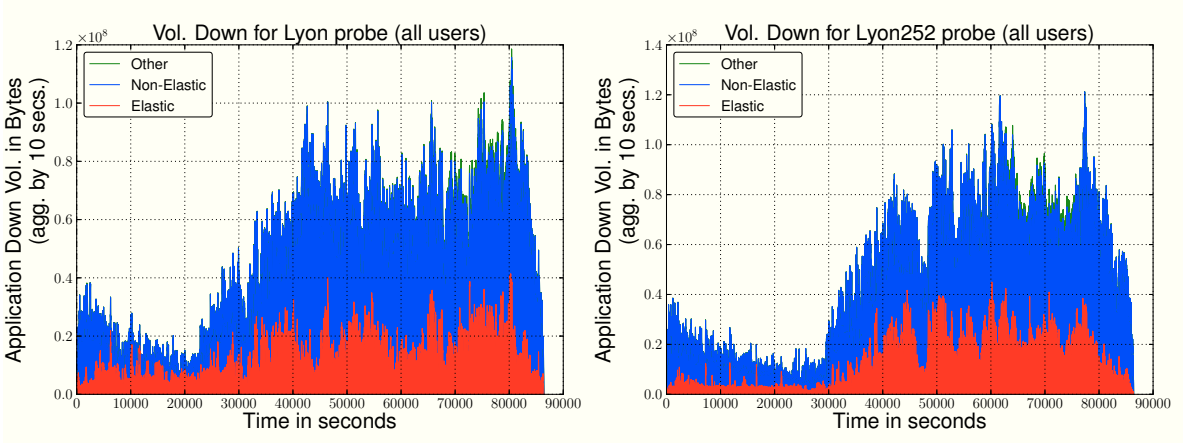
non-elastic: the traffic generated by Streaming, Web, VOIP, Games and Mail;

other: the traffic generated by all the other applications.

We adopt this classification to make clear that a perturbation in elastic traffic shall not perturb a user, whereas non-elastic traffic should be given some priority.

In Fig. 3.24, the downstream volume generated by elastic applications is stable throughout the night and increases in the evening, whereas the non-elastic traffic is almost absent during the night and increases a lot during the day. We include in Fig. 3.25a and 3.25a the daily profiles to get a better view of what is happening.

The platform dimensioning is usually done on the peak traffic, so we should focus on highest load periods. A reasonable policy for dimensioning is to upgrade the local infrastructure when 80% of the capacity is reached every day. This means that the increase of elastic traffic during peak hours is a problem whereas during the night it does not cause any harm. A reasonable policy for an ISP could thus be to limit P2P throughput during peak hours, but not at other times. As heavy hitters represents a



(a) Tuesday 5th July **(b)** Sunday 10th July

Figure 3.25: Volume evolution on the platform for the week on Lyon's probe

large part of the total traffic, another policy could be to apply the P2P limit only of these users during peak hours.

The goal of these policies is to reduce the rate only of application that support it: by adapting their throughput without the user being perturbed. Moreover, we propose to apply it only when it's useful. We explore these two policies in next sections.

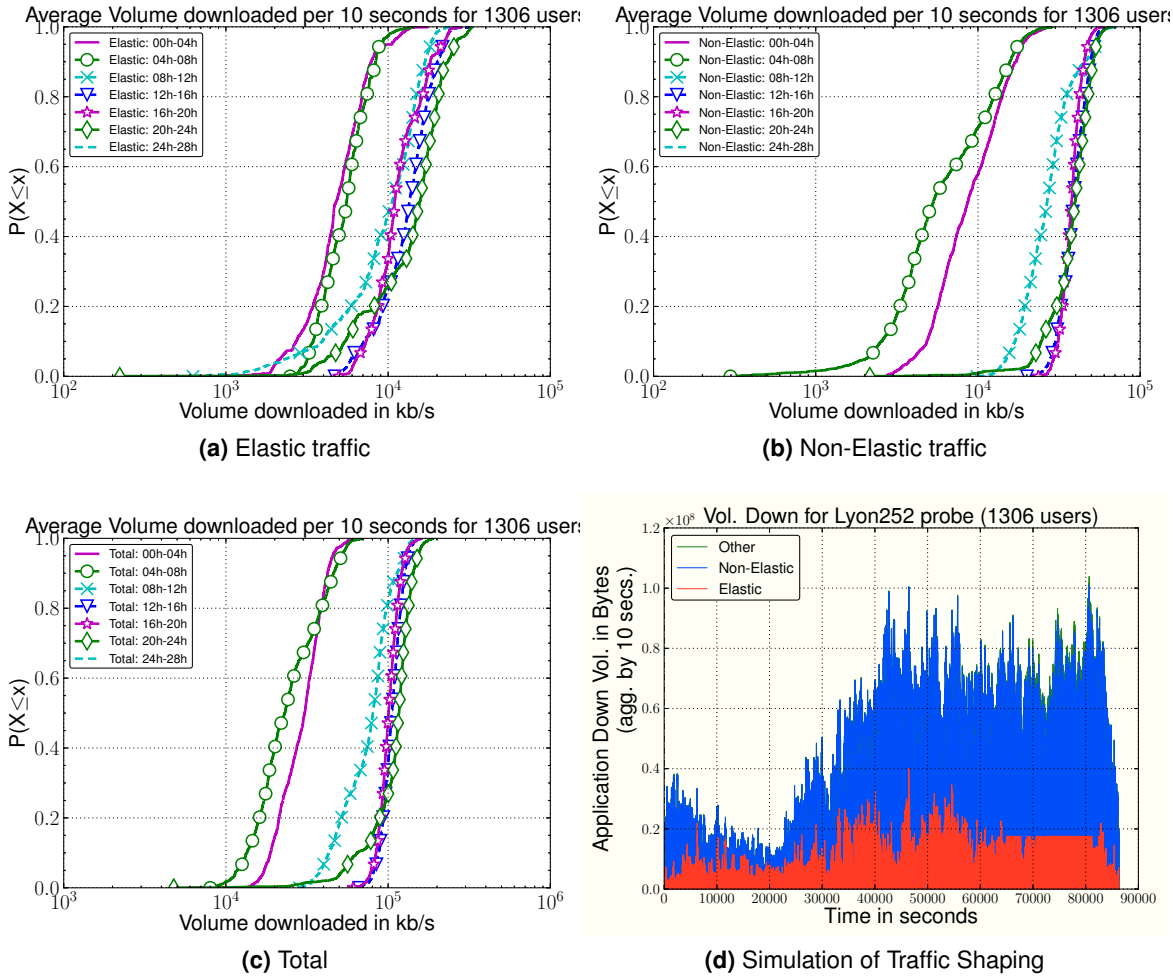
3.4.1 P2P Rate Limit at Peak Hour

We show in Fig. 3.26a and 3.26b the CDF of volume generated for elastic and non-elastic traffic respectively. We separate slices of 4 hours to show what is the impact of the period on the volume generated. We show the results for the 5th July, but all days follow the same pattern. The non-elastic traffic is much higher during the day (especially in the afternoon). Overall for the total traffic in Fig. 3.26c, the busiest period is from 8 pm to midnight.

We have thus performed a simulation of limiting the overall P2P traffic (for all users together) at 14 Mb/s during the busy period (more precisely from 7 pm to 11 pm) in Fig. 3.26d. The result is a clear reduction of total traffic, especially the peaks. If we measure the 95 percentile of traffic per 10 seconds periods, we achieve a reduction from 104Mb/s to

- 100Mb/s if we limit elastic traffic only during busy hours;
- 97.5Mb/s if we limit elastic traffic all day long.

A reduction of the peak traffic of 4 to 8% can be important for ISPs that would like to delay platform upgrade.



(a) Elastic traffic

(b) Non-Elastic traffic

(c) Total

(d) Simulation of Traffic Shaping

Figure 3.26: CDF of the Throughput generated by the platform per hour on Tuesday 5th July

3.4.2 P2P Rate Limit at Peak Hour for Heavy Hitters only

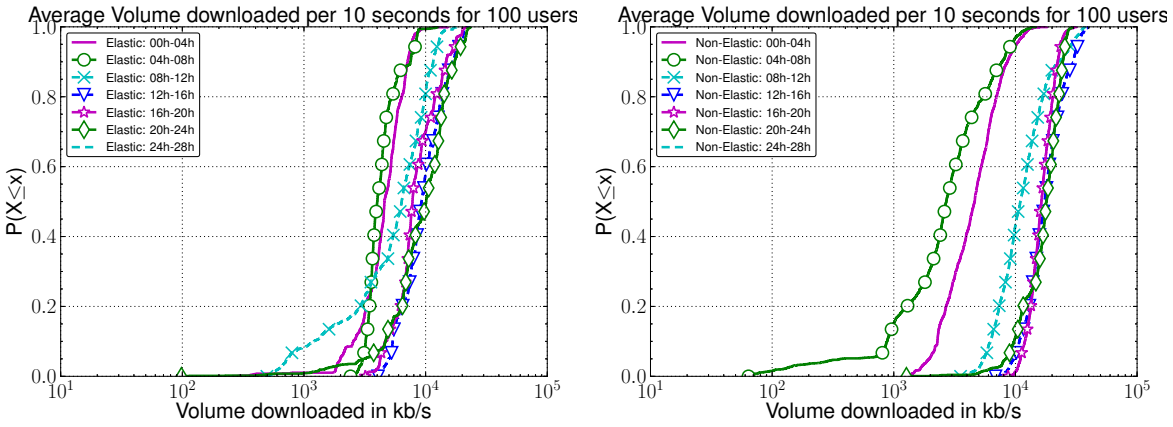
In this section, we try to apply the elastic traffic limit only to heavy hitters to reduce the number of users impacted by the throughput limitation.

Focusing on 100 top users, we have a much lower difference of throughput for elastic traffic during the night in Fig. 3.27a as compared to Fig. 3.26a. This means that heavy hitters have a more stable usage of elastic traffic than other users.

The top 100 users generate approximately the same amount of traffic than all other users (1206 on Tuesday the 5th July) as seen in Fig. 3.28. But the two main differences are:

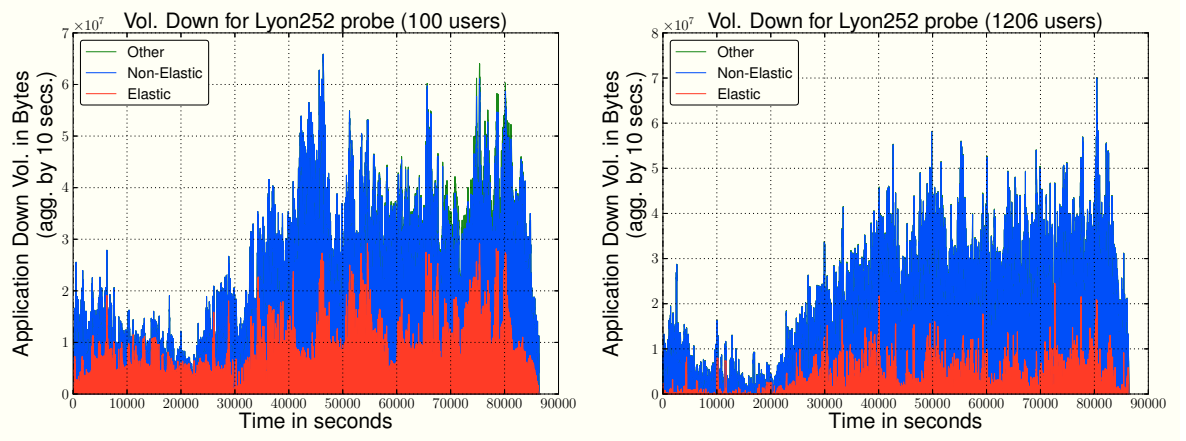
- top 100 users have much more elastic traffic;
- their traffic is much more irregular.

We have simulated a limitation of the elastic traffic of these users at 1 Mb/s (for 100 users). The application of this elastic traffic limit leads to the following reduction of 95 percentile of throughput per 10 seconds:



(a) Elastic traffic **(b)** Non-Elastic traffic

Figure 3.27: CDF of the Throughput generated by 100 top users per hour on Tuesday 5th July



(a) top 100 users **(b)** All users except top 100

Figure 3.28: Volume evolution on Tuesday 5th July on Lyon's probe

- from 60 Mb/s to 55 Mb/s if we limit all day long;
- from 60 Mb/s to 56 Mb/s if we limit only during busy hours.

For attenuating the peaks, a limitation of top 100 users only during busy hours seem sufficient. The gain for the total platform to limit elastic traffic of 100 top users at 10 Mb/s is almost the same as if limiting the elastic traffic of all users at 17 Mb/s:

- if we limit top 100 users all day long, the achieved rate at 95 percentile is 98.8 Mb/s;
- whereas limiting them on busy hours leads to a 95 percentile value of 100 Mb/s.

These values are very close to those obtained in Sect. 3.4.1.

3.4.3 Conclusion on Dimensioning

In this section, we have studied the evolution of downstream traffic volume in the perspective of platform dimensioning. If we limit the usage of elastic traffic on the platform, we can reduce the 95 percentile from 4 to 8% depending on the duration of the limitation. Moreover if we apply this limit to top 100 users only, we achieve a similar rate reduction, but with a much lower impact on users (globally only 1 user out of 10 is impacted). This can delay the upgrade of a local platform by several months, and can thus be very interesting for ISPs.

CHAPTER 4

A Longitudinal View of HTTP Video Streaming Performance

J'aime passionnément le mystère, parce que j'ai toujours l'espoir de le débrouiller.[1]

Charles Baudelaire,
Le Spleen de Paris

In this chapter, we analyse HTTP streaming traffic. Nowadays, as seen in Chap. 3, Web driven content represents about half of the Internet traffic due to the surge of many video sharing sites, and the decrease of P2P [18, 33, 36, 37].

The main video sharing sites in Europe are YouTube, DailyMotion and MegaVideo[2]. They provide a free online video sharing service, which is very popular for sharing user generated content and also video clips (videos are at most 10 minutes long on YouTube). Moreover, the streaming traffic is interactive in the sense that the user is actually watching the video during download and not after download completion as it is the case in P2P. The data is transmitted using TCP, which is not designed for interactive usage, but for elastic traffic.

We analyse the streaming traffic from an ISP perspective: this means that our main focus is on the ISP customer's perception of the Internet. The difficulty in HTTP streaming traffic analysis is that not only the network characteristics and the TCP congestion control mechanisms play a role in the user's viewing experience, but also the video sharing site itself.

We study different points that impact the video stream at flow level and relate them to user perceived interruptions. We use passive packet captures and vary important factors such as: the network access type (ADSL[3] *vs.* FTTH[4]), the video sharing site

[1] *I desperately love mystery because I always have the hope to resolve it.* – my translation

[2] The MegaVideo web sites have been shut down by the FBI on January 19 2011.

[3] ADSL (Asymmetric Digital Subscriber Line) is the main Internet access type for European residential customers.

[4] FTTH (Fiber To The Home) is a technology offering access rates up to $100\,Mb/s$, which is currently being deployed in Europe.

(mainly YouTube *vs.* DailyMotion) and the time of the day (lightly loaded afternoon *vs.* highly loaded evenings).

HTTP streaming works as follows: when a user wants to watch a video on a video sharing site, he first selects the video, *e.g.* by browsing the site portal or by receiving a direct link. Then at least 3 HTTP flows (over TCP) arise:

(i) download of the embedding web page;

(ii) download of the video player (only once in a session);

(iii) download of the video itself.

The success of these video sharing sites comes from the fact that the user can start watching the video after a very small buffering period (typically several seconds). The rest of the video is downloaded while watching, therefore its name progressive download (PDL). A user can also abandon downloading and watching if she is not interested any more.

As the main user interest and most of the volume generated by HTTP streaming comes from the actual video download, we focus on this part and we explain in Sect. 4.2 how we identify the video flows. Then, we give some general information on video sharing sites in Sect. 4.3. In Sect. 4.4, we focus on the flow level network indicators to assess the reception quality of the videos watched by the users. Finally in Sect. 4.5, we highlight the impact of streaming quality on the user *download behavior*.

4.1 Novelty of this Work

We use eight different packet traces to answer a number of important questions such as:

- Do the different video sharing sites enforce peak or mean limitations on their streams and do these limitations change over time?
- How does the YouTube CDN perform server selection for the clients of the ISP and what is the implication on the reception quality?
- How do users of video sharing sites view videos and is their viewing behavior affected by the reception quality?

Our work differs from the previous work on video sharing sites in several important aspects:

(i) Instead of characterizing all the videos available on the YouTube servers, we analyse the characteristics of *videos actually watched by users*.

(ii) We analyse video transfer characteristics to explain the performance of HTTP video streaming.

(iii) We compare two video sharing sites, namely YouTube and DailyMotion, which is one of its popular competitors. This comparison reveals a number of interesting differences, both w.r.t. performance aspects and the way these two video sharing

Table 4.1: Traces description

Type & Location	ADSL M	FTTH M	ADSL M	FTTH M	ADSL R	FTTH M	ADSL R	FTTH M
Date	**2008**/07	**2008**/07	2009/11	2009/11	2009/12	2009/12	2010/02	2010/02
Start time	$20\,h$	$20\,h$	$20\,h$	$20\,h$	$20\,h$	$\mathbf{14\,h}$	$20\,h$	$20\,h$
Duration	$1\,h\,30$	$1\,h$	$1\,h\,20$	$0\,h\,38$	$1\,h$	$0\,h\,58$	$1\,h$	$0\,h\,28$
Active Web/Str. users†	1121	1198	650	2502	795	2009	607	2763
Streaming users§	109	121	96	336	113	252	74	279
Streaming videos	428	630	405	1462	334	865	258	866
YouTube users§	41	30	48	185	47	106	46	153
YouTube videos	215	142	210	660	140	400	176	496
DailyMotion users§	25	20	16	48	12	20	13	29
DailyMotion videos	83	154	45	84	53	35	25	44

† with at least 10 flows (Web and Streaming)
§ watching at least 1 video

sites serve requests for videos. Moreover, the distribution policies of these two sites differ a lot, leading to an interesting discussion of design choices for existing video sharing sites.

(iv) Our traces cover the time from 2008–2010, which allows us to to measure the impact at network level of the modification in the infrastructure of the YouTube CDN that was put in place in the second half of 2008.

(v) We show that in our traces, the server chosen to stream YouTube videos is often not the closest one (in terms of RTT) or the one that assures the best video reception quality. These results are not in line with previous measurements [25, 53, 3].

(vi) We are the first to investigate what fraction of a video users actually download and we are also able to show that poor reception quality affects the fraction of the video downloaded.

4.2 Trace Characteristics

The main source of information for our analysis is IP packet captures taken at Broadband Access Server (BAS) level of a large European ISP. We have performed multiple packet captures at different locations. The data consists of eight approximately one hour snapshots collected on ADSL and FTTH probes from 2008–2010. The probes are equipped with dedicated capture cards (Endace DAG® card). Users[5] have been anonymised at capture time by hashing their VP/VC (ATM identifier) for ADSL and the MAC address of OLT (Optical Line Termination) for FTTH. Note that the capture reports of the cards ensure that *no packets have been lost during the capture*.

To focus on streaming flows, we first filter on the `contenttype` field of HTTP header using the same regexp as in [36]. We also remove all non-video flows such as embedded player download and advertisement contents by filtering out the keyword `player` in

[5] IP address is not used because it is not sufficient to identify users [36].

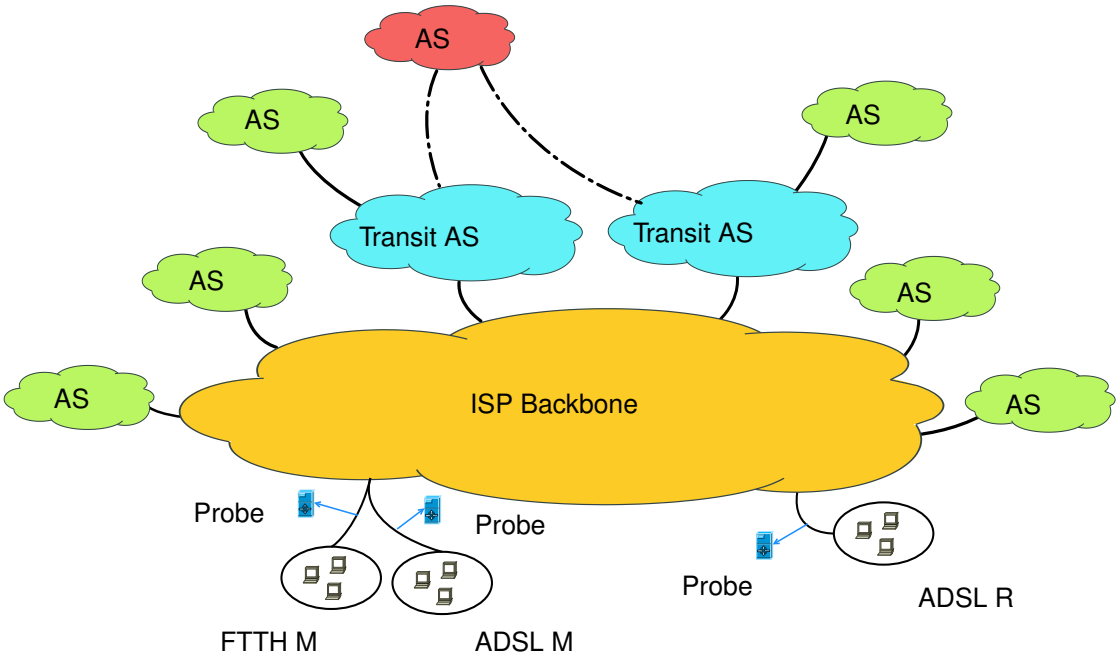


Figure 4.1: Internet seen by ISP Clients

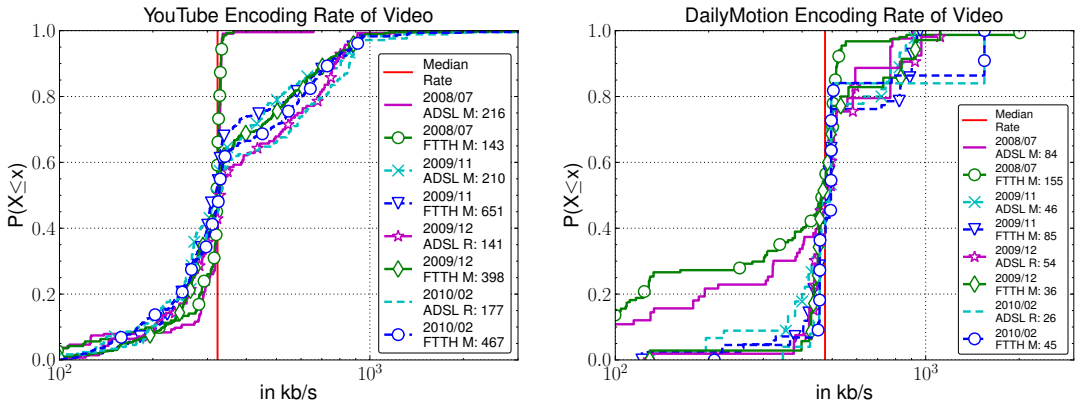
the resource name or respectively well know advertisement URLs. We process packet traces with tools to extract flow information including peak rates, RTTs and losses.

We have a specific tool to process streaming traffic that extracts relevant information about the content (mainly URL, size, and duration of the video) out of the HTTP and video headers.

We have enhanced this data with information from BGP routing tables collected at the time of capture at the ISP level, which allows us to accurately map the IP addresses of streaming servers onto their Autonomous System (AS). Most TCP traffic indicators have been derived via an internal packet processing tool and some loss indicators have been calculated using the `tstat` software [54].

The details of the packet captures are given in Tab. 4.1. We have two *old* traces from July 2008, and six traces taken in 2009 and 2010. After the acquisition of YouTube by Google, changes to the architecture of the YouTube CDN occurred in the end of 2008. We are able to see the impact of these modifications in our data (mainly the switch from the YouTube AS to the Google AS and to a new YouTube EU AS). Since then, no notable infrastructure changes happened. Note that FTTH M 2009/12 trace has been taken in the afternoon, whereas all the other traces have been captured in the evening, which is the period of highest network load for residential customers. Traces are captured in two geographically different locations and labelled with their access type and location indication. We label traces taken at a central site near the *Main* ISP peering point with an **M**, and with an **R** those taken at a *Regional* site.

Figure 4.1 shows a global view of the Internet and the location of our probes. We show the connection of the ISP customers along with ASes with direct connection as well as ASes indirectly connected to the ISP Backbone.



(a) YouTube **(b)** DailyMotion

Figure 4.2: Video Encoding Rates for YouTube and DailyMotion

4.3 HTTP Streaming Context

Due to the prominent usage of HTTP streaming [33], this traffic is important for ISPs in terms of resources required inside the ISP and at the peering points. After a brief description of the most popular video sharing sites, we evaluate the video encoding rates of the main video sharing sites as it is a key factor of the video quality and network resource consumption. Then, we briefly explain how DNS resolution works as it will be useful for the further analysis. Finally, we give an example of the distribution of the traffic across the different ASes of the YouTube CDN.

4.3.1 Most Popular Video Sharing Sites

The most popular video sharing sites in our traces are YouTube followed by DailyMotion and MegaVideo aggregating respectively 30%, 14% and 11% of the total streaming volume. The rest of the streaming volume comes from TV channels offering *replay* of their programs and porn sites. Note that the total streaming downstream volume represents about 40% of the total traffic for ADSL traces and about 30% for FTTH traces.

YouTube is the most popular video sharing site all over the world with more than 100 Million viewers per month just in US [14]. It has been bought by Google in November 2006. One of its major competitors in Europe is DailyMotion which is among the top 50 most frequented websites in the world with 70 Millions unique visitors monthly [8].

4.3.2 Video Encoding Rate

We are interested in video encoding rates to have an idea of the necessary mean reception rate of a flow required to watch the video without interruptions. The trend of encoding rates gives also an interesting insight into content providers choices and adoption of higher quality formats. We compute the encoding rate by dividing the content size announced in HTTP header and the content duration announced in the video

Table 4.2: Distribution of Volumes (in percent) and delays (median value of minimal upstream RTT per flow) in milliseconds per AS for YouTube and DailyMotion

	YouTube										DailyMotion			
	YT EU AS 43515		GOO AS 15169		YT AS 36561		C&W AS 1273		GBLX AS 3549		DM AS 41690		LL AS 22822	
	Vol.	RTT	Vol.	RTT	Vol.	RTT	Vol.	RTT	Vol.	RTT	Vol.	RTT	Vol.	RTT
2008/07 ADSL M	–	–	39%	21	61%	113	–	–	–	–	67%	2	33%	14
2008/07 FTTH M	–	–	–	–	100%	114	–	–	–	–	61%	1	39%	14
2009/11 ADSL M	90%	114	1%	21	–	–	5%	21	4%	117	100%	2	–	–
2009/11 FTTH M	91%	108	1%	20	–	–	5%	215	3%	106	100%	1	–	–
2009/12 ADSL R	93%	116	7%	32	–	–	–	–	–	–	100%	14	–	–
2009/12 FTTH M	80%	101	20%	20	–	–	–	–	–	–	100%	1	–	–
2010/02 ADSL R	56%	126	32%	38	–	–	9%	29	3%	52	100%	14	–	–
2010/02 FTTH M	60%	110	18%	25	–	–	19%	24	3%	108	100%	1	–	–

header (FLV, MP4, 3GPP). Note that this is coherent with the bit-rate announced in the video header but easier to compute with multiple video formats. Video encoding rates are quite standardized inside a video sharing site (see Fig. 4.2).

Median encoding rates of two most popular video sharing sites are quite close: for YouTube, the median encoding rate of $330\,kb/s$ is slightly lower than for DailyMotion, which is $470\,kb/s$. Encoding rates for YouTube vary quite a lot and many YouTube videos use encoding rates above or below the median rate (see Fig. 4.2a). Encoding rates of DailyMotion videos in recent years show little variance and the majority of videos have an encoding rate equal to the median rate (see Fig. 4.2b).

4.3.3 Domain Name System (DNS)

As explained in [2], retrieving a YouTube video begins with a connection to the YouTube web server that returns the URL of the video stored in the YouTube data center (cache server URL *e.g.* `v7.lscache1.c.youtube.com`). This URL is then resolved via a DNS lookup, which returns the IP address of a server delivering the video.

Load balancing techniques by the operator of the video sharing sites can be applied via DNS resolution: the recursive nature of DNS resolution allows the DNS server of the domain to take into account internal policies to answer with the most appropriate server IP address. If a content is highly requested and replicated (as are videos of main sharing sites), the DNS server of the video sharing site can choose to redirect the *same* URL to one of several servers. This technique can be used to balance the load but also to better take into account network path characteristics (*e.g.* return the server closest to the user). As we will see later, the same URL can even be resolved to IP addresses in different Autonomous Systems (AS), which may greatly impact the flow characteristics (see Tab. 4.2 and 4.3).

Table 4.3: Distribution of number of distinct YouTube ASes per client for clients with at least 4 YouTube videos

Trace	Total # ASes	# distinct ASes per client 1	2	3	4
2008/07 ADSL M	3	33%	53%	13%	–
2008/07 FTTH M	1	100%	–	–	–
2009/11 ADSL M	3	65%	5%	30%	–
2009/11 FTTH M	4	71%	14%	12%	4%
2009/12 ADSL R	2	50%	50%	–	–
2009/12 FTTH M	2	58%	42%	–	–
2010/02 ADSL R	3	21%	53%	26%	–
2010/02 FTTH M	4	13%	53%	21%	13%

4.3.4 Distribution of Traffic across ASes

We present in Tab. 4.2 the main characteristics of the ASes providing YouTube and DailyMotion videos. The measured delay corresponds to the round trip time *from the probe towards the server and back*, also referred to as upstream RTT and defined in Sect. 4.4.1. We see that the *former* YouTube AS (36561) is no more used today. The YouTube EU AS (43515) streams the majority of the bytes in all 2009–2010 traces, which is quite different to what was observed in previous studies [53, 3] that had identified the Google AS (15169) as the one serving most of the streams. Note that we measure for the YouTube EU AS an upstream RTT in the order of $100\,ms$, which corresponds to the RTT between Europe and the East Coast in the US. The Google AS, which also serves YouTube videos, has a much lower upstream RTT between 20 and $40\,ms$. Other ASes (Cable&Wireless and Global Crossing) are also used for streaming YouTube videos, but only marginally.

Previous work [53] showed that the server selected by the YouTube CDN for streaming the video is usually the closest one to the user with notable exceptions only at peak hours. For our traces, this finding does not hold since the AS that is farthest away is used to serve the majority of videos (up to 90% in terms of volume the for 2009/11 and 2009/12 traces).

In Tab. 4.3, we see that the same client can be directed to a different AS when requesting multiple videos, even in a timescale of one hour. As this redirection mechanism happens via DNS, the video sharing site has full control to select the AS and the server that will stream the video. We shall see in Tab. 4.5 that the choice of the originating AS has a significant impact on the video reception quality.

In the case of DailyMotion, almost all videos are served by the DailyMotion AS (41690) which has a median delay of $2\,ms$ over all traces (resulting in a total RTT of $42\,ms$ on ADSL and $7\,ms$ on FTTH). The only exception is found in our 2008 traces where about $1/3$ of the videos were coming from the LimeLight AS (22822) with a median delay of $14\,ms$.

4.4 Flow Performance Indicators

In this section, we measure various metrics such as RTT, peak rate, mean rate, or loss rate in order to understand the performance experienced by the flows. One of the novelties of our analysis is that we compute all these metrics for each different AS that host servers of the video sharing site, which allows us to reveal the existence of considerable performance differences between different ASes of the *same* video sharing site. In all the graphs, the number after the label in the legend indicates the number of data samples (videos).

Finally, we derive a decision graph for monitoring a platform (here a BAS connecting ISP customers to the Internet).

4.4.1 Round Trip Time

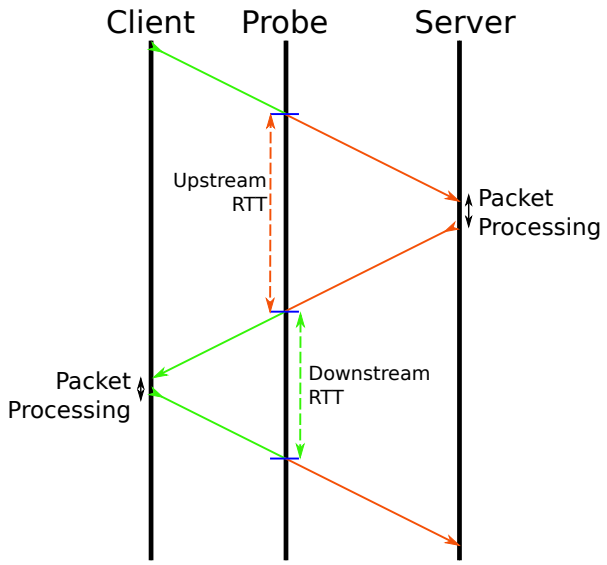


Figure 4.3: RTT Computation Schema

Round Trip Time (RTT) is defined as the time between the emission of a data packet and the reception of its acknowledgement, as shown in Fig. 4.3. In order to get an idea of the distance between the client and the server, we use the minimum RTT of all the RTT measures of a flow. As the probe that captures the packets is located between the customers and the server, we separate the RTT in two parts:

upstream RTT delay from the probe towards the server (in the Internet) and back;

downstream RTT delay from the probe towards the local user and back.

As the infrastructure between the probe and the remote site is the same on different access types, this allows to compare the distance to streaming servers across the two different access types, ADSL and FTTH. Note that *real* RTT between the client and the server is the sum of the upstream and downstream RTT.

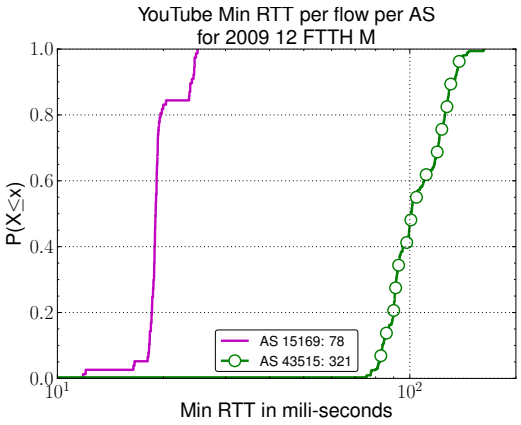


Figure 4.4: Upstream RTT according to source AS for YouTube 2009/12 FTTH M trace

As the probes are quite close to the clients, downstream RTTs are very short and very stable (also for non-streaming flows). The CDF of downstream RTTs (between the BAS and the customer), not shown here, reveals that:

- almost all ADSL video flows have approximately $40\,ms$ of downstream RTT;
- whereas FTTH video flows have downstream RTTs between $1\,ms$ to $5\,ms$.

For upstream RTTs, we often have a CDF that is multi-modal, which can be explained by looking at the AS of the flows. The RTT of the two main ASes used by YouTube differ in their upstream RTT by almost one order of magnitude (see Fig. 4.4). Also for each AS, the RTTs are very stable showing little variance. We present the results only for one trace, as they are similar for the other traces.

4.4.2 Peak Rate

We define the peak rate of a flow as the maximum number of bytes received over a sliding window of $100\,ms$. Studying the peak rate allows us to find out rate limitation policies of video sharing sites and their interplay with the access rate.

Note that the peak rates are **not** directly related to the end-user performance as the conversion of peak rate into mean rate is not straightforward.

4.4.2.1 Peak Rates for FTTH flows

As we compute peak rate over a sliding window of $100\,ms$, we can expect that our measure will not be systematically perturbed by the congestion control of TCP. We use only the FTTH traces for which the access rate is $100\,Mb/s$ and where the receiver window size is large enough to assure that the sender rate is not limited by too small a receiver window. In Fig. 4.5, for the FTTH M traces of 2009 and 2010 the median receiver window size for clients is in the order $600\,kBytes$ and the maximum around $3.5\,MBytes$.

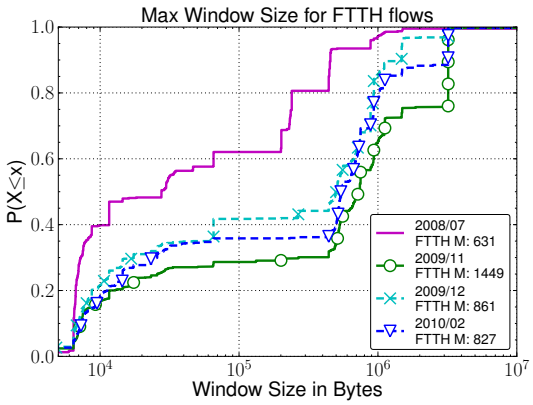
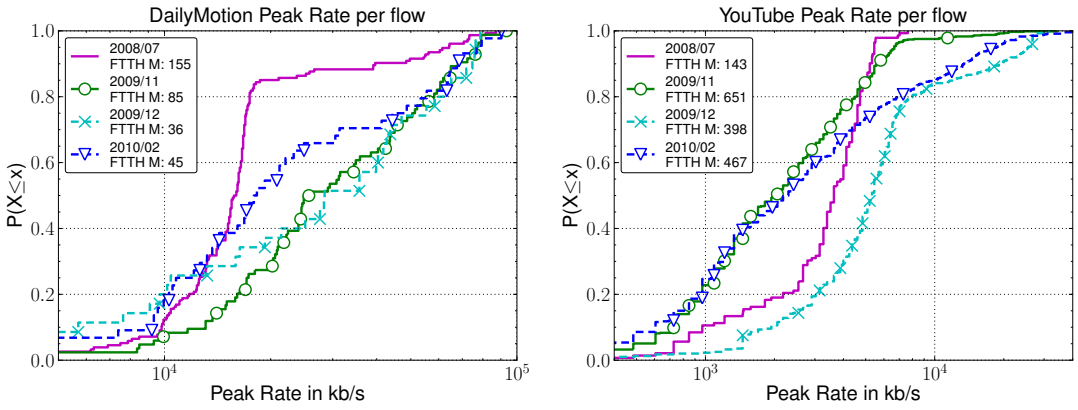


Figure 4.5: CDF of window sizes for FTTH M streaming flows



(a) DailyMotion **(b)** YouTube

Figure 4.6: Peak Rates per Flow

Thus we rely on FTTH M flows to grossly evaluate the available capacity along the path and identify if there is a peak rate limitation policy for video sharing sites. In case the measured peak rate is below $100\,Mb/s$ (default access rate), it is limited either on server side or between the server streaming the video and the BAS. This path between the BAS and the server comprises Backbone links, a connection from the Backbone towards the destination AS (traffic can pass through multiple ASes if there is no direct link), and the links inside the destination AS.

We consider YouTube and DailyMotion separately as they have very different rate limitation policies.

4.4.2.2 DailyMotion

In Fig. 4.6a, we plot the CDF of the peak rates for DailyMotion FTTH M flows. First note that a few DailyMotion FTTH M flows achieve peak rates up to $100\,Mb/s$: this indicates that the path between the BAS and the DailyMotion servers is well provisioned and that DailyMotion streaming servers do not limit peak rate. In Fig. 4.6a, we see a difference

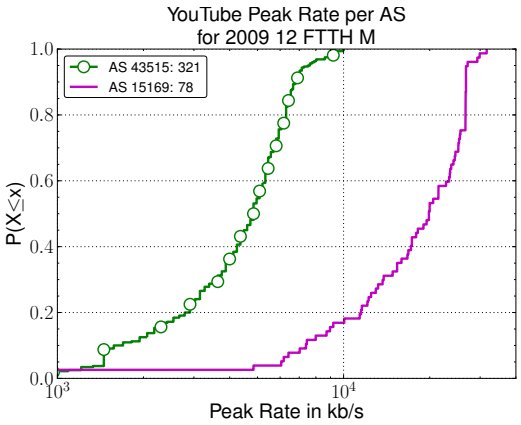


Figure 4.7: YouTube Peak Rates per AS for the FTTH M 2009/12 trace

in distribution policy for 2008/07 trace where most of the flows have peak rates at about $15\,Mb/s$.

4.4.2.3 YouTube

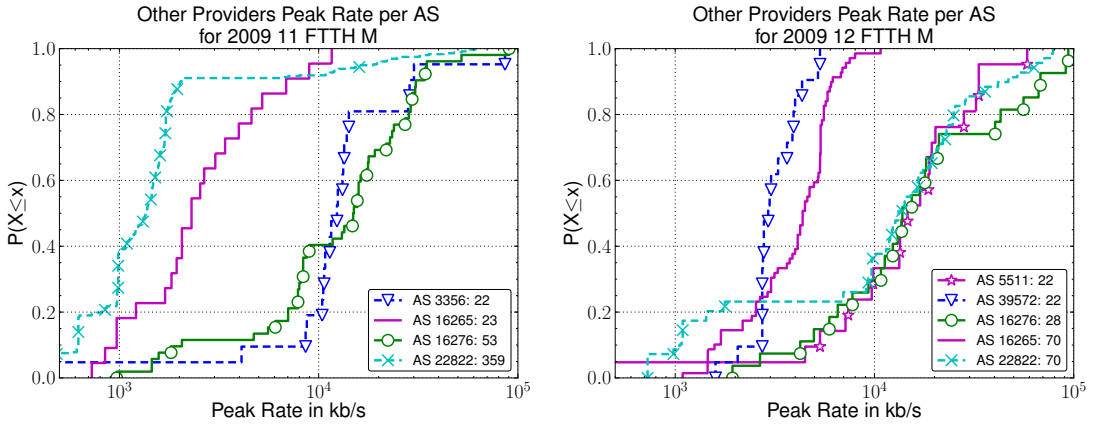
For YouTube in Fig. 4.6b, less than 20% of the flows achieve peak rates just above $10\,Mb/s$, while in the case of DailyMotion more than 80% had rates above $10\,Mb/s$. None of the flows for YouTube achieve a peak rate even close to $100\,Mb/s$.

We would like to show how a server side peak rate limit is influencing the shape of the CDF of FTTH M peak rates. We choose YouTube as their video streams have peak rates that are *always* lower than the peak rates of other streams received by the same clients. We focus on the 2009/12 trace taken in the afternoon when the ISP Backbone is lightly loaded. In fact, the CDF of YouTube peak rates for trace FTTH M 2009/12 is bimodal, so we distinguish YouTube videos per originating AS in Fig. 4.7. This allows us to see that for both, the YouTube and the Google AS, the distribution policy is to limit the peak rate of flows, however, the limit value is different. If the providing AS is Google, we have much higher peak rates (up to $25\,Mb/s$) than if the video comes from the YouTube AS (only up to $8\,Mb/s$). Indeed the convex and bounded shape of the curve in Fig. 4.7 is characteristic for a *server side limit*.

We shall see later in Tab. 4.5 that these server side peak rate limits do not influence video reception quality.

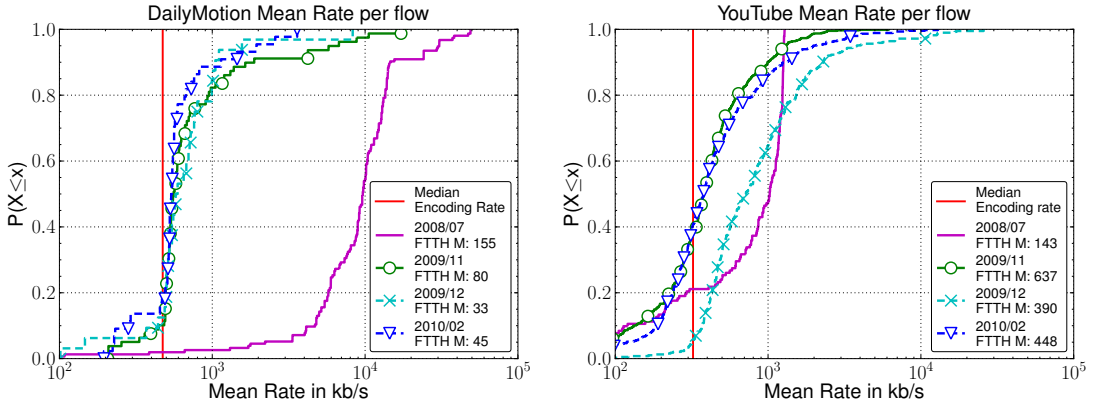
4.4.2.4 Other providers

For other providers (all except YouTube and DailyMotion), we expect a large discrepancy among providers. So we plot in Fig. 4.8 the peak rates per AS and not per video sharing site as these other providers do not often have an internal infrastructure, but rely on third party servers to distribute their videos. We also have a few flows achieving peak rates up to $100\,Mb/s$. Some providers have server side limits whereas other do not have. We can construct a convex index to derive an indication of the distribution



(a) FTTH M 2009/11 **(b)** FTTH M 2009/12

Figure 4.8: Other Providers Peak Rates



(a) DailyMotion **(b)** YouTube

Figure 4.9: Mean Flow Rate of Videos for FTTH M traces

type by comparing ratio of the median and the 90^{th} percentile with a threshold:

$$(\text{Server side limit}) \Leftrightarrow \left(\frac{50^{th} \text{ percentile}}{90^{th} \text{ percentile}} > 0.65\right)$$

4.4.3 Mean Flow Rates

In this section, we focus on the mean flow rate of video transfers, which is defined as:

$$\text{mean flow rate} = \frac{\text{total flow volume}}{\text{total flow duration}}$$

Mean flow rate is an important metric as it is related to the user perceived quality as we will see later (*cf.* Tab. 4.5).

In Fig. 4.9, we plot the CDF of the mean flow rate of YouTube and DailyMotion. We also plot the median video encoding rate for each site to be able to compare the reception rates with the standard encoding rate. As we are interested in server side limitations,

we only plot the mean rates for FTTH M traces, as they are much less likely to be limited by their access speed. Flow mean rates are generally not very high: few videos achieve rates above $1\,Mb/s$ even for FTTH traces.

4.4.3.1 DailyMotion Mean Flow Rates

As for DailyMotion in Fig. 4.9a, we have very homogeneous mean rates in all traces that show a large accumulation point just above the median video encoding rate at $500\,kb/s$, except for 2008/07 trace. Thus, even if there is no peak rate limitation (see Fig. 4.6a), there is a mean rate limit for DailyMotion videos set slightly above the median video encoding rate. While such a choice of the rate limit should allow for a correct reception (and viewing) quality for most of the videos, the reception can be very sensitive to any network problem that may cause the reception rate to fall below the encoding rate for some limited time. In the FTTH M trace of 2008, we see that the mean rate limit originally was higher at about $12\,Mb/s$.

Such modifications in the rate limitation policies made by the video sharing sites are usually not known in advance to the ISP. However, they may have an important impact on the network: unlimited peak rates and moderate mean rates may lead to much more bursty traffic arriving in the router queues.

We would like to emphasize the fact that the peak rate and mean rate limits of TCP flows are independent: indeed, DailyMotion has no peak rate limit (at least up to $100\,Mb/s$) but a strict mean rate limit at $500kb/s$.

4.4.3.2 YouTube Mean Flow Rates

In Fig. 4.9b, we can see that the policy concerning the mean rate limitation of YouTube has evolved over time. For the 2008/07 trace, there is a sharp mean rate limit at $1.2\,Mb/s$ that has been previously observed [41].

Such a limitation of both peak rate and mean rate, as in the case of YouTube, was *most likely* implemented using a well-known open-source rate limiter, the Token Bucket Filter over Hierarchical Token Bucket (HTB [29]) with two buckets (one limiting peak rate and one limiting mean rate). Note that YouTube uses a **new distribution policy** since 09/2010, so the conclusions on YouTube peak and mean rates do not hold any more.

The FTTH M 2009/12 afternoon trace achieves average flow bit-rates superior to the median video encoding rate for 95% of the videos. As for mean rate, the shape of the graphs does not allow to infer any mean rate limitation.

For the traces taken in the evening, around 40% of the videos achieve a mean reception rate that is inferior to the median encoding rate. The curves are concave with no clear limitation. As we will see later (in Tab. 4.5), such low reception rates result in bad reception quality.

In Fig. 4.10, we closer look at the different ASes used by YouTube. There is no indication for server side limitation of the mean rate, as it was the case in 2008. The CDF of the mean rates is concave for all ASes. Even if the shape of the CDF for mean rates is

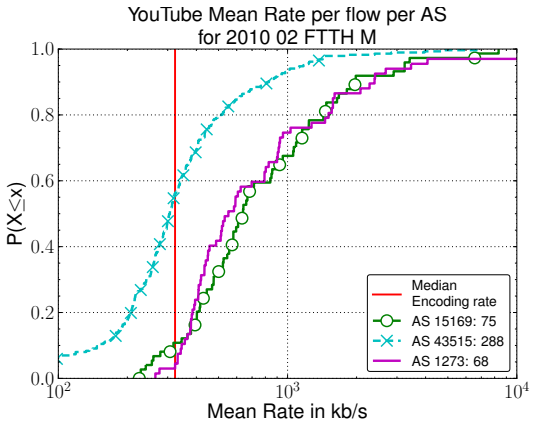


Figure 4.10: Reception Rate of YouTube videos per serving AS for FTTH M 2010/11 trace

similar among ASes, the YouTube EU AS (43515) clearly has lower mean rates for in most of the cases, with close to 50% of the videos achieving a mean reception rate below the median encoding rate. We will come back to this point when we discuss in detail the video quality (see Sect. 4.5.2).

Studying the achieved mean rates has allowed us to understand the distribution policies used by the two main video sharing sites. We have seen that the mean rate is not related to the peak rate, and that videos from the same video sharing site achieve very different mean rates (independently of the ISP policy) depending on the AS delivering the video.

4.4.4 Loss Rate

There are different ways to estimate losses at packet level, depending on *where* the loss happens.

lost event (LST) a packet with a sequence number lower than the previous one;

retransmitted packets, *i.e.* packets carrying a sequence number already seen; for the flows in downstream direction this allows to measure **access loss**;

out of order packets, *i.e.* packets with an unexpected sequence number ($<$ $\min\{seq.\ nb.\}$ or $> \max\{seq.\ nb.\} + pkt\_size$) but not retransmitted; for the flows in downstream direction this allows to measure **Backbone loss**.

burstiness of loss an index to represent if losses occur in a burst or not, defined as:

$$\text{burstiness} = 1 - \frac{\text{Nb of LST events}}{\text{Nb of OOO packets}+\text{Nb of RTM packets}}$$

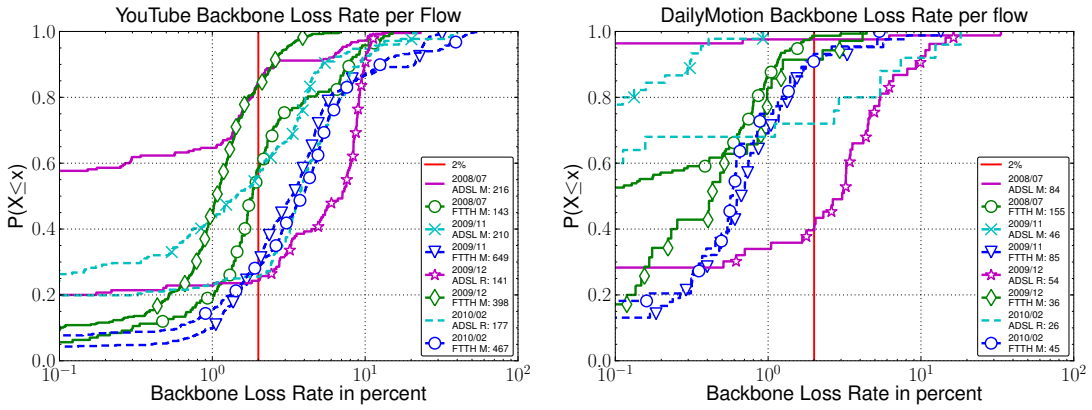
We also count the **unique** bytes received *vs.* the **total** bytes received: it gives an indication of the fraction of the flow affected on the data transfered that may differ from the packet view.

As the probe is located at the BAS level, all packets from/to the customers of the ISP *must* pass through, which ensures that our measures are not biased by multiple paths taken by the packets.

Table 4.4: Explanation of Loss Evaluation with downstream packets

	Pkt. Length	Seq. Nb.	LST§	OOO¶	RTM†	loss location‡
	500	0	–	–	–	
	500	500	–	–	–	
	500	1000	–	–	–	
	1500	1500	–	–	–	
	1500	3000	–	–	–	
	500	**500**	X	–	X	access network
	500	**1000**	–	–	X	access network
	1500	4500	–	–	–	
	1500	9000	–	–	–	
	1500	**6000**	X	X	–	backbone
	1500	**7500**	–	X	–	backbone
	1500	10500	–	–	–	
Total	13000 (total bytes)	12000 (unique bytes)	2	2	2	

§ loss event
¶ out of order packet (but not previously seen) ⇒ backbone loss
† retransmitted packet ⇒ access loss
‡ we do not consider spurious retransmissions or packets not lost but missed by the probe



(a) YouTube **(b)** DailyMotion

Figure 4.11: Backbone Loss Rates

We illustrate the loss estimations with Tab. 4.4. This shows that losses before probe (in the backbone) are caught as out of order, as opposed as packets seen twice that most likely indicate loss after the probe (in the access network).

As for video streaming most of the data are transmitted from server towards the client, we focus on losses between the server and the BAS, which is referred to as Backbone loss. The access loss rates of most of the flows are below 1% (details are not shown here).

In Fig. 4.11, we look at Backbone loss for YouTube and DailyMotion.

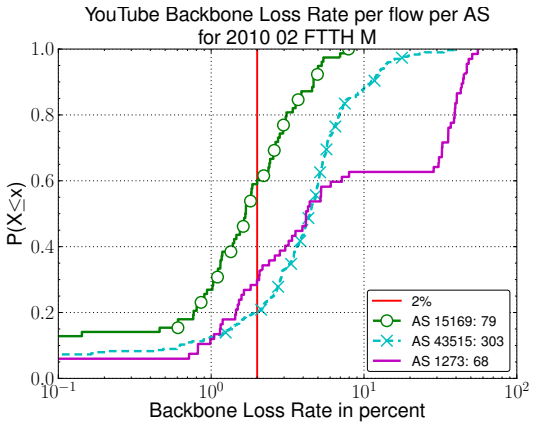


Figure 4.12: Backbone Loss Rates per AS for YouTube videos on 2010/02 FTTH M trace

4.4.4.1 YouTube Loss Rate

In the case of YouTube, it is interesting to understand how the AS connectivity to the ISP can greatly influence the loss rate. The CDF of the **Backbone loss rate**, which is defined as the ratio of the number of packets lost in Backbone to the total number of packets, is shown in Fig. 4.11a. If we focus on the 1% loss region, in all traces (except the 2008 traces and the 2009/12 afternoon FTTH M trace) between $60 - 80\%$ of the flows experience more than 1% packet loss along the path from the server to the capture point. For a TCP connection, the through achieved is inversely proportional the square root of its loss rate. Accordingly, the mean flow rate of all the YouTube flows with more than 1% Backbone loss is only $285\,kb/s$ (including FTTH flows), whereas the median encoding rate is $330\,kb/s$ (Fig. 4.2a).

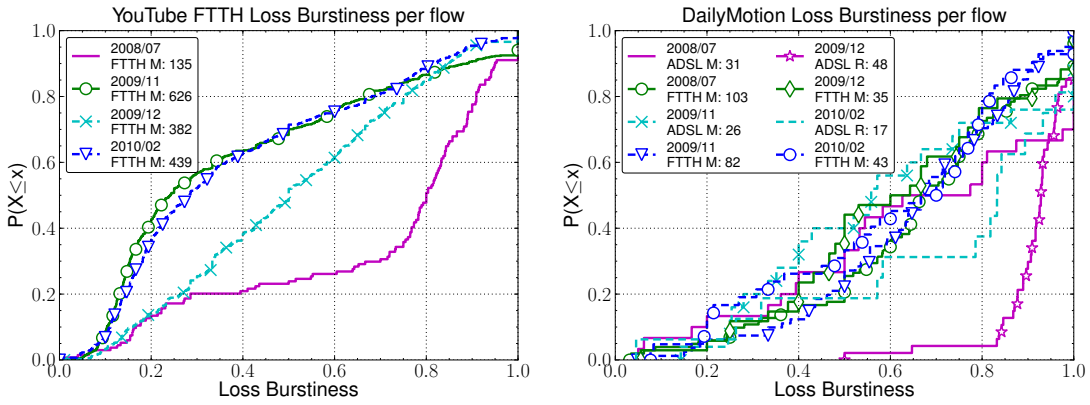
A threshold of 2% on loss rates allows us to discriminate traces in Fig. 4.11a. For example, 2009/12 FTTH M and 2008/07 ADSL M are the only traces where the large majority of flows have less than 2% loss rate. We will see in Tab. 4.5 that these are also the only traces with consistently good reception quality.

Distinction per AS In order to better understand the differences among traces, we distinguish the YouTube flows per originating AS. We focus on Backbone loss in Fig. 4.12 and shall see how losses occur per AS.

In fact, the loss rate of C&W is dependant of the prefix used (not shown here). Thus, we have C&W prefixes with high loss rates (higher than YouTube EU AS) and other with lower one (as Google one). What is clear in this figure is that the 2% out of order packets ratio seem to be a good indicator of *health* of the sending AS: if 80% of the videos distributed by an AS have less than 2% loss, we can assume this AS has no specific problem.

4.4.4.2 DailyMotion Loss Rate

We show in Fig. 4.11b the CDF of Backbone loss rates for DailyMotion. Most flows see less than 1% upstream loss rate. We also plot the 2% loss rate, which is adequate



(a) CDF of loss burstiness for YouTube FTTH M traces

(b) CDF of loss burstiness for DailyMotion

Figure 4.13: Burstiness of Losses

to discriminate DailyMotion videos according to their reception quality. Indeed, both of the ADSL R traces (2009/12 and 2010/02) encounter much more losses (above the threshold of 2%), and we shall see in Tab. 4.5 that these are exactly the traces where a lot of videos experience a bad reception quality.

4.4.4.3 Burstiness of Loss

The loss burstiness is defined at the beginning of the Sect. 4.4.4. An example of burstiness value is when you have only single out of order packet or retransmitted packet at a time, then you have a 0 burstiness index. The lower the burstiness index, the more disperse are losses. A burstiness index above 0.8 means that on average each loss event affected at least 5 packets. We plot in Fig. 4.13 the CDF of loss burstiness for YouTube FTTH M flows.

In Fig. 4.13a, there are two traces (2008/07 FTTH M and 2009/12 FTTH M) that have more burst losses than others. They correspond to the use of former YouTube AS (36561) and to the afternoon trace. As the AS path from the ISP towards YouTube EU encounters congestion problems at evenings, we have a low loss burstiness for evenings traces. Indeed, this kind of losses are happening on a large transatlantic link shared between many flows and two consecutive packets of the same flow are unlikely to be lost. We can check this with the detail of burstiness per AS in Fig. 4.14, where YouTube EU (AS 43515) has a much lower loss burstiness index than other ASes providing YouTube videos.

For DailyMotion in Fig. 4.13b, we have a much higher loss burstiness index than YouTube. As the DailyMotion AS is well connected to the ISP, the flows should attain very high TCP congestion window values. Thus, when a loss occurs, it will probably affect multiple packets in the TCP congestion window, resulting in higher loss burstiness. We mention that traces from the regional BAS encounter the highest burstiness index: in this case, the bottleneck is in the ISP on the path between the interconnection and the regional BAS. On this path, the flows are not multiplexed with other flows leading to bursts losses.

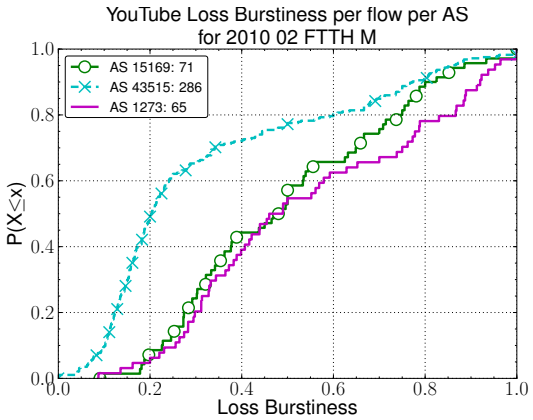


Figure 4.14: CDF of loss burstiness for YouTube FTTH M 2010/02 trace per AS

The burstiness study shows different trends between YouTube and DailyMotion that are due to the cause of limitation:

- interconnection link shared with other traffic for YouTube (less likely to have burst losses in a flow);
- good connection with a direct peer for DailyMotion.

We generalize the analysis of burstiness as follows: if the flows on a set of users towards a specific AS encounter mostly bursty losses, we deal with occasional congestion on a single bottleneck path; otherwise, non-bursty flows indicate a congestion on a path shared with many other flows.

4.4.5 Methodology for Monitoring

Here we try to combine main messages of the previous analysis into a flow chart to be able to derive the state of a monitored platform out of flow level information in Fig. 4.15. In our case, we consider a BAS connecting ISP users to the Internet, but it could be more generic. As the situation can be very diverse, we shall apply this method for each AS individually. Moreover we think it should apply on video streaming only. Indeed, HTTP streaming nowadays aggregate a lot of flows and volume, and other popular applications cannot allow this analysis:

- P2P software usually limits individual flow rates and take too many different path;
- non-streaming HTTP transfers are often short and do not allow TCP to achieve its full rate.

One click hosting sites could also lead this kind of analysis but free users often encounter many limitations on these sites, so we think it should not be as representative.

This method is quite simplistic but gives us clues on how the traffic is handled without being mislead by caveats such as servers intentionally limiting peak or mean rates of its flows.

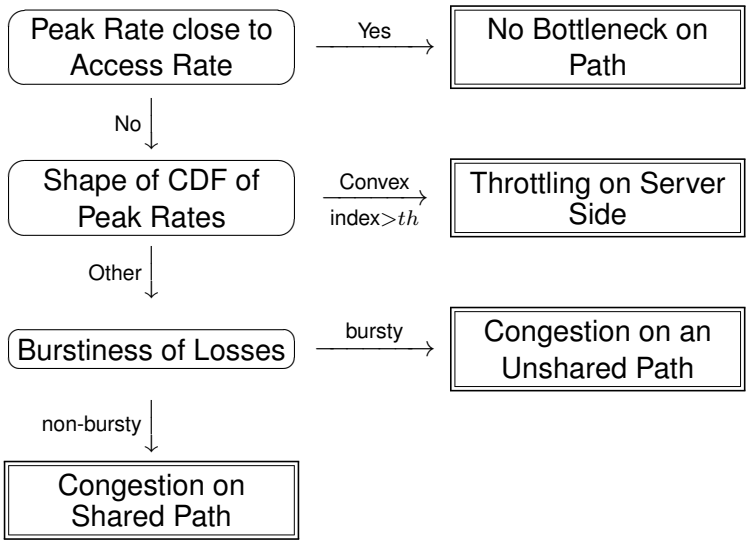


Figure 4.15: Monitoring Diagram: to apply to each streaming provider

Our flow chart starts by comparing peak rates to access rate: if the flows achieve their theoretical maximum rate, no bottleneck has been encountered on the path. Next we figure out throttling on server side out of the shape of the CDF of the peak rates (with our convex index). The burstiness of losses allows us to see congestion on shared path (when there is no bursts) or occasional congestion on a single path (when there are bursts).

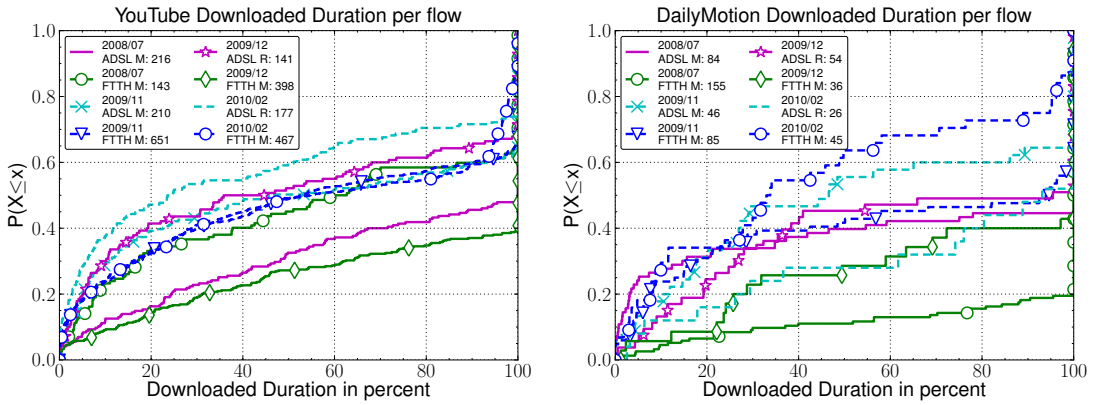
4.5 User Behavior Study

In this section, we want to study how users view videos and also how users adapt their viewing behavior in response to bad reception quality.

Firstly, we globally measure how much of a video the users download. Secondly, we define a simple metric for user experience to differentiate videos with good reception quality from others. Finally, we relate this indicator to the fraction of video downloaded to the fact that a user has completely downloaded the video, and to the video length.

4.5.1 Downloaded Duration

Ideally, we would like to know how much of a video the user is actually watching. However, as the video interactions (like pausing/resuming the video) are not transmitted to the server, we cannot retrieve them at network level. Instead, we approximate how much of a video a user watches by how much of the video she has downloaded, which provides us with an *upper bound* on how much she has watched. For instance, the fact that a video has been fully downloaded does *not* mean that the user watched the video completely (if the video was paused and never resumed).



(a) YouTube **(b)** DailyMotion

Figure 4.16: Fraction of Video Downloaded per Trace

We define the downloaded duration:

$$\text{downloaded duration} = \frac{\text{size of downloaded flow}}{\text{video encoding rate}}$$

We have also checked that **the distribution of video length does not change for our different traces**. Also, the distribution of video durations for the videos watched by the ISP clients matches the size of videos as seen in [25]: the most frequent durations are videos of 3–4 minutes videos (most likely video clips).

In Fig. 4.16, we plot the CDF of the fraction of the video downloaded for each trace for YouTube and DailyMotion. We see that the distribution can vary a lot among traces. Globally, *not more than* $40 - 60\%$ *of the videos are completely downloaded*. Such behavior seems to indicate that progressive download induces users to "browse" videos without necessarily watching each video to the end.

Focusing on YouTube in Fig. 4.16a, we first notice two traces with a much higher fraction of videos that are completely downloaded: the 2008/07 ADSL M and 2009/12 FTTH M, which are the traces with the lowest loss rates (see Fig. 4.11a). In all the other traces, the fraction of videos that is fully downloaded is 40% or less.

As for DailyMotion (see Fig. 4.11b), the difference in download behavior among the traces is more pronounced than for YouTube. We have no good explanation why this is the case, except that there are fewer samples in each trace for Fig. 4.11b. The FTTH M 2010/02 trace has the largest number of completely downloaded videos (80%), which is significantly more than what we have seen for YouTube.

Since many of videos are not downloaded, and thus not watched, until the end, we want to understand the reasons: Is it lack of interest, bad reception quality, or video duration?

4.5.1.1 Download Duration per AS

We can think that this reaction of interrupting the video download may be not directly be due to video characteristics but also to interest in the content. To check this, we plot in

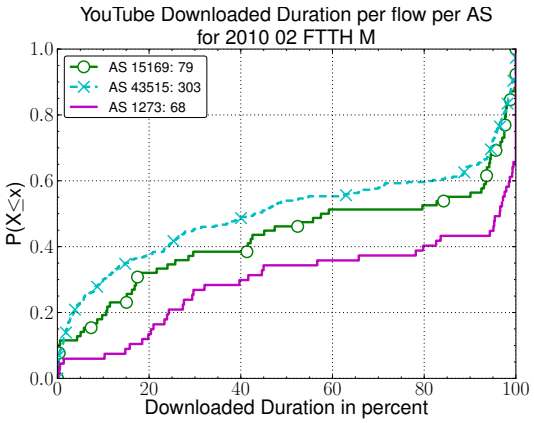


Figure 4.17: Percentage of downloaded volume for YouTube 2010/02 trace per AS

Table 4.5: Fraction of Videos with Bad Reception Quality (normalized rate ≤ 1)

	YouTube					DailyMotion	
Trace	YT EU AS 43515	GOO AS 15169	YT AS 36561	C&W AS 1273	GBLX AS 3549	DM AS 41690	LL AS 22822
2008/07 ADSL M	–	1%	7%	–	–	4%	49%
2008/07 FTTH M	–	–	18%	–	–	1%	3%
2009/11 ADSL M	49%	0%	–	47%	50%	11%	–
2009/11 FTTH M	34%	0%	–	88%	5%	12%	–
2009/12 ADSL R	74%	50%	–	–	–	30%	–
2009/12 FTTH M	6%	10%	–	–	–	15%	–
2010/02 ADSL R	68%	45%	–	56%	80%	20%	–
2010/02 FTTH M	52%	17%	–	1%	69%	8%	–

Fig. 4.17 the fraction of downloaded volume for trace FTTH M 2009/11 on YouTube with each AS separated. In this graph, we see that the downloaded size clearly depends on the origin AS, but the shape of the CDF is the same for each AS. If we relate this to the loss rates in Sect. 4.4.4.1, we have a good match between the AS providing high flows throughputs and the AS where users do not fully download the video.

As the load balancing of DNS resolution is user independent[6], we conclude that the impact of individual video degradation is instantaneously applied: if a user is not able to watch a video correctly, he will interrupt it independently of previous videos watched.

4.5.2 Simple User Experience Metric

To “measure” the user experience we want to know if the video was interrupted during playback. First, we define the normalized rate for each video as:

$$\text{normalized rate} = \frac{\text{mean flow rate}}{\text{video encoding rate}}$$

[6] 2 requests on the same video cache URL from the same user can result in 2 different addresses (without DNS cache).

In lack of a better metric, we say a video has *good reception quality* if its normalized rate is above 1, and *bad reception quality* otherwise. We admit that this is quite a crude measure. However, we have done several controlled lab experiments under different network conditions. We have recorded both, the packet traces and the occurrences of video playout interruptions, and have found that the normalized rate is a reasonable indicator for the reception quality. We report in Tab. 4.5 the video quality for YouTube and DailyMotion per AS streaming the video.

What is striking is that for YouTube, the reception quality depends a lot on the AS that serves the video. Many videos coming from the YouTube EU AS (43515) have a bad reception quality. If we relate this to the traffic distribution given in Tab. 4.2, we see that the AS that serves most of the YouTube videos for this particular ISP is the one providing the worst performance. YouTube videos coming from other ASes usually have a good reception quality.

The afternoon trace FTTH M 2009/12 is the only one with a good reception quality for streams served from the YouTube EU AS. This makes us conclude that in the evening hours there are not sufficient bandwidth resources along the path from the YouTube EU AS to the ISP.

In the case of DailyMotion, the reception quality among traces is much more uniform. In the 2008 ADSL trace, the LimeLight AS (22822) had much lower reception quality than the DailyMotion one. For DailyMotion, the time of day has no impact on reception quality as the afternoon FTTH M 2009/12 trace does not have better performance than the other FTTH traces.

The case of the two ADSL R traces is worth considering separately: for both traces, the reception quality of a large number of videos coming from either the YouTube or from DailyMotion is bad. As this affects videos being served by the other ASes, these seems to indicate that some of links internal to the ISP are congested.

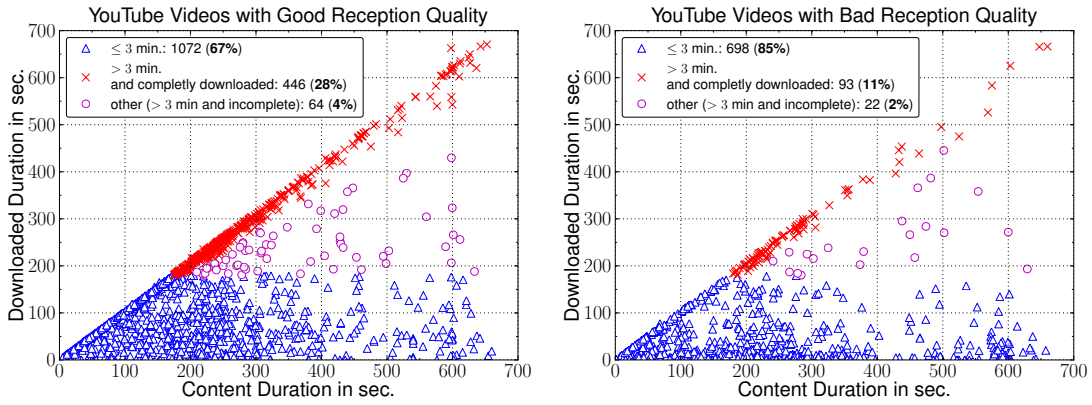
4.5.3 How do Users watch Videos

In this section, we want to understand why users decide to interrupt video downloads. We first analyse the downloaded duration in function of content duration. Then, by discriminating on the reception quality, we are able to see that videos with bad reception quality have much shorter downloaded durations than others. Moreover, the decision of interrupting the download is taken very quickly for videos with bad quality.

4.5.3.1 Relation of Video Length to Reception Quality

In Fig. 4.18, we plot for each video the fraction of the video downloaded in function of the video length.

We see that videos with good reception quality have more complete downloads than videos with bad reception quality. To analyse the graph, we first look at the videos that have download durations of less than 3 minutes, which make up the majority of the videos, independently of the reception quality. The usage for YouTube seems to be either to download less than 3 minutes or to download the video completely. In case



(a) Good reception quality videos **(b)** Bad reception quality videos

Figure 4.18: Fraction of Video Downloaded in function of Video Length for YouTube

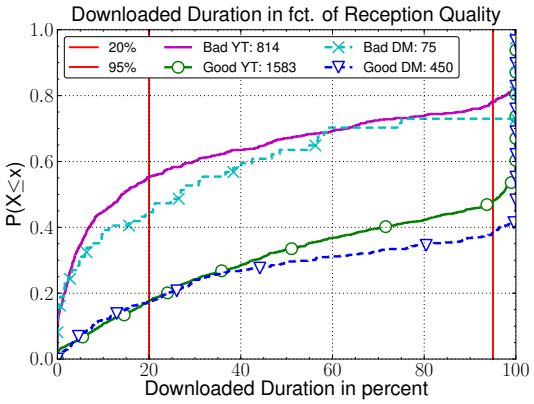


Figure 4.19: Fraction of Video Downloaded as function of Video Reception Quality

of good reception quality, about three times as many videos longer than 3 minutes are fully downloaded (28%) as when the reception quality is bad (11%).

4.5.3.2 Relation of Download Duration to Reception Quality

As far as the downloaded duration is concerned, we can see in Fig. 4.18 that in case of good reception quality, 34% of the videos have a downloaded duration of 3 minutes or more, while in the case of bad reception quality their share drops to only 15%.

In Fig. 4.19, we relate the *fraction* of the video downloaded to the video quality. Again, we clearly see the impact of the reception quality on the downloaded portion of the video. The results for both sites, YouTube and DailyMotion are very similar. We have clearly two zones:

- completely downloaded videos ($\geq 95\%$);
- videos for which only a small portion has been downloaded ($\leq 20\%$).

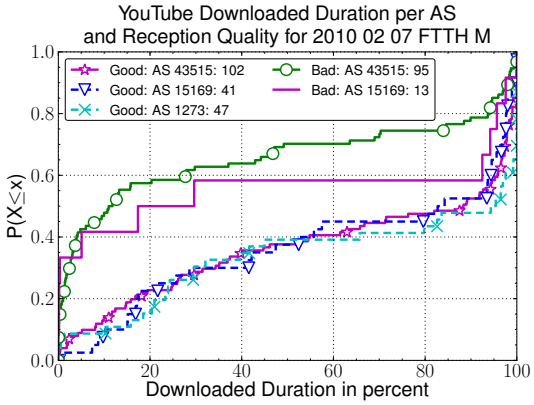


Figure 4.20: Fraction of Video Downloaded as function of Video Reception Quality for Trace FTTH M 2010/02 for YouTube ASes

In case of bad reception quality, very few videos are completely downloaded. Moreover, the decision to stop downloading a video is taken quickly (in the first 20% of the video duration).

In case of good reception quality, about half of all videos are completely downloaded and the decision to interrupt download is not taken right from the beginning but at any point during the viewing.

We have seen in Tab. 4.3 that in the case of YouTube, the same user, when requesting multiple videos, will be served with high probability from machines that are located in different ASes. This observation leads us to carry out one more analysis in order to validate that there exists a positive correlation between video reception quality and the fraction of the video downloaded.

For the FTTH M 2010/02 trace, we take all clients that meet either one of the following two conditions: clients having received at least one video with good reception quality *(i)* from both AS 43515 and AS 1273, or *(ii)* from both AS 43515 and AS 15169.

In Fig. 4.20, we plot for all the clients that meet condition *(i)* or *(ii)* the fraction of video downloaded as function of the reception quality for the three different YouTube ASes. We see that independently of the AS that serves the video, the fraction of the video downloaded is much higher for videos with good reception quality than for bad videos.

CHAPTER 5
Conclusion of Part I

"There are three kinds of lies: lies, damned lies and statistics."

Mark Twain,
Autobiography

Part I has explored different types of passive measurements to draw insights on the traffic of residential Internet customers. We have used data collected on multiple local platform aggregation point from different geographical locations in France. In Chap. 3, we have used connection summaries over a timespan of one week and on three different probes to give an updated view of the components of the Internet traffic of residential users. Chapter 4 has taken advantage of this knowledge (mainly the predominance of Streaming traffic) to focus more deeply into the performance of HTTP video streaming, and to understand the impact of the quality of service on the usage of video streaming. In the remaining of this chapter, we recall the main results of Part I.

5.1 Conclusions on the Analysis of Week-long Connection Statistics

In Chap. 3, we have refreshed the view of residential Internet traffic with connection summaries from one week of July 2011 on three different probes in France. We have first detailed the global characteristics of our data to show that top applications are Streaming, Web, Download and P2P in this order. The composition of Streaming traffic shows that most of the flows come from advertisements whereas most of the volume comes from video clips, also YouTube is by far the most popular video streaming service.

Studying traffic indicators at user's scale has allowed us to understand how main applications function. In particular, P2P transfers usually take advantage of plenty of parallel connections (with low throughput) to achieve an acceptable global download rate. This is true for some users, but overall the P2P throughputs are very low even if aggregated

among all the parallel connections. We have also seen that some ISPs tend to loss much more P2P packets than others.

The application mix of users has been studied through a clustering analysis on top 100 users. Most of these users (that carry about the half of the total platform volume) are grouped in the class of Web+Streaming. This means the large majority of all platform users (with the notable exception of the first 20 heavy hitters) are using only these two applications. Moreover the timescale of the analysis plays only a minor role in the conclusions of the clustering analysis: only taking late night hour snapshots of traffic leads to different conclusions than the whole week analysis.

We finally have come to the question of platform dimensioning. Here, the problematic is how to manage the peaks of downstream traffic. With this in mind, we have proposed some ways to reduce the peaks without perturbation for the end users. The global method is to limit the rate of elastic applications (consisting of the traffic that is not interactive). An interesting result is that we can achieve a reduction of about 4% of the 95 percentile of downstream peaks by reducing the rate of these applications during only 4 hours per day and for only 1 user out of 10. These results may not be directly applicable, but can play a role in local platform management.

5.2 Conclusions on the Performance of HTTP Video Streaming

In Chap. 4, we have carried out a detailed analysis of HTTP video streaming based on the actual videos downloaded by the clients of an ISP. We rely on eight different traces captured over three years, at two geographically different capture sites and for two different access technologies and considered two of the main video sharing sites.

We underline the fact that the video sharing sites have a major control over the delivery of the video and its reception quality through DNS redirection and server side streaming policy. Also, the AS chosen to serve the video is often not the one that assures the best video reception quality.

5.2.1 YouTube Architecture and Video Servers Selection

YouTube videos can be served by machines located in different ASes. YouTube streaming servers enforce a peak rate limit whose value depends on originating AS. In recent years, there seems to be no limit on the mean rate.

The selection of the machine serving the video is done via DNS and under the full control of YouTube. Contrary to the findings of previous studies, YouTube server selection does not seem to apply the usual metrics such as proximity. For a client, the choice of the AS can have a big impact on the reception quality. In the case of the YouTube EU AS, which serves most of the videos in our data, the RTTs and loss rates are high and the reception quality of many videos is bad.

5.2.2 DailyMotion Delivery Policy

In our data, DailyMotion imposes a mean rate limitation that is slightly above the median encoding rate, but does not limit the peak rate.

Since 2009, there is only one AS that serves all requests assuring for most of the videos a good reception quality.

5.2.3 Users' Viewing Behavior

This study is the first to look at the user viewing behavior and the influence of the reception quality on the viewing behavior. We use the normalized reception rate as a simple indicator of video reception quality. We see that videos with bad reception quality are rarely fully downloaded and that bad reception quality results in reduced viewing durations. What is equally interesting is that even when the reception quality is good, only half of the videos are fully downloaded, which indicates that both, the reception quality and the interest in the content, impact the fraction of the video downloaded.

5.2.4 Next Steps on Utilising Passive Packet Traces to Understand Video Streaming Traffic

In the future, we want to further explore how the different flow metrics, such as RTT, loss rate, or mean reception rate can be used to detect and locate performance problems (Fig. 4.15). This approach is similar in spirit to the one in [23] where certain TCP flow records on mobile data networks are used to estimate achievable download speed.

Capturing traces from several video sharing sites will allow to detect problems of a particular site by both, comparing measurements made at different points in time and also by comparing measurements for different video sharing sites made at the same time.

We have used the normalized rate to assess the reception quality of a video. This indicator is a first approximation of the quality of experience of video streaming. The next step is to precisely study the playout behavior of the Adobe Flash Player to "emulate" the video playout and to detect buffer underflows that cause playout stall events. Using this type of study seem not to scale large scale measurements. We shall detail in Part II our model to emulate the video playout for YouTube using active measurements. This will help us validate the indicator of video reception quality used in Chap. 4, and understand in which case this indicator is useful.

Part II

Active Measurements

CHAPTER 6

Active Measurements Context and Challenges

Savoir, et ne point faire usage de ce qu'on sait, c'est pire qu'ignorer.[1]

Alain,
Propos sur l'éducation

We shall now address the question of active measurements. The main concern with active measurements from an operator point of view is: "Do I really measure what my customers see?" Indeed, the traffic we generate may have a different behavior than the one of the users: it can take a different path, the access rate may be different... Also inserting data directly in customers access lines is not an option. These limitations justify why we have deeply developed the passive measurement methods in Part I. But to gain new understanding of the target infrastructure, to be able to modify specific parameters (such as DNS), to undertake reproducible measures, to be able to compare results between ISPs, *etc...* active measurements are a very useful tool. As explain in Sect. 6.1, we shall focus only on HTTP video streaming and more precisely on YouTube, the most popular video streaming sharing site.

6.1 Active Measurements of HTTP Video Streaming

As seen in Part I, web-driven contents carry most of the bytes of residential Internet traffic with video streaming being the most important (in terms of volume). Among the different online video services, HTTP video streaming (using Flash technology) is the most popular one, and YouTube is by far the most popular service. Services such as blogs and social networks are also enabling users to embed personal videos, thus expanding the video streaming audience. As the users usually watch HTTP video streaming while downloading (it's also named *Progressive Download*), the impact of quality degradation is directly perceived. It is thus of primary interest for ISPs.

[1] *Knowing, and not using what we know, is worse than ignoring* – my translation

In this part of the thesis, our goal is to understand the YouTube distribution policy and its impact on Quality of Experience (QoE) from an end-users point of view. For this purpose, we have designed an active measurement tool to evaluate QoE of YouTube videos with the number of stalls in the video as a primary indicator. The purpose of our work is to shed a new light on YouTube video delivery policies and its infrastructure.

This chapter is organised as follows: we first review related work in Sect. 6.2; and in Sect. 6.3, we state the main results and the novelty of our work. In Chap. 7, we present our tool and methodology before explaining our results and the knowledge gained on the YouTube infrastructure.

6.2 Related Work

We have already detailed the passive measurement works on YouTube in Sect. 2.3. The user experience and the impact of network performance on user behavior has been studied in [20, 41, 39] based on packet traces captured at an ISP. The main results are that usually default configurations are used and users often jump inside the videos. With good network conditions, this may lead to a large amount of wasted bytes (downloaded but not watched). On the user side, the response to deteriorated network performance is to shorten their video watching sessions.

We focus in the rest of this section on the active measurement studies of YouTube.

In [46], PlanetLab nodes are used to probe and compare the server infrastructure of three different HTTP video streaming services (including YouTube). The authors use the time to download the first MByte of video as their primary performance indicator. They investigate the service delay distribution according to geographical location of users and to the characteristics of the video (age and popularity).

PlanetLab active measurements are also used in [3] to understand the dynamics of YouTube video server selection by studying the mechanisms of load-balancing (static, semi-dynamic through DNS and dynamic through HTTP redirect). In [1], the same authors pursue the investigation of the YouTube infrastructure, and give many insights on the YouTube video cache server hierarchy.

The impact of DNS resolvers has been compared in terms of latency and caching [4] (not specifically to YouTube). Application level monitoring for ISPs (a goal that we also share) has been studied in [49], with applications such as quality evaluation but also routing policy management.

The study of YouTube QoE has been undertaken with a crowdsourcing approach in [28]. The paper shows that the primary QoE factors in YouTube video playback are the number of stalls followed by their duration. In [48], an estimator of YouTube video QoE has been designed to predict future stall events.

6.3 Contributions

6.3.1 Main Results

We have performed many measurements over one year with different configurations. In order to compare results between ISPs, we have taken advantage of a multi-connected lab with Internet accesses provided by different ISPs but with similar access rates. We have completed these findings with some specific European measurements. We also benefit from simultaneous measures taken by a class of Kansas City students that allows us to compare the distribution policies between Europe and the US.

The main results are that video delivery policies vary a lot even inside the same country and for the same geographical location. The traffic distribution is highly dependent on ISP and this can impact the end-user QoE. Finally, the network and server load-balancing policies dictates the choice of the cache site used by YouTube to deliver the video, whereas the geographical location is not as important. We have also seen that these policies are very volatile and can change abruptly even at the timescale of several months.

6.3.2 Novelty of our Work

Our work differs from others since we are not only interested in the network performance to access the YouTube video streaming servers, but also in the perceived video playback quality. One work towards these goals asks for manual user feedback via crowd-sourcing [28]. In this perspective, one of the main interests of our study is to automatically present an objective measure of video playback quality without having to ask users for their feedback.

The difference in our dataset and PlanetLab based measurements is also important. Indeed, we focus on residential accesses that have very different characteristics than PlanetLab accesses (often hosted by large Universities or Research Centers). This leads to different treatment in the YouTube video delivery. The diversity of ISPs in our data allows us to show that the delivery policy (mainly video server selection Sect. 7.3.1) highly depends on ISP. Finally, many changes observed during the duration of our measurements (almost one year) show that the YouTube infrastructure is highly dynamic even at the limited timescale of several months.

CHAPTER 7

Impact of YouTube Delivery Policies on the User Experience

This chapter presents an in-depth study of YouTube video service delivery. We have designed a tool that crawls YouTube videos in order to precisely evaluate the quality of experience (QoE) as perceived by the user. We enrich the main QoE metric, *the number of video stalls*, with many network measurements and use multiple DNS servers to understand the main factors that impact QoS and QoE.

This tool has been used in multiple configurations: first to understand the main delivery policies of YouTube videos, then to understand the impact of the ISP on these policies, and finally to compare US and Europe YouTube policies.

Our main results are that:

(i) geographical proximity does not matter inside Europe or US, but link cost and ISP-dependent policies do;

(ii) usual QoS metrics (RTT) have no impact on QoE (video stall);

(iii) QoE is not impacted nowadays (with good access networks) by access capacity but by peering agreement between ISPs and CDNs, and by server load.

We also indicate a network monitoring metric that can be used by ISPs to roughly evaluate the QoE of HTTP video streaming of a large set of clients at a reduced computational cost.

7.1 Methodology

The ability to measure QoE of HTTP video streaming is important as it represents a large part of Internet traffic. Passive monitoring can be used to easily monitor a large set of users, but in this case the perturbations between the probe and the end-user are not taken into account in the analysis. Moreover as the video data transfered during HTTP video streaming can become huge, large scale passive monitoring would need

too much processing. Therefore we have chosen to monitor the HTTP video streams directly from the end-users computer.

The interruptions to video playback can be attributed only to insufficient network conditions in HTTP streaming. Indeed, the video quality only depends on the encoding: once the definition has been chosen (by default on YouTube: 360×640), no other image degradation is possible (*e.g.* pixeling). Thus we have chosen only to decode the timestamps of the Flash Video (FLV) frames: this allows us to have a precise evaluation of video time without the cost of video decoding. With this information, we have **reverse engineered the YouTube flash video player** to model its behavior.

7.1.1 Tool Presentation

We have designed a tool, `Pytomo` [43], to measure QoS and QoE of YouTube videos. Our tool functions as follows: After a bootstrapping phase, where we collect the URLs of the most popular videos of the week, we process each URL as follows:

(i) retrieval of video server URL;

(ii) DNS resolution to obtain the IP address of video server;

(iii) QoS statistics collection;

(iv) QoE statistics collection.

Videos related to the current video (obtained through YouTube API) are then added to the list to crawl.

7.1.1.1 Network Statistics

Our tool collects the following statistics (see [30] for a detailed description):

- Ping statistics: min, max, average (over 10 packets);
- Video information: format, duration, length, mean encoding rate;
- Download statistics: average throughput, initial throughput (over the first 3 seconds), maximum instantaneous throughput (over a TCP read).

These network statistics are collected per video for each IP address of the video servers.

7.1.1.2 Model of Video Playback

The goal of this model is to be able to detect and count interruptions in the streaming video playback. Indeed a large scale QoE study on YouTube quality [28] has shown that **interruptions (stall) in the videos are the main Quality of Experience (QoE) indicator for video streaming**. Indeed, at the time of writing, seamless video rate adaptation is not available on main video streaming sites, such as YouTube. Thus, the

only way to cope with reduced network throughput is to wait for more data to come. To model the streaming video playback, we maintain two metrics:

$D(t)$: seconds of video content downloaded up to time t *ie.* the amount of video that has been downloaded in terms of playback duration (obtained through the timestamps of FLV tags);

$P(t)$: seconds of the video consumed up to time t *ie.* the amount of video that has been watched.

These two timescales correspond respectively to the gray and red bars in the YouTube player. Obviously when the red bar corresponding to the playback is getting close to the gray bar corresponding to the downloaded video, the playback is interrupted. Thus, we have:

$$D(t) - P(t) < \text{minimal-playout-buffer} \Rightarrow \text{Playback stops}$$

The restart of the playback occurs when there is enough amount of video that has been downloaded:

$$D(t) - P(t) > \text{minimal-restart-buffer} \Rightarrow \text{Playback resumes}$$

By keeping track of the state of the playback, we are able to infer the number of interruptions during video playback. This model does not take into account jumps inside the video or playback pauses initiated by the user. We are aware of these limitations and think this model should reflect the *usual* user behavior. Moreover, in case of a jump inside the video, the model is still valid: as shown in [41], a jump in a part of the video that is not already downloaded creates a new connection starting at the requesting time (instead of beginning).

We collect the following statistics for the video playback:

- playback statistics: initial buffering duration, number of interruptions, total buffering duration, seconds buffered at the end of the download.

We explain in Sect. 7.1.2 how we obtain the values for the model.

7.1.1.3 Design Implications

As we start our crawl with the most popular videos of the week (by default), we are biased towards popular videos. This is a deliberate attempt so as to assess the QoS and QoE for the content that most users watch. In [1], the authors show that *cold* (unpopular) videos are much more likely to encounter HTTP Redirect, mainly due to cache miss in the video datacenter. This implies that in our case, HTTP Redirect should be due to high video server load (and not cache miss).

7.1.2 Validation Process

In order to be able to obtain reliable results, the validation and calibration of our tool has been carefully undertaken. We use a local server to deliver the video so that we completely control the video delivery during the calibration process. We thus simultaneously launch a video download with our crawler and a video playback in a browser.

The video is delivered by our local server: note that we had to use a proxy for the video player in the browser, since the domain security parameters in the YouTube Shockwave player do not allow queries on other domain than `youtube.com` (such as `localhost`). We have visually checked that the playback as modeled in the tool correspond to the playback in the browser. The total control of the video server allows us to precisely check the threshold values for the model (see 7.1.1.2). Here are the main values found:

- seconds of video content initially buffered (*initial buffer*): 2.0 seconds;
- seconds of video content needed to continue playback (*minimal playout buffer*): 0.1 seconds;
- seconds of video content needed to resume playback (*minimal restart buffer*): 1.0 seconds.

These values agree with the ones chosen in [48] to infer the video quality based on browser events. Decoding the FLV timestamps has allowed us to determine precise values of these parameters.

7.2 Datasets Details

Our tool is able to run on any PC with minimal setup, so that we were able to run it in very different situation.

7.2.1 Volunteer Crawls

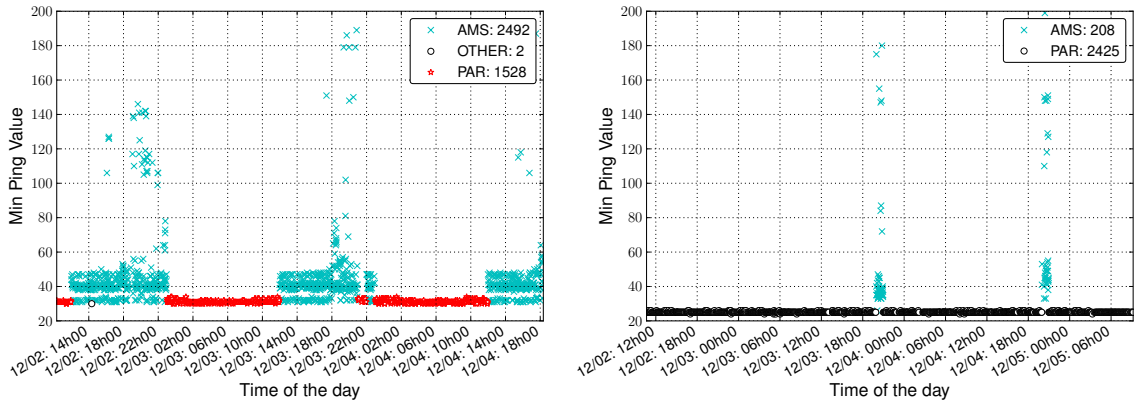
We have a large number (145) of volunteer crawls done by ourselves and many colleagues and friends in Europe and the US. This has allowed us to first test our tool, and then to have many different vantage points for analysis. Theses crawls started in March 2011 and are still in use at the time of writing (February 2012). Their durations vary from a few hours to many days.

They are used to test the tool and understand the YouTube functioning system.

7.2.2 Controlled Crawls

We have also benefited from a set-up in a single location connected to different ISPs that provides 7 ADSL, 1 Fiber and 1 Cable Internet accesses. Note that the ADSL accesses have exactly the same access bit-rate. These controlled crawls have been useful to launch specific tests across multiple ISPs with comparable setup: only the ISP is changing (geographical location and bit-rate are the same). We focus on 8 crawls of at least two days that have been done between September 2011 to January 2012. We present data from only one crawl in September 2011 and one in December 2011. Indeed, these are the only two datasets where a significant amount of video stalls occurs. This indicates that

- the quality of YouTube videos is highly dynamic;



(a) SFR-V4 **(b)** Free-V5

Figure 7.1: Ping time in Milli-seconds to Main YouTube Cache Sites observed in a controlled crawl in December 2011

- overall at the time of measurement, only few videos stalls were observed.

They are used to compare YouTube's policy with respect to the ISP without any access link difference.

7.2.3 Kansas City Crawls

Finally, a complete class of UMKC students were assigned to run simultaneous crawls from their homes on the 8th December 2011 for two hours. These 70 Kansas City crawls have given us some insights on the US market. For YouTube, US represents about 15% of YouTube traffic and 28% of YouTube users.

They are used to compare the findings from European and the US crawls.

Note that the instantaneous throughputs recorded in the crawls allow to validate that no access network limitation has been encountered: either by access rate limit, or by excess usage of other applications while using our crawler.

7.3 Results

In this section, we expose the main results from our experiments. We show the impact of the DNS server used and of the ISP on the selection of IP address and video server respectively. We show that these two key components have an unexpected impact on the QoS of video streaming.

7.3.1 Video Server Selection

7.3.1.1 YouTube Video Server URLs

The URL of a YouTube video is usually:

- `http://www.youtube.com/watch?v=XXXXXXXXXXX`
- or `http://youtu.be/XXXXXXXXXXX` (with short URLs).

The YouTube video webpage comprises of multiple parts: the main video in the flash player and all the rest (comments, related videos, ads...). The video played in the flash player comes from another TCP connection. This connection is responsible of the video delivery: our analysis focuses only on the connection towards the video server only.

The URL of the video server is customised according to the IP address of the requesting user. We have listed the main types of URLs in Tab. 7.1 for controlled crawls (in France) and in Tab. 7.2 for Kansas City crawls (in the US). We adopt the same naming convention as in [1]. The 2 main types of URLs in our data are: `lscache` and `nonxt`. They represent primary cache locations of the YouTube infrastructure. A city code is always included in the URL and indicates the preferred location of the YouTube cache site. For some ISPs, a specific URL including the ISP name along with the city code is given: this should direct users to cache sites dedicated to the ISP.

As seen in [1], the mapping of video ID to the URL of video servers is *fixed*. This means that if a video is served by a primary cache site as `...v6.lscache2...` with one ISP, it can be directed to another primary cache site but with the same `v6` and `lscache2` in the video server URL.

In our data, the secondary and tertiary cache locations of the YouTube infrastructure are used only in the case of redirections. Their URLs are of the form:

- `...v[1-24].cache[1-8].c.youtube.com`.

Note that there is also unicast hostnames to directly address physical servers:

- `r[1-24].CITY_CODE.c.youtube.com`.

We encounter these URLs only in the case of redirection.

7.3.1.2 In Europe

The most common form of video server URL is `lscache` as shown in Tab. 7.1 for the controlled crawls. In these crawls from France, the city codes are `par` and `ams` for Paris and Amsterdam respectively. From Tab. 7.1, this preferred location clearly depends on the ISP. Here are the main findings from Tab. 7.1:

- ISP B has all its video server URLs on one cache site (`par08s01`) in Paris;
- ISP N has all its video server URLs with Paris cache site as preferred location, but with two different logical names (`par08s01` and `par08s05`);

Table 7.1: Number of Videos for each ISP according to Regexp on Video Server Url for a controlled crawl in December 2011

URL Regexp	ISP A	B-A	B-F	F-R	F-V	N	O-L	S-E	S-V
o-o.preferred.par08s01.v[1-24].nonxt[378].c.youtube.com	0	1	2	0	0	0	0	0	0
o-o.preferred.ams03g05.v[1-24].nonxt[378].c.youtube.com	0	0	0	0	2	0	0	0	4
o-o.preferred.par08s01.v[1-24].lscache[1-8].c.youtube.com	0	2676	2677	0	0	1890	0	1967	1528
o-o.preferred.par08s05.v[1-24].lscache[1-8].c.youtube.com	1636	0	0	952	2425	799	0	0	0
o-o.preferred.ams03g05.v[1-24].lscache[1-8].c.youtube.com	150	0	0	0	206	0	0	3033	2488
o-o.preferred.orange-par1.v[1-24].lscache[1-8].c.youtube.com	0	0	0	0	0	0	2591	0	0
other	0	0	0	0	0	0	1	0	2

- ISP O has a dedicated cache site (`orange-par1`), and the IP addresses of this site belongs to a specific AS (36040);
- ISPs S and F are directed to cache sites in Paris or Amsterdam with different proportions: about $2/3$ to Amsterdam for ISP S and 10% for ISP F.

This highlights that the customisation of video server URLs is done for each ISP.

The network impact of the location on the ping time is found to be very low. For example, the minimum ping time to Paris video servers is of $23.8\,ms$ and of $28\,ms$ to Amsterdam (because of relatively small distance between the two cities). But the main point is that the choice of the preferred location is dependent of the time of day as shown in Fig. 7.1. This indicates a deliberate choice depending on ISP. Moreover, even if the difference on minimum ping value can be very low, the cross traffic on the path from France to Amsterdam can increase the ping value as high as $200\,ms$. Indeed, Fig. 7.1 shows a large variance in ping times towards Amsterdam video servers. Overall, the average ping time to Paris video URLs is of $25.6\,ms$, whereas it's $53.8\,ms$ to Amsterdam.

7.3.1.3 Impact on QoE

Focusing on the QoE, we show in Fig. 7.2 the average number of interruptions per period of 60 minutes for 2 ISPs during a controlled crawl in December 2011. This demonstrates that the preferred location has almost no impact on interruptions: indeed, in this crawl, Free-V5 access has lots of periods with many videos affected by stalls, while using video servers based in Paris. Whereas SFR-V4 access has no stalls even though being mainly served by video servers in Amsterdam. Indeed, the factor affecting the interruptions is the average throughput and not minor differences in the delay to the server.

At TCP level, a ping time of 200 ms means that 5 TCP windows can be transmitted per second. With a window size of 64 kBytes (minimal value), it leads to a maximum throughput of $320\,kBytes/s$. The average encoding rate of videos in our data is $555\,kb/s$ or about $70\,kBytes/s$. This means that this delay to the video server allows, when there is no congestion (losses), a throughput largely above the one needed to achieve playout without stalls.

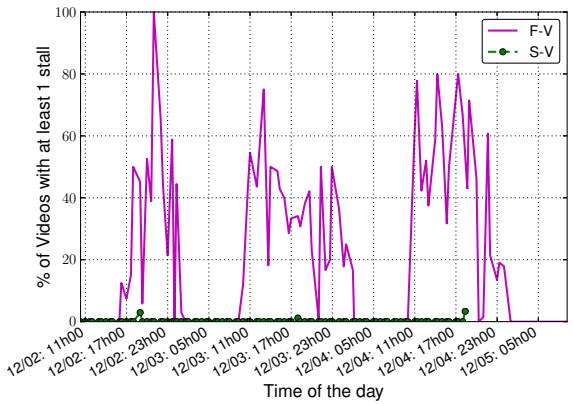


Figure 7.2: Evolution of the percentage of videos with at least one stall over Time (per period of 60 minutes) for two ISPs during December 2011 controlled crawl

Table 7.2: Ping times according to video server URLs for Kansas City crawls

URL Regexp	Location¶	Nb. of samples	Average Ping time
WEST			
http://o-o.preferred.iad09g05.v[1-24].lscache[1-8].c.youtube.com	Washington DC	1439	97
http://o-o.preferred.sjc07s11.v[1-24].lscache[1-8].c.youtube.com	San Jose	446	73
http://o-o.preferred.lax04s12.v[1-24].lscache[1-8].c.youtube.com	Los Angeles	147	75
http://o-o.preferred.iad09s12.v[1-24].lscache[1-8].c.youtube.com	Washington DC	44	60
http://o-o.preferred.sjc07s15.v[1-24].lscache[1-8].c.youtube.com	San Jose	10	61
MID-WEST			
http://o-o.preferred.comcast-dfw1.v[1-24].lscache[1-8].c.youtube.com	Houston	719	50
http://o-o.preferred.dfw06g01.v[1-24].lscache[1-8].c.youtube.com	Houston	308	59
http://o-o.preferred.dfw06s08.v[1-24].lscache[1-8].c.youtube.com	Houston	190	24
http://o-o.preferred.mna-mci1.v[1-24].lscache[1-8].c.youtube.com	Kansas City	71	184
http://o-o.preferred.ord12s01.v[1-24].lscache[1-8].c.youtube.com	Chicago	64	1105
http://o-o.preferred.kanren-lwc1.v[1-24].lscache[1-8].c.youtube.com	Lawrence	50	38
EAST			
http://o-o.preferred.mia05s05.v[1-24].lscache[1-8].c.youtube.com	Miami	660	261
http://o-o.preferred.lga15s20.v[1-24].lscache[1-8].c.youtube.com	New York	89	53

¶ we mention the city corresponding to the airport code inside the URL

In summary, the **video server selection clearly depends on the user's ISP and mainly obeys engineering and load-balancing considerations** rather than *closest source* or similar strategies.

7.3.1.4 In the US

We use the Kansas City crawls to compare the knowledge gained while analysing European data with the US. The large number of URLs of video servers (14) indicates that YouTube allows much more cache sites to be included in the distribution of video in US than in Europe. In Tab. 7.2, for each video server URL regexp we count the number of videos and the average ping time to the servers. Note that we only include the data from the main prefix (/24) for each URL regexp.

In Tab. 7.2, most of the cache sites are located in West and Mid-West region of US. The most frequent location is Washington DC even if it is about twice as far (ping-wise) as Houston. We also have some video servers that are far from Kansas City such

as Chicago or Miami. If we closely look at the ping values, for some cities we have seemingly bizzare results: for example, Houston cache sites can have either about 24 ms or about 50 ms ping values. The reason is that some caches of the Houston sites have IP addresses in the Google AS (15169) and others in the YouTube AS (43515). The path towards these distinct AS can thus be different, resulting in a difference of ping times. Note that in our data, the larger ping times correspond to cache sites in the YouTube AS: this is the case for both Houston and Washington DC.

This validates that the proximity of the cache site plays only a secondary role in video server selection, and that interconnection between ISPs and ASes is a primary factor in network performance.

7.3.2 DNS impact

DNS is often used by CDNs to control the delivery of their contents. Indeed as we shall see, the recursivity of the process allows the requested site to completely control the IP address returned according to the requesting DNS server. YouTube also uses this process to control the distribution of videos on top of URL customisations (as seen in Sect. 7.3.1).

7.3.2.1 Address Resolution Mechanism

We first recall how the DNS resolution works. To retrieve a content on the Internet, one must request it at an IP address. To obtain an IP address corresponding to a URL, DNS servers are used. We use the common term URL but only the domain name part of the URL is mapped to an IP address. Note that with replicated contents (such as YouTube videos) many IP addresses correspond to the same URL[1].

The DNS resolution process is recursive: in order to retrieve the IP address corresponding to `www.youtube.com` URL, the DNS authoritative server responsible of `.com` domain is first queried then the DNS authoritative server for `youtube.com` is queried until the authoritative server for the complete domain is reached.

Note the successive requests are done by the DNS server used by the client: in case of default ISP DNS server, the IP returned by the DNS authoritative server can thus be tailored for the ISP (as we shall see in the rest of this section). In our tool, we have chosen to ask for DNS resolutions on three different servers: default ISP DNS server, Google Public DNS [26], Open DNS [34]. This allows us to clearly see the modifications induced by the choice of the DNS server.

7.3.2.2 URL to IP addresses mapping

The first impact of the DNS server lies in the choice of the IP addresses. We might think that for the same user's location and video server URL, the same IP address should be

[1] In the case of YouTube videos, the same content can even been mapped to multiple cache sites depending on ISP (see Sect. 7.3.1).

Table 7.3: Number of distinct IP addresses obtained with the 3 DNS servers (in percent)

ISP	Alice	Bouygues_ADSL	Bouygues_Fibre	FreeV5	FreeV6	NC	Orange_LB2	SFRV4	SFRV5
1 IP @	0	0	0	0	0	0	0	74	75
2 IP @	0	5	3	0	0	14	9	23	20
3 IP @	100	94	96	100	100	85	89	2	3

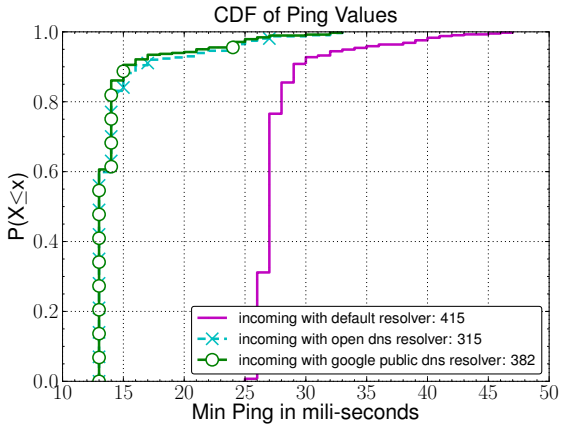


Figure 7.3: Ping Statistics differentiated by DNS for an ISP in Europe in May 2011

returned. This is in fact not always the case. Indeed, we have many crawls from our controlled lab where most of the video server URL were getting 3 different IP addresses depending on the requesting DNS server. In Tab. 7.3, depending on the ISP, we have

- the 3 DNS servers returning the same IP address: Alice and Free (that share the same infrastructure);
- the 3 DNS servers returning 3 different IP addresses most of the time: for SFR;
- a small fraction of URLs resolved on 2 different IP addresses (usually Google Public DNS and Open DNS get the same IP, and the other one comes from the default ISP DNS): for Bouygues, Orange and Numericable.

The difference in IP addresses is not problematic if they are situated in the same data-center or at the same distance to the user. As a representative example of the impact of DNS disagreement, in Fig. 7.3, we show that the ping time towards the IP address corresponding to the URL of the video server can be twice larger with the default ISP DNS server than with Google Public DNS. In this case, they even do not share the same prefix. Note that no general rule can be drawn from this observation: it can vary a lot with time and according to ISP. Also the video quality is not directly related to this QoS measure.

7.3.3 Evaluation of QoE Approximation Techniques

We have tested how *precise* could be an approximation of video stall by an indicator relating download throughput and encoding rate: this could be useful for large scale analysis where complete analysis of the download is not possible. We explore two

metrics: one based on Deep Packet Inspection (DPI) and the other only based on flow level statistics.

7.3.3.1 DPI Metric

We have chosen the same metric as in [39]:

$$\text{reception\_ratio} = \frac{DownloadThroughput}{VideoEncodingRate}$$

A download throughput lower than encoding rate should result in interrupted playback: reception ratio < 1. In this case, one can use a DPI tool to retrieve the video encoding rate from the video streaming flow.

To evaluate the accuracy of this method, we use two standard metrics usually used in classification field (as in [36]). They are based on the concepts of

- *True Positive **TP**:* reception ratio > 1 and the video had no stall;
- *False Positive **FP**:* reception ratio > 1 but the video had at least one stall;
- *True Negative **TN**:* reception ratio < 1 and the video had at least one stall;
- *False Negative **FN**:* reception ratio < 1 but the video had no stall.

Out of these notions, we build these evaluation metrics:

- **recall** = TP / (TP + FN): this corresponds to the fraction of uninterrupted videos correctly evaluated;
- **precision** = TP / (TP + FP): this corresponds to the ratio of uninterrupted videos in the videos with reception ratio > 1.

Here are the results on Free-V5 December 2011 crawl: 91.8% of recall and 88.5% of precision. This means that the reception_ratio based on video encoding rate is quite accurate to determine stall in the videos.

7.3.3.2 Pure-Network Metric

If we further explore the idea of computationally efficient evaluation, we can construct another metric without any DPI phase: we compare the download throughput to the default encoding rate. We have measured this default encoding rate at $555kb/s$ in our data. The metric is thus: simple_reception_ratio $= DownloadThroughput/(555\,\text{kb/s})$.

This leads to the following evaluation of the metric: 28.7% of recall and 100% of precision. This means that this non-DPI metric can surely assess that a video is interrupted, but would class a lot of interrupted videos as good ones.

7.3.3.3 Application of these Metrics

The conclusion of this evaluation is that to **roughly evaluate video streaming QoE**, we can focus on network throughput (instead of parsing all the FLV timestamps) but **a DPI engine is needed** to have a precise evaluation of the encoding rate of the video. Another advantage of this method is that we do not need to be at the end-user side (as the throughput is limited from end to end by TCP): this means it can be applied to monitoring probes high in the network (thus connecting a lot of clients).

7.4 YouTube Infrastructure

From the knowledge gained in Sect. 7.3, we try to understand some of the YouTube infrastructure. In this section, we use the controlled crawls but do not separate data per ISPs as we are interested in the global YouTube infrastructure.

7.4.1 Datacenters sizes

7.4.1.1 URLs to IP prefix mapping

The YouTube video servers with the same /24 IP prefix are usually sharing the same location. In Tab. 7.4, based on data from our controlled lab, we indicate for the main URL regexps the number of prefixes /24 found with the default DNS server of each ISP. First note that we have joined two regexps (`lscache` URL with `par08s01` and `par08s02`) because they share the same prefix. Also the Paris site has fewer prefixes than the Amsterdam site. Moreover, the number of prefixes used in Amsterdam has grown fast in 3 months: from 4 prefixes to 12. An interesting point is that the /24 prefixes are quite dispersed and cannot be merged in larger prefixes. Also, the prefixes are distinct between the URL regexps.

Finally, we have to mention that when the QoS (here ping times) are so small, these differences do not translate in QoE differences. And as seen in Sect. 7.3.1.2 for Free-V5, closer videos servers do not guarantee good QoE.

7.4.1.2 IP address count

In Tab. 7.5, we count the number of IP addresses for each video server URL Regexp. For each URL Regexp, we have exactly 192 different hostnames[2] (also seen in [1]). This means that for Paris datacenter, we have fewer IP addresses (160) than hostnames. Also note the volatility in the distribution of IP addresses: in September 2011 (Tab. 7.5a), the 160 IP addresses were **shared** between the two main Paris `lscache` URL regexps, whereas in December 2011 (Tab. 7.5b), 80 **distinct** IP addresses are assigned to each `lscache` URL regexp.

[2] this corresponds to the whole range of possibilities

Table 7.4: Distribution of IP prefixes (/24) of video servers of all ISPs for controlled crawls

(a) September 2011

URL Regexp	# /24
o-o.preferred.par08s0[15].v[1-24].lscache[1-8].c.youtube.com	1
o-o.preferred.ams03g05.v[1-24].lscache[1-8].c.youtube.com	4
o-o.preferred.orange-par1.v[1-24].lscache[1-8].c.youtube.com	2

(b) December 2011

URL Regexp	# /24
o-o.preferred.par08s0[15].v[1-24].lscache[1-8].c.youtube.com	1
o-o.preferred.ams03g05.v[1-24].lscache[1-8].c.youtube.com	12
o-o.preferred.orange-par1.v[1-24].lscache[1-8].c.youtube.com	2

Table 7.5: YouTube Datacenters sizes according to the Number of IP addresses seen for crawls of all ISPs on each URL Regexp

(a) September 2011

URL Regexp	# IPs
o-o.preferred.par08s01.v[1-24].lscache[1-8].c.youtube.com	160†
o-o.preferred.par08s05.v[1-24].lscache[1-8].c.youtube.com	160†
o-o.preferred.ams03g05.v[1-24].lscache[1-8].c.youtube.com	328
o-o.preferred.orange-par1.v[1-24].lscache[1-8].c.youtube.com	98

† these 160 IP addresses are the same

(b) December 2011

URL Regexp	# IPs
o-o.preferred.par08s01.v[1-24].lscache[1-8].c.youtube.com	80‡
o-o.preferred.par08s05.v[1-24].lscache[1-8].c.youtube.com	80‡
o-o.preferred.ams03g05.v[1-24].lscache[1-8].c.youtube.com	494
o-o.preferred.orange-par1.v[1-24].lscache[1-8].c.youtube.com	130

‡ two distinct subsets of 80 IP addresses

We also have sent ping probes to missing IPs in the range: there is usually no reply to the TCP ping on these IP addresses. This means the **load-balancing used by YouTube** allows us to **cover most of the alive machines and all of the hostnames** of the datacenter even with a 2 days probing period.

As for the distribution of URL regexps, the video server URL clearly depends of the user's ISP. So in December 2011, for some ISPs, distinct subsets of the video servers prefix are used.

Sect. 7.4.1.1 has shown that the Amsterdam site is larger than Paris in terms of prefixes. This is also the case for the number of IP addresses. For the same amount of hostnames, we have much more IP addresses in Amsterdam than in Paris. We also have 50% increase in the number of IP addresses in Amsterdam from September to December 2011.

Table 7.6: Percentage of Redirection per ISP for December 2011 controlled crawl

ISP	Percentage of Redirection
Alice	29.22
Bouygues-ADSL	29.58
Bouygues-Fibre	30.83
Free-Revolution	26.57
Free-V5	25.33
Numericable	30.19
Orange-LB2	12.69
SFR-Evolution	49.02
SFR-V4	45.99

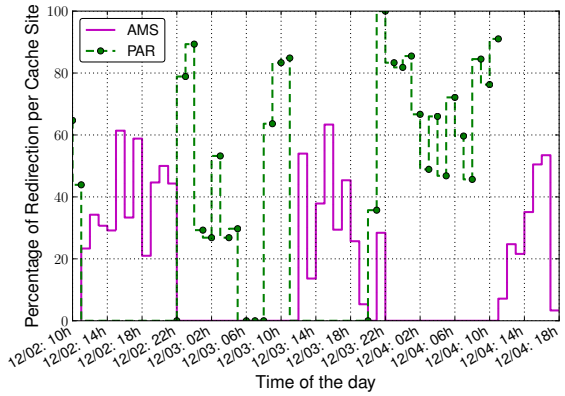


Figure 7.4: Percentage of Redirection (over all videos) per YouTube Cache Site for ISP SFR-V4 per hour

7.4.2 Redirections

YouTube uses redirections to add a level of dynamic load-balancing (on top of the DNS policy and cache site selection according to ISP, which are both centralized). The number of redirections in our controlled crawls is quite variable: in Tab. 7.6, some ISPs can have up to 50% of videos redirected while others have only 12% of them. Moreover, these **redirections occur throughout the day** and not specifically at peak hours. The redirect URLs have mainly a unicast form (see Sect. 7.3.1.1). In our data, the redirections are usually sent to these cache sites: `par` (Paris), followed by `ams` (Amsterdam), `fra` (Frankfort) and `lhr` (London).

In [1, 53], the explanation for redirections lies in the un-availability of the requested video or in datacenter load. Due to our choice of bootstrapping the crawls on popular videos, the chances to have redirections because of cache miss are unlikely. As for the load, the redirections also occur during off-peak hours.

In Tab. 7.6, the repartition of redirection is dependent of ISP. In the case of ISP customised URL (like ISP O), there are only 12% of redirections, whereas in case of ISP S, half of the videos encounter redirections. This is surprising as the cache sites are shared between ISPs. In Fig. 7.4, we plot the distribution of redirection over time for the same ISP than in Fig. 7.1a. The graph does not show any correlation between time of day (peak *vs.* off-peak hours) and the percentage of redirections. Also the redirections

are usually sent to another cache site: in this case, mainly to London (34%), Frankfort (30%), Paris (26%). This means that even if the distribution policy sends a user to Amsterdam cache site, the redirections can send him back to Paris. We conclude that the primary focus of HTTP redirection (excepting cache miss) is to spare YouTube infrastructure.

So this seems to indicate that **the centralised distribution policies (through cache site selection and DNS) addresses the traffic load balancing, whereas the decentralised distribution policy (through HTTP redirects) addresses the server load.**

CHAPTER 8

Conclusion of Part II

The cause of this meeting [of the seer and the seen] is ignorance.

Patanjali (Gregor Mahele's translation), *The Yoga Sutra*

In Part II, we have focused on active measurements of HTTP video streaming. To this purpose, we have presented in Chap. 7 a reliable tool to automatically evaluate the playback quality[1] of YouTube videos as experienced by users. One of the main objectives of this tool is to understand the delivery policy of YouTube and relate it to the DNS resolution policy.

In our study, we have used many volunteered crawls to infer the main delivery policies of YouTube videos. We have completed these crawls with controlled crawls in a specific lab to show the difference of treatment between ISPs for accessing the same service. Finally, we have used many simultaneous crawls from Kansas City in the US to comment on the difference of infrastructure between Europe and the US.

The main findings of our study are that geographical proximity does not really matter inside Europe or the US, but network/server load-balancing and ISP-dependent policies do. Usual QoS metrics (RTT) have no impact on QoE (video stalls). The number of HTTP redirect is quite high in our data, indicating a globally high load on the YouTube video servers. Finally, QoE is not impacted nowadays by access capacity but by peering agreement of ISPs and by server load.

The general conclusion is that YouTube and more generally the CDNs have many ways to control the content delivery:

1. By customising the URL of the video server, which is done by the YouTube front-end servers (Sect. 7.3.1);

2. By resolving the URL of the video server to a different IP address, which is done by the YouTube authoritative DNS server;

[1] which is much more complex than raw throughput measure

3. By using HTTP redirect messages at the video server level, which is done at the cache site level (Sect. 7.4.2).

Note that the HTTP redirect messages usually occur when the server decides not to serve the request (*e.g.* when the server is too loaded): thus, this is a decentralised process. On the contrary, the URL customisation and the specific DNS resolution can be centrally controlled. Thus, we would like to emphasise that YouTube has a large number of knobs to decide from which server and which AS a particular video gets served. From our data, it seems that the primary goal of the video delivery is to use the *best* paths and to spare infrastructure.

Moreover as the routing modifications (for example) are usually not advertised by YouTube to ISPs, this may lead to sub-optimal infrastructure usage. A collaboration between YouTube (and more generally the CDNs) and the ISPs is thus needed to use the Internet at his full potential, and for the benefit of end-users.

From an operational point of view, we have shown that a network metric (download throughput) and a minimal DPI engine (to retrieve the video encoding rate) can lead to satisfactory results in evaluating video QoE of HTTP video streaming. This can be efficiently used by an ISP to monitor from a central point the perceived quality of a large number of clients.

CHAPTER 9
Conclusion

Il y a dans ce livre deux textes simplement alternés ; il pourrait presque sembler qu'ils n'ont rien en commun, mais ils sont pourtant inextricablement enchevêtrés. . . [1]

Georges Perec,
W ou le souvenir d'enfance

In this thesis, we have addressed the problem of Internet measurement from the point of view of an ISP. We have used both passive and active measurements to figure out the components of the residential Internet traffic, but also the behavior of the residential users. For an operator, knowing and understanding the traffic of its customers is of primary importance for planning the evolution of its network according to the needs of the users of this network. We now present the main results and take-away messages of this thesis.

We have first used passive measurements to gain insight on the users' behavior from local ISP platforms. We have shown how the residential traffic is influenced by a small set of users, and also given some ideas for a better management of a local platform. We have mainly focused on the video streaming traffic as not only it represents most of the bytes carried by residential users nowadays, but also as most of residential customers utilize it[2]. The question of the management of a local platform of an ISP has been studied, and we have proposed some innovative ways to reduce the peaks of traffic at reduced cost for the customers. As for video streaming, the influence of traffic characteristics and performance on the usage of this service has been studied and quantified.

We have then developed an active measurement tool to shed a new light in the delivery process of HTTP video streaming. The design of this tool has been strongly influenced by the knowledge gained in the passive measurement phase. This has given us the possibility to discover what is the impact of DNS and ISP dependent policies or video

[1] *There are in this book two distinct texts; it may seem they have nothing in common, yet they are somehow deeply intertwined* – my translation

[2] this was not the case when most of the residential traffic was P2P: at that time, only a small part of the users were generating the vast majority of the traffic volume.

server selection (to name a few) on the quality of experience as seen by users. Then again, this precise evaluation of video quality has allowed us to understand the key factors in the user's perception of the video streaming service. We have thus created and validated efficient ways to measure video streaming quality that can be used at large scale with non-intrusive passive captures to monitor the quality of video streaming customers of an ISP.

Future Work This study opens several perspectives for future work. First, the need of a monitoring infrastructure operating all the time is underlined and new indicators closely following users' expectations are continuously needed. One of the important methodologies developed in this thesis is the focus on a subset of customers (*e.g.* those generating most of the volume) to well understand how they use their Internet access. A clustering method based on the application mix of users has proven to be useful to select a representative subset of customers. The focus on a small set of users can thus help us to figure out global characteristics but also to apply a more precise analysis (possibly using DPI).

We have developed some tools for accurate analysis of video streaming and we expect to refine their use in the future. The ratio of the average download throughput over the encoding rate gives a good approximation of the quality of experience, and we plan to use this indicator to monitor streaming quality. Nevertheless, a larger set of dedicated indicators is needed to monitor all the popular services used by residential Internet customers. With this information, we would like not only to monitor the state of the platform, but also to determine what and where is the cause of a performance problem. We have also shown how performance problems can result in a change in the service utilization of video streaming (Sect. 4.5). We would like to generalize this methods to more services to infer performance problems not detected by usual network indicators.

Finally, we have shown how active measurements can reveal the dynamics of the findings of our monitoring tools. We plan to extend our video streaming tool to other services, so that we can actively and instantaneously follow the evolutions of the video streaming usage. The development of similar tools specific to other important applications is also planned.

Part III

French Summary

CHAPITRE 10

Introduction

10.1 Mesure du réseau Internet

La mesure du réseau Internet peut être effectuée à différents niveaux : depuis l'ordinateur de utilisateur final jusqu'aux routeurs du cœur de réseau. La quantité de données collectée est alors un compromis entre la précision et la capacité de stockage (ou d'analyse) du système. Un vue très grossière d'un réseau peut être donnée par le nombre total d'octets ou de paquets transmis par une interface réseau : c'est typiquement le cas des routeurs transmettant des Giga-octets de trafic par seconde. La mesure la plus précise dans ce contexte est la capture au niveau des paquets réalisée par des logiciels dédiés. La précision de la mesure ne doit pas être déterminée par la capacité de la sonde, mais par la précision d'analyse voulue.

La méthode de capture est aussi importante que les données collectées : *activement* requêter sur un serveur ou dupliquer *passivement* des paquets réseaux sont deux méthodes complètement différentes qui ne répondent pas aux mêmes objectifs. Les techniques actives peuvent être utilisées pour mesurer comment on accède à un service ou quelle est la performance de ce service selon l'installation utilisée. En ce qui concerne les mesures passives, elles sont généralement effectuées à plus grande échelle mais sans la possibilité de personnaliser les requêtes.

Le but de la mesure du réseau n'est pas que de collecter des données, mais surtout de les comprendre. En général, il s'agit d'expliquer les performances d'une connexion Internet. Mais la performance des connections peut recouvrir diverses significations selon le point de vue :

- au niveau d'un routeur, le taux de perte des paquets dans les files d'attentes (indépendamment des connexions) est le principal indicateur ;
- sur un lien de transit, on se focalise sur la charge du lien ;
- pour un utilisateur du Web, le délai pour accéder à son site favori est la principale mesure de satisfaction ;

- alors qu'un utilisateur P2P sera plutôt intéressé par le débit total des ses transferts de fichiers ;
- enfin, un expert TCP peut définir la performance d'une connexion comme le rapport du nombre de paquets dé-séquencés sans retransmission sur le nombre total uniquement dans les périodes de transfert de masse.

Dans ce cas aussi, la définition choisie doit l'être en fonction de l'objectif de l'analyse et pas selon la disponibilité de métriques pré-calculées.

10.2 Le point de vue des opérateurs

Les Fournisseurs d'Accès Internet (FAI) proposent un service *au mieux* : leur but principal est d'acheminer les paquets de leurs clients vers leur destination. De nombreux facteurs ont une influence sur les connexions des clients :

- la capacité du réseau d'accès (mais aussi l'infrastructure de collecte : ATM ou Giga-Ethernet) ;
- le réseau du FAI du point de collecte local vers la destination ;
- la capacité du lien entre le FAI et le prochain AS vers la destination ;
- les politiques de routage de tous les AS traversés par le paquet jusqu'à la destination.

Les FAI ne contrôlent que peu de ces facteurs. Néanmoins, le principal protocole de transport dans l'Internet est TCP qui est un protocole de bout-en-bout. C'est à dire qu'une analyse au niveau paquet d'une connexion (quel que soit le point de mesure) peut donner des informations sur la capacité de tout le chemin et sur la performance vue depuis l'utilisateur final.

La motivation des FAI est de fournir la meilleure performance possible à ses clients pour un coût donné (que ce soit pour sa propre infrastructure ou pour les liens avec les autres AS). L'utilisation de diverses méthodes de mesure et d'analyse, comme présenté dans cette thèse, est donc extrêmement intéressant pour les FAI. Cela peut aboutir à de nouveaux moyens de gérer le trafic que ce soit au niveau des plateformes de collecte ou au niveau de la configuration de TCP en fonction du service demandé.

10.3 Organisation de la thèse

Les deux principales parties de la thèse sont différenciées selon le type de méthode de mesure : *passive* dans la partie I et *active* dans la partie II.

Le contexte de la thèse ainsi que les travaux connexes sur les mesures passives de l'Internet sont présentés au chapitre 2. Cette première partie sur les mesures passives se base sur des données collectées de diverses sources et à des échelles de temps différentes. Au chapitre 3, nous avons analysé des statistiques de connexions sur plus d'une semaine pour trois plateformes ADSL. Cela nous permet de comprendre

quelles sont les applications utilisées par les clients et de mesurer la performance de quatre mille utilisateurs différents. Au chapitre 4, nous avons utilisé plusieurs captures au niveau paquet pour se focaliser sur la performance du Streaming HTTP. Il s'agit de captures courtes et ponctuelles sur tous les utilisateurs de plateformes ADSL durant une période de 3 ans. Cela montre comment les conditions du réseau influencent le comportement des clients regardant des vidéos en Streaming. La première partie est conclue au chapitre 5.

La seconde partie de la thèse détaille les mesures actives du trafic Internet. Le chapitre 6 introduit les défis de ce type de mesure du réseau ainsi que les travaux connexes. Le chapitre 7 présente les données collectées par notre outil de mesure de qualité d'expérience des vidéos YouTube. De nombreux volontaires dans le monde ont participé à la collecte des données, nous bénéficions aussi de données obtenues dans un laboratoire connecté à Internet par divers accès. Toutes ces données nous ont permis de comprendre les principales causes de perturbation dans la qualité ressentie par l'utilisateur. La leçon à retenir est que les coûts des liens et les politiques de routages spécifiques aux FAIs ont beaucoup plus d'impact sur la qualité que les traditionnelles métriques de qualité de service. De plus, la capacité d'accès des réseaux ADSL actuels (et d'autant plus des réseaux FTTH) ne constitue pas le goulot d'étranglement pour l'accès aux services de vidéo Streaming. Le chapitre 8 conclut la seconde partie.

Enfin, la conclusion globale de la thèse est présentée au chapitre 9.

CHAPITRE 11

Principales Contributions

Dans ce chapitre, nous présentons les résultats des analyses décrites dans la thèse. Les sections suivantes (11.1 et 11.2) résument respectivement les parties I et II.

11.1 Mesures Passives

Dans cette section, nous nous focalisons sur des captures passives depuis des clients résidentiels en France. Les données ont été collectées sur une plateforme de collecte locale appelée BAS (Broadband Access Server). Nous avons utilisé des sondes dédiées développées en interne réparties sur toute la France, et avons vérifié que les données sont cohérentes entre elles. Cela nous permet de se focaliser sur un petit nombre de sondes. Nous commençons par analyser une semaine de statistiques de connexions dans la section 11.1.1. Ces données nous permettent d'avoir une vue récente des principales caractéristiques du trafic Internet résidentiel. Dans la section 11.1.2, nous ne considérons que le trafic vidéo Streaming sur des traces paquet durant des périodes d'une heure chacune. Cette étude s'étend sur une durée de trois ans, ce qui nous permet de suivre l'évolution du trafic Streaming de manière longitudinale.

11.1.1 Analyse d'une semaine de trafic ADSL

11.1.1.1 Détail sur les données

Nous avons collecté les résumés statistiques de toutes les connexions sur 3 sondes ADSL en France (à Lyon, Montsouris et Rennes) sur une période d'une semaine (du 5 au 12 juillet 2011). Ces résumés contiennent des informations sur les connexions TCP de tous les clients de ces sondes pour chaque jour de capture. Voici les principaux indicateurs calculés pour notre analyse :

Cnx Id les adresses IP et Ports TCP source et destination, ainsi que les horaires de début et fin de la connexion ;

TABLE 11.1: Détails des traces

	Nb de Clients		Nb de Cnx	
Trace	Total	Moy par jour	Total	Moy par jour
Lyon252	1354	1284	66,231,068	7,788,835
Mont151	1009	951	50,008,566	6,251,070
Renn257	1139	1099	41,320,018	5,165,002

Application reconnue par un outil DPI développé en interne, nous avons accès à l'application, aux web-apps (comme Facebook) et à une partie du trafic crypté eMule et BitTorrent, néanmoins nous nous contenterons de nous référer au *type* d'application (par exemple P2P, Streaming. . .) ;

Volumes le nombre d'octets transférés et le nombre de paquets pour chaque sens de la connexion, nous utilisons aussi le volume maximum transféré par période de 20 secondes ;

TCP Performance nous définissons le numéro de séquence attendu comme le numéro de séquence maximum observé plus la taille du paquet correspondant, ceci nous permet de considérer un paquet comme **perdu** si son numéro de séquence est plus grand que celui attendu, alors que nous le comptons comme une **retransmission** s'il est plus petit ; nous avons aussi à disposition une évaluation du RTT Tous ces indicateurs sont calculés pour chaque sens de la connexion.

Les principales caractéristiques de ces traces sont données dans le tableau 11.1

11.1.1.2 Répartition des applications

Nous résumons dans le Tab. 11.2 la répartition des applications en fonction de leur volume. Notez que nous ne considérons que les classes d'application.

Dans le Tab. 3.3, le Streaming est de loin l'application la plus utilisée pour le volume descendant. Les deux classes d'application suivantes sont le Web et Download avec une part très similaire du volume descendant. La 4$^{\text{ème}}$ application la plus populaire est le P2P. L'ordre est très stable selon les jours ou les lieux différents. Le volume descendant généré par les autres applications est très faible (moins de 10%) par rapport à l'une des 4 premières classes de l'application.

Nous nous focalisons sur la répartition du trafic Streaming dans le Tab. 11.3. Le trafic Streaming est principalement constitué de Clips si l'on se focalise sur le volume descendant (Tab. 11.3a. Néanmoins les publicités et le trafic non-reconnu (probablement des publicités) représentent la moitié des connexions. Ces catégories sont obtenues par *pattern matching* basé sur des URLs de services connus.

Nous avons aussi classé les sites de Streaming dans le Tab. 11.3b en fonction de leur volume descendant. Le volume montant est très faible pour le Streaming de manière générale. YouTube représente plus d'un cinquième du volume descendant total, et le deuxième ne génère que la moitié de ce volume (10%). Ensuite viennent les sites pornographiques et les sites de replay. Les sites de Streaming musicaux représentent 5% des connections, mais une part plus faible du volume.

TABLE 11.2: Répartition des classes d'application en fonction du volume descendant par jour et par sonde

(a) Lyon

	Top Applications (pourcentage du volume total descendant)				
Date	1	2	3	4	5
05/07/2011	Streaming (47.69 %)	Web (18.75 %)	Download (18.13 %)	P2P (8.49 %)	Games (2.45 %)
06/07/2011	Streaming (47.95 %)	Web (19.56 %)	Download (17.29 %)	P2P (9.25 %)	Games (2.78 %)
07/07/2011	Streaming (47.79 %)	Download (19.53 %)	Web (18.22 %)	P2P (10.26 %)	Mail (1.66 %)
08/07/2011	Streaming (44.73 %)	Download (21.40 %)	Web (18.66 %)	P2P (6.98 %)	Games (3.48 %)
09/07/2011	Streaming (48.82 %)	Download (21.67 %)	Web (15.93 %)	P2P (10.31 %)	Unknown (1.60 %)
10/07/2011	Streaming (53.38 %)	Download (17.90 %)	Web (17.24 %)	P2P (8.46 %)	News (1.02 %)
11/07/2011	Streaming (49.01 %)	Web (20.52 %)	Download (15.93 %)	P2P (9.52 %)	Unknown (1.97 %)
12/07/2011	Streaming (51.64 %)	Web (19.19 %)	Download (14.29 %)	P2P (9.78 %)	Unknown (2.62 %)

(b) Montsouris

	Top Applications (pourcentage du volume total descendant)				
Date	1	2	3	4	5
05/07/2011	Streaming (38.86 %)	Web (25.47 %)	Download (21.37 %)	P2P (7.81 %)	Mail (3.18 %)
06/07/2011	Streaming (44.78 %)	Web (22.48 %)	Download (17.19 %)	P2P (7.64 %)	Mail (4.17 %)
07/07/2011	Streaming (43.26 %)	Web (23.62 %)	Download (18.67 %)	P2P (6.28 %)	Mail (3.84 %)
08/07/2011	Streaming (44.94 %)	Web (22.99 %)	Download (17.42 %)	P2P (5.38 %)	Mail (4.67 %)
09/07/2011	Streaming (48.70 %)	Web (21.94 %)	Download (15.70 %)	P2P (7.42 %)	Unknown (2.94 %)
10/07/2011	Streaming (48.21 %)	Web (17.00 %)	Download (16.42 %)	P2P (13.64 %)	Unknown (2.12 %)
11/07/2011	Streaming (42.76 %)	Web (23.87 %)	Download (20.79 %)	P2P (5.65 %)	Mail (4.19 %)
12/07/2011	Streaming (39.86 %)	Download (24.96 %)	Web (21.23 %)	P2P (7.25 %)	Mail (3.72 %)

(c) Rennes

	Top Applications (pourcentage du volume total descendant)				
Date	1	2	3	4	5
05/07/2011	Streaming (47.23 %)	Download (24.07 %)	Web (16.12 %)	P2P (5.38 %)	News (3.19 %)
06/07/2011	Streaming (46.35 %)	Download (23.55 %)	Web (15.93 %)	P2P (7.74 %)	Games (2.40 %)
07/07/2011	Streaming (47.34 %)	Download (23.48 %)	Web (16.43 %)	P2P (7.80 %)	Mail (1.73 %)
08/07/2011	Streaming (43.81 %)	Download (26.73 %)	Web (16.25 %)	P2P (6.09 %)	Enterprise (3.41 %)
09/07/2011	Streaming (44.21 %)	Download (25.54 %)	Web (15.53 %)	P2P (8.56 %)	Enterprise (3.19 %)
10/07/2011	Streaming (41.58 %)	Download (22.86 %)	Web (19.06 %)	P2P (11.12 %)	Games (2.60 %)
11/07/2011	Streaming (36.92 %)	Download (19.52 %)	Web (15.81 %)	P2P (11.52 %)	Unknown (6.29 %)
12/07/2011	Streaming (40.15 %)	Download (19.92 %)	Web (16.78 %)	P2P (10.66 %)	Unknown (5.03 %)

TABLE 11.3: Composition du trafic Streaming sur la semaine pour la sonde de Lyon (les tableaux sont ordonnées en fonction du volume descendant)

(a) Répart. du type de trafic Streaming

Service	Conn.	Volume Down
Clip	30.36 %	77.83 %
CatchUp TV	0.22 %	6.71 %
RadioLive	0.97 %	4.90 %
Unknown	39.23 %	4.86 %
TVLive	0.66 %	3.84 %
Advertisement	26.54 %	1.56 %
Chat	0.10 %	0.18 %
Games	1.92 %	0.12 %

(b) Popularité des sites de Streaming

Service	Conn.	Volume Down
Unknown	68.15 %	27.83 %
YouTube	4.27 %	22.02 %
MegaVideo	1.42 %	11.26 %
DiversX	4.88 %	9.71 %
Orange	2.06 %	6.80 %
M6	0.08 %	3.94 %
DailyMotion	1.10 %	3.67 %
AutresRadio	0.16 %	3.16 %
FranceTelevision	0.49 %	1.97 %
Deezer	4.99 %	1.70 %
Apple	0.11 %	1.53 %
Canal+	0.21 %	1.20 %
RadioFrance	0.75 %	1.08 %

TABLE 11.4: Seuils horaires par application et utilisateur pour déterminer l'usage d'une application

Classe	Volume		Nombre de connexions	Politique
	Down	Up		
WEB	300kB	500kB	20	All
P2P	1 MB	1 MB	10	Any
STREAMING	1 MB	1 MB	–	Any
DOWNLOAD	2 kB	1 kB	–	Any
MAIL	30kB	3 kB	–	All
GAMES	5 kB	5 kB	–	Any
VOIP	200kB	200kB	–	All
CHAT	10kB	10kB	–	Any

11.1.1.3 Regroupement des utilisateurs

Dans le but de comprendre les usages de l'Internet, nous utilisons des techniques de regroupement (ou "clustering"). Nous reprenons l'approche utilisée dans [37] en l'appliquant à diverses échelles de temps pour comprendre l'impact de durée de l'analyse.

L'idée de cette section est de ne pas se focaliser sur les volumes générés par les applications, mais l'usage de ces applications. Nous commençons par définir empiriquement un seuil pour l'usage d'une application (Tab. 11.4). Pour chaque utilisateur, nous obtenons donc un vecteur binaire correspondant à l'usage de chaque application.

Nous utilisons alors une technique de regroupement hiérarchique ("hierarchical clustering"). Nous construisons un arbre des utilisateurs pour former des groupes entre eux. Tout d'abord, nous affectons chaque utilisateur à une feuille de l'arbre. Ensuite pour construire les branches, nous considérons cette méthode agglomérative (ou de bas en haut) :

1. les deux plus proches nœuds[1] de l'arbre sont regroupés ;
2. ces deux nœuds sont remplacés par un nouveau nœud ;
3. le nouvel ensemble de nœuds est à nouveau agrégé jusqu'à ce qu'il n'y ait plus qu'un seul nœud racine.

Nous avons besoin d'une métrique pour définir la distance entre les nœuds, et avons choisi la métrique Tanimoto [51].

Nous nous focalisons dans le reste de cette section sur les 100 utilisateurs générant le plus de volume. En effet, ces utilisateurs font la majeure partie du trafic de la plateforme (Tab. 11.5).

Les utilisateurs qui génèrent le plus de trafic ont un usage particulier et très important. À titre d'exemple, nous nous focalisons sur les 5 principaux utilisateurs de la plateforme dans la semaine (Tab. 11.6). Le plus important dans la Tab. 11.6 est que la principale application utilisée est extrêmement majoritaire (de 68 à 96% du trafic de l'utilisateur). D'autre part, ces utilisateurs génèrent autour de 1.5% du volume downstream de la pla-

[1] au départ, les nœuds sont les feuilles

TABLE 11.5: Pourcentage de Volume des top utilisateurs

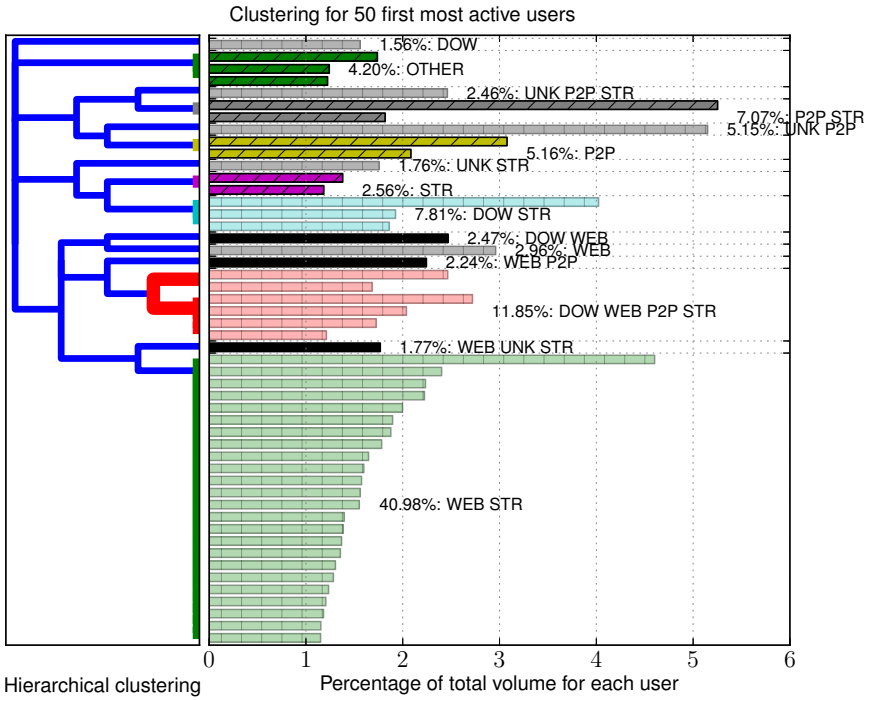
	Pourcentage du volume généré par top 50			top 100		
Heure	Down	Up	Nb. Conn.	Down	Up	Nb. Conn.
07/05	38%	51%	17%	55%	64%	35%
00	79%	66%	28%	90%	76%	71%
01	78%	90%	42%	83%	99%	82%
02	82%	67%	65%	83%	70%	83%
03	81%	82%	64%	82%	83%	77%
04	79%	66%	67%	80%	67%	78%
05	69%	86%	63%	71%	87%	76%
06	81%	51%	49%	86%	65%	67%
07	64%	65%	38%	80%	81%	63%
08	62%	55%	21%	77%	73%	39%
09	61%	40%	26%	77%	68%	47%
10	55%	62%	21%	70%	76%	44%
11	58%	40%	21%	74%	65%	34%
12	58%	35%	15%	71%	70%	44%
13	50%	53%	17%	68%	64%	32%
14	57%	50%	17%	73%	69%	32%
15	58%	49%	26%	75%	70%	43%
16	59%	63%	22%	76%	75%	35%
17	58%	51%	17%	74%	71%	35%
18	58%	42%	23%	76%	65%	39%
19	58%	57%	21%	74%	80%	42%
20	66%	50%	26%	79%	77%	47%
21	60%	73%	24%	76%	87%	44%
22	69%	46%	30%	81%	72%	45%
23	64%	63%	22%	79%	85%	44%

teforme : ils consomment donc 18 fois plus de ressources qu'une répartition équitable (pour 1200 utilisateurs).

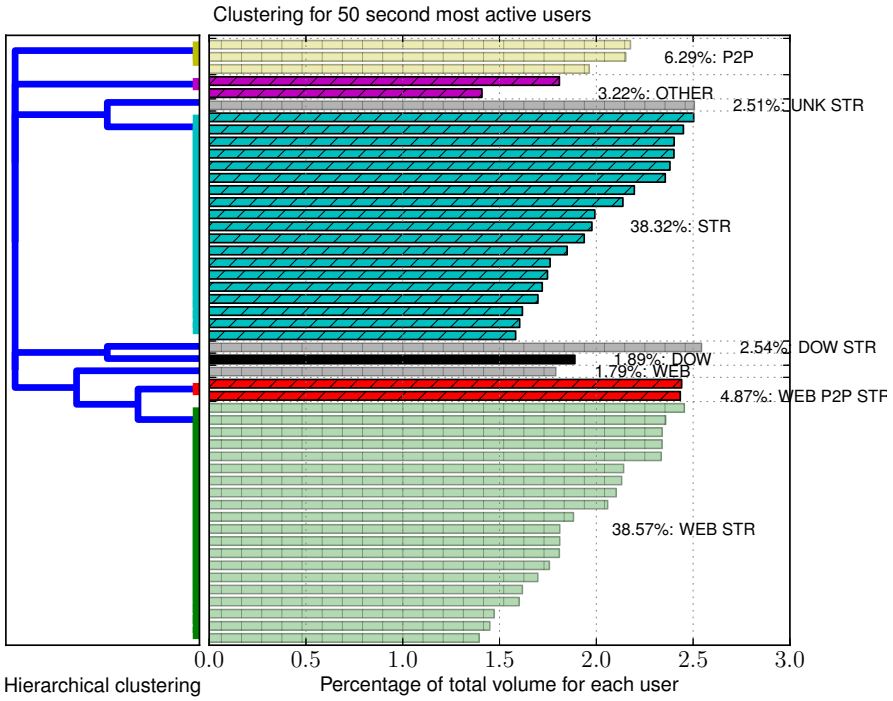
TABLE 11.6: Top 4 utilisateurs (le plus de volume up+down) statistiques sur la semaine

Rang	Appli. princ.	Vol. Tot	Vol. Down	Vol. Up	Nb. Conn.	$\frac{vol\_down}{tot\_down}$	$\frac{vol\_up}{tot\_up}$	$\frac{nb\_conn}{tot\_conn}$
1	P2P (95%)	63.6 GB	15.2 GB	48.5 GB	157,642	0.5%	15.1%	0.3%
2	Streaming (68%)	56.9 GB	55.0 GB	2.0 GB	828,782	1.7%	0.6%	1.3%
3	P2P (77%)	56.6 GB	49.8 GB	6.8 GB	1,355,444	1.5%	2.1%	2.2%
4	Download (94%)	51.0 GB	50.8 GB	0.2 GB	140,298	1.5%	0.1%	0.2%
5	P2P (95%)	37.6 GB	30.0 GB	7.6 GB	298,478	0.9%	2.4%	0.5%

Dans la Fig. 11.1a, nous avons réalisé le regroupement des 50 utilisateurs générant le plus de volume. Il y a une très grande variabilité des principaux utilisateurs comme vu pour les 5 premiers utilisateurs. Cela reste vrai pour les 25 principaux utilisateurs. Par contre, nous voyons déjà se dessiner une large fraction d'utilisateurs dans la classe mixant Web et Streaming. Ces deux applications regroupent la majorité du trafic des 50 principaux utilisateurs (même si ceux-ci utilisent plus le P2P que les autres). Mais dans la Fig. 11.1b où nous nous focalisons sur les utilisateurs de la 51$^{\text{ème}}$ à la 100$^{\text{ème}}$ place, nous avons seulement deux classes de mix d'applications : Streaming seulement, et Web et Streaming. Cela correspond aux résultats trouvés dans [37], et cela montre que l'usage de l'Internet résidentiel est passé en 6 ans du P2P au Web et Streaming (en comparant avec [40]).



(a) 50 premiers utilisateurs



(b) 50 deuxièmes utilisateurs

FIGURE 11.1: Analyse par regroupement des utilisateurs pour la semaine entière sur la sonde de Lyon

Les études à d'autres échelles (par jour ou par heure) donnent le même résultat. Cela signifie que cet étude des usages de l'Internet résidentiel peut s'appliquer à plusieurs

TABLE 11.7: Description des traces

Type & Localisation	ADSL M	FTTH M	ADSL M	FTTH M	ADSL R	FTTH M	ADSL R	FTTH M
Date	**2008**/07	**2008**/07	2009/11	2009/11	2009/12	2009/12	2010/02	2010/02
Heure de début	$20\,h$	$20\,h$	$20\,h$	$20\,h$	$20\,h$	$\mathbf{14\,h}$	$20\,h$	$20\,h$
Durée	$1\,h\,30$	$1\,h$	$1\,h\,20$	$0\,h\,38$	$1\,h$	$0\,h\,58$	$1\,h$	$0\,h\,28$
Utilisateurs actifs Web/Str. †	1121	1198	650	2502	795	2009	607	2763
Utilisateurs Streaming§	109	121	96	336	113	252	74	279
Videos Streaming	428	630	405	1462	334	865	258	866
Utilisateurs YouTube§	41	30	48	185	47	106	46	153
Videos YouTube	215	142	210	660	140	400	176	496
Utilisateurs DailyMotion§	25	20	16	48	12	20	13	29
Videos DailyMotion	83	154	45	84	53	35	25	44

† au moins 10 connections (Web et Streaming)
§ regardant au moins 1 vidéo

échelles et donner un résultat cohérent. La seule restriction observée dans nos données intervient si l'on se base sur les heures creuses au milieu de la nuit.

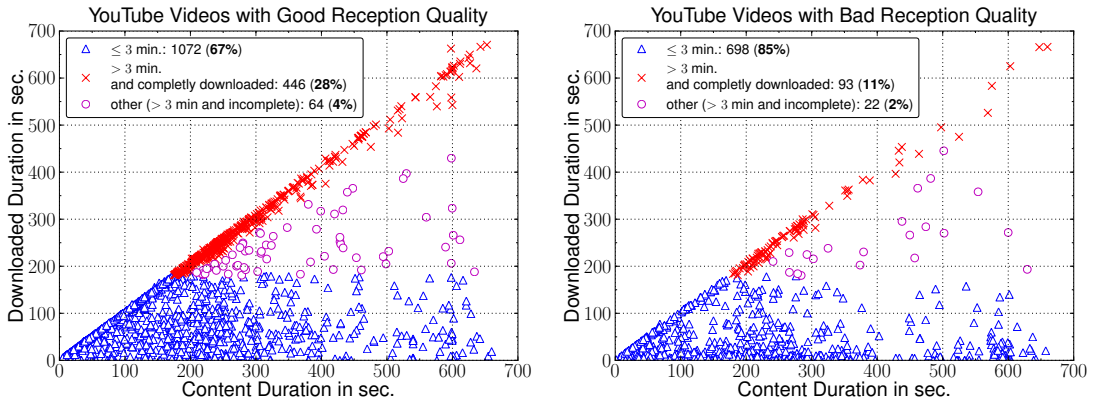
Les principaux enseignements de cette section sont donc que l'usage des clients est principalement sur le Web et le Streaming. Même si la fraction de trafic des tout premiers utilisateurs (top 20) peut influencer les caractéristiques générales de la plateforme (et avec des applications particulières), une large majorité d'utilisateurs est focalisée sur ces applications "interactives". Cela doit indiquer aux opérateurs que l'amélioration des débits (satisfaisant les usagers du P2P et Download) n'est pas une solution pour l'ensemble des clients d'une plateforme.

11.1.2 Analyse de la performance du vidéo Streaming

Dans cette section, nous nous focalisons sur l'analyse du trafic Streaming. En effet, nous venons de voir qu'il s'agit d'une application générant le plus de volume sur les plateformes Internet résidentiel, et qui en outre est utilisée par une grande part des clients. Nous étudions principalement YouTube, qui est le service le plus populaire de vidéo Streaming, mais aussi DailyMotion qui est un de ses principaux concurrents à titre de comparaison.

11.1.2.1 Détail sur les captures longitudinales

Nous avons utilisé le même type de trace que dans la Sect. 11.1.1. Les sondes sont réparties sur 2 sites : Montsouris et Rennes. Les captures se répartissent sur une durée de 3 ans : 2 traces de référence en 2008, et 6 traces de fin 2009 à début 2010. Les détails de ces traces sont données dans la Tab. 11.7.



(a) Vidéos avec une bonne qualité de réception **(b)** Vidéos avec une mauvaise qualité de réception

FIGURE 11.2: Pourcentage de vidéo téléchargée en fonction de la durée de la vidéo pour YouTube

11.1.2.2 Analyse du comportement des utilisateurs

Nous essayons de voir quel est l'impact de la qualité perçue sur l'usage du Streaming. Nous définissons donc un indicateur simple d'expérience utilisateur :

$$\text{taux normalisé} = \frac{\text{débit réseau moyen}}{\text{débit d'encodage}}$$

Si le débit d'encodage est supérieur au débit réseau moyen, nous sommes sûrs que la vidéo ne peut être visionnée correctement. Au contraire si le débit d'encodage est inférieur au débit réseau moyen, la vidéo a de fortes chances d'être visionnée correctement. Nous sommes conscients que si le *taux normalisé* est proche de 1, l'évaluation de la qualité est imprécise.

Dans la Fig. 11.2, pour chaque vidéo présente dans nos traces, nous la positionnons selon sa durée totale et la durée téléchargée. De plus, le marqueur utilisé dans le graphe nous permet de distinguer les vidéos comme suit :

- les vidéos courtes (moins de 3 minutes) ;
- les vidéos longues (plus de 3 minutes) et complètement téléchargées ;
- les vidéos longues (plus de 3 minutes) et téléchargées partiellement.

L'intérêt de cette figure est de présenter les données en fonction de la qualité des vidéos : dans la Fig. 11.2a, nous présentons les vidéos avec un taux normalisé supérieur à 1 ; alors que dans la Fig. 11.2b, seules les vidéos avec un taux normalisé inférieur à 1 sont représentées.

Tout d'abord, YouTube est principalement utilisé pour visionner des vidéos courtes (indépendamment de la qualité) : 3/4 des vidéos regardées dans nos traces durent moins de 3 minutes. Pour le cas des vidéos longues, soit elles sont téléchargées complètement, soit elles sont téléchargées pour moins de 3 minutes. L'impact de la qualité de réception est sur la répartition de ces types d'usage : en cas de bonne qualité de réception, trois fois plus de vidéos longues sont complètement téléchargées (28%) qu'en cas de mauvaise qualité de réception (11%).

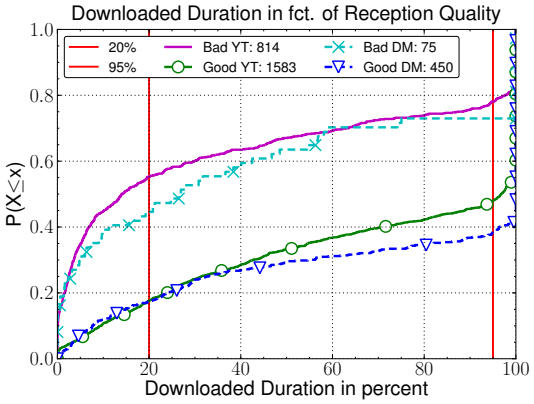


FIGURE 11.3: Pourcentage de vidéo téléchargée en fonction de la qualité de réception

Nous traçons finalement dans la Fig. 11.3 la CDF du pourcentage de vidéo téléchargée (par rapport à sa durée totale). Dans ce graphe, nous distinguons les vidéos avec une bonne qualité de réception de celles avec une mauvaise qualité. D'autre part, pour vérifier que les conclusions ne sont pas dues au service, nous incluons les données pour YouTube et DailyMotion.

L'impact de la qualité de réception sur la durée téléchargée est très claire dans ce graphe. De plus les courbes pour YouTube et DailyMotion sont extrêmement similaires. Deux zones se détachent :

- les vidéos complètement téléchargées ($\geq 95\%$) ;
- les vidéos très peu téléchargées ($\leq 20\%$).

En cas de mauvaise qualité de réception, très peu de vidéos sont téléchargées complètement. De plus, la décision d'interrompre le téléchargement est prise très tôt (dans les premiers 20% de la durée de la vidéo) : la courbe est concave au début puis quasi-plate. En cas de bonne qualité de réception, la moitié des vidéos sont téléchargées complètement. La décision d'interrompre le téléchargement n'est pas prise au début : la courbe est quasi-linéaire sur toute la plage des durées.

Ceci implique que la qualité de réception et, à plus forte raison, la qualité d'expérience ont un impact important sur la façon de réagir des utilisateurs. De plus, comme les vidéos de bonne ou mauvaise qualité peuvent se succéder (les détails ne sont pas inclus dans ce résumé), la réaction est due directement à la vidéo regardée, et pas vraiment à un phénomène à plus long terme (lassitude face à plusieurs vidéos avec une mauvaise qualité de réception. . .).

11.2 Mesures Actives

Dans cette section, nous nous focalisons sur les mesures actives sur le vidéo Streaming et en particulier sur YouTube (le service le plus populaire, comme vu en Sect. 11.1.1). Il s'agit de lancer des requêtes pour des vidéos YouTube et de mesurer la qualité perçue. Le principal problème dans ce contexte est d'être sûrs que les mesures effectuées correspondent bien à ce que les utilisateurs perçoivent. L'étude des mesures passives

en Sect. 11.1.1 nous garantit une bonne compréhension du sujet, et nous permet de jouer sur les paramètres qui se révèleront importants pour les mesures actives.

11.2.1 Outil d'évaluation de la qualité d'expérience

L'évaluation de la qualité d'expérience (QoE) est intrinsèquement difficile car cela implique une prise en compte du ressenti de l'utilisateur ainsi que le contexte de l'expérience. Dans le cas du vidéo Streaming, nous approximons la QoE ressentie par l'utilisateur principalement par le nombre d'interruptions obtenues dans le visionnage de la vidéo. En effet, la distribution de la vidéo s'effectue à débit d'encodage constant et sur support HTTP (au dessus de TCP). Ce mécanisme de transport fiable implique que l'intégralité de la vidéo sera distribuée même en cas de congestion. Comme les services de vidéo Streaming utilisent la technique de *Progressive Download* (téléchargement progressif), le visionnage du début de la vidéo commence pendant que le reste est téléchargé. Donc en cas de congestion, les données vidéo peuvent manquer et la lecture être interrompue. Ce sont ces interruptions que nous considérons comme le premier indicateur de QoE dans nos mesures.

11.2.1.1 Présentation de notre outil

Nous avons donc crée un outil, `Pytomo` [43], pour mesurer la QoS et la QoE des vidéos YouTube. L'outil commence par collecter les liens des vidéos les plus populaires de la semaine dans une liste de vidéos à analyser, il boucle ensuite sur ces différentes actions :

1. sélection d'une vidéo dans la liste des vidéos à analyser ;
2. récupération des liens des vidéos liées à cette vidéo et mise dans la liste des vidéos à analyser ;
3. récupération de l'URL du serveur délivrant la vidéo ;
4. évaluation des statistiques de QoS ;
5. évaluation des statistiques de QoE.

Les statistiques de QoS sont les suivantes :

- ping : minimum, moyenne, maximum (calculé sur 10 échantillons) ;
- informations vidéo : format, durée, taille, encodage moyen ;
- téléchargement : débit moyen, débit initial (sur les 3 premières secondes), débit maximum instantané (sur une fenêtre TCP).

11.2.1.2 Modèle de lecture des vidéos Streaming

L'évaluation des interruptions a été réalisée à partir d'un modèle de visionnage des vidéos. Pour se faire nous téléchargeons une vidéo et construisons en permanence deux échelles de temps :

$D(t)$: durée de vidéo téléchargée à la date t *ie.* la quantité de vidéo qui a été téléchargée en termes de durée de visionnage (obtenu à partir des *timestamps* des tags FLV) ;

$P(t)$: durée de vidéo visionnée à la date t *ie.* la quantité de vidéo qui a été regardée.

Ces deux échelles correspondent respectivement aux barres rouge et grise du lecteur vidéo YouTube. Quand la barre rouge correspondant au visionnage se rapproche trop de la barre grise correspondant au téléchargement, la lecture est interrompue. Nous avons donc :

$$D(t) - P(t) < \text{minimal-playout-buffer} \Rightarrow \text{Interruption de la lecture}$$

La lecture reprend si il y a assez de vidéo téléchargée :

$$D(t) - P(t) > \text{minimal-restart-buffer} \Rightarrow \text{Reprise de la lecture}$$

Avec cette évaluation de l'état de la lecture, nous sommes capables de déterminer le nombre d'interruptions durant le visionnage de la vidéo. Ce modèle ne prend pas en compte les sauts dans la vidéo, ni les pauses dues à l'utilisateur. Nous sommes conscients de ces limitations, et avons construit ce modèle uniquement dans le but de transcrire l'utilisation nominale de YouTube. De plus, dans le cas d'un saut dans la vidéo, le modèle est encore valide comme vu dans [41] : un saut dans une partie non-téléchargée crée une nouvelle connexion pour récupérer la vidéo à partir du moment choisit (au lieu du début).

Pour les statistiques de QoE, nous retenons les informations suivantes :

- le nombre d'interruptions ;
- la durée totale des interruptions ;
- la durée du buffer après 30 secondes de téléchargement.

11.2.2 Présentation des données collectées

Notre outil peut être lancé sur n'importe quel PC avec un accès Internet sans installation nécessaire. Nous avons donc bénéficié de diverses configurations pour lancer nos analyses.

11.2.2.1 Mesures de volontaires

Nous avons un grand nombre (145) de volontaires (principalement des collègues ou amis) qui ont accepté de lancer des mesures pour nous. Ces mesures effectuées principalement en Europe ou aux États-Unis nous ont tout d'abord, permis de tester notre

outil, et ensuite de comprendre l'influence des divers points de mesure sur le résultat. La durée de ces mesures va de quelques heures à plusieurs jours.

Ces mesures sont utilisées pour tester notre outil et comprendre le fonctionnement de YouTube.

11.2.2.2 Mesures en environnement contrôlé

Nous bénéficions d'un laboratoire en France connecté à plusieurs Fournisseurs d'Accès Internet (FAI) : 7 par ADSL, 1 par fibre et 1 par câble. Les accès ADSL ont exactement les même débits d'accès. Nous avons lancé des mesures simultanées sur ces divers accès pour obtenir des résultats comparés entre FAI alors que le lieu et le débit d'accès restent les mêmes. Nous utilisons des mesures de septembre et décembre 2011 : ces mesures ont subit un certain nombre d'interruptions (contrairement aux autres campagnes de mesures), ce qui nous permet de voir que la qualité de réception des vidéos YouTube est très dynamique ; et que de manière générale seulement peu d'interruptions dans les vidéos ont pu être observées.

11.2.2.3 Mesures d'étudiants de Kansas-City

Enfin, un ensemble d'étudiants de l'Université de Kansas-City (UMKC) a lancé des mesures simultanées depuis leur accès personnel le 8 décembre 2011 pour 2 heures. Ces 70 mesures nous ont permis de comprendre la répartition des serveurs de cache YouTube aux États-Unis, ainsi que de le comparer aux résultats en Europe.

11.2.3 Résultats

11.2.3.1 Sélection des serveurs vidéo en Europe

Nous présentons ici les principaux résultats des mesures contrôlées. L'URL du serveur vidéo indique à la fois sa localisation et son niveau dans la hiérarchie des caches YouTube (non détaillé dans ce résumé). Dans la Tab. 11.8, nous comptons pour chaque opérateur quel serveur de vidéo a distribué les vidéos YouTube.

Les serveurs observés se situent soit à Amsterdam, soit à Paris. Nous avons ces différences entre FAI :

- FAI B reçoit toutes ses vidéos d'un seul site de serveurs (`par08s01`) à Paris ;
- FAI N reçoit toutes ses vidéos d'un seul site de serveurs à Paris, mais avec deux types de nommage logique (`par08s01` et `par08s05`) ;
- FAI O bénéficie d'un site de serveurs dédiés (`orange-par1`), les adresses IP de ce site sont dans un AS spécifique (36040) ;

TABLE 11.8: Nombre de vidéos pour chaque FAI en fonction de Regexp sur les Urls des serveurs vidéo pour une mesure contrôlée en décembre 2011

URL Regexp	ISP A	B-A	B-F	F-R	F-V	N	O-L	S-E	S-V
o-o.preferred.par08s01.v[1-24].nonxt[378].c.youtube.com	0	1	2	0	0	0	0	0	0
o-o.preferred.ams03g05.v[1-24].nonxt[378].c.youtube.com	0	0	0	0	2	0	0	0	4
o-o.preferred.par08s01.v[1-24].lscache[1-8].c.youtube.com	0	2676	2677	0	0	1890	0	1967	1528
o-o.preferred.par08s05.v[1-24].lscache[1-8].c.youtube.com	1636	0	0	952	2425	799	0	0	0
o-o.preferred.ams03g05.v[1-24].lscache[1-8].c.youtube.com	150	0	0	0	206	0	0	3033	2488
o-o.preferred.orange-par1.v[1-24].lscache[1-8].c.youtube.com	0	0	0	0	0	0	2591	0	0
other	0	0	0	0	0	0	1	0	2

- FAI S et F sont redirigés vers les sites de serveurs de Paris ou Amsterdam avec des fréquences différentes : environ $2/3$ vers Amsterdam pour FAI S et 10% pour FAI F.

Ceci montre que la configuration des URLs des serveurs vidéo est très spécifique en fonctions des FAI.

11.2.3.2 Impact sur la qualité d'expérience

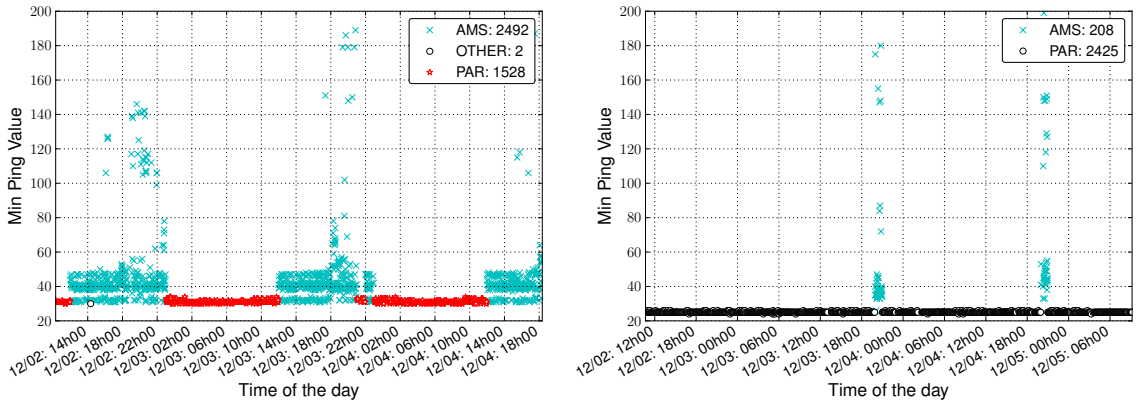
Dans la Fig. 11.4, nous montrons comment se répartissent les vidéos vers les deux sites de cache vidéo utilisés :

- pour le FAI S (Fig. 11.4a), de midi à minuit c'est le site d'Amsterdam alors que de minuit à midi c'est celui de Paris ;
- pour le FAI F (Fig. 11.4b), le principal site utilisé est celui de Paris, sauf de 20h à 21h où c'est celui d'Amsterdam qui est utilisé.

Le fait que ces changements se produisent à heure fixe montre qu'il s'agit de configurations statiques. D'autre part, ces configurations sont très dépendantes du FAI. Dans ce graphe, nous voyons aussi que le RTT entre notre laboratoire et le site de Paris est à environ 30 ms et très stable. Alors qu'avec le site d'Amsterdam, le RTT est environ à 40 ms mais a une grande variance avec des pointes à 200 ms.

Dans la Fig. 11.5, nous traçons pour les 2 FAI S et F (les mêmes que pour la Fig. 11.4) la courbe du pourcentage de vidéos avec au moins une interruption au cours du temps. Alors que le FAI S n'a quasiment aucune interruption, le FAI F souffre de périodes avec toutes ses vidéos interrompues au moins une fois. Si on relate cela aux observations de la Fig. 11.4, cela implique que même si le site d'Amsterdam est situé à une plus grande distance (en termes de RTT) et soumis à plus de variation de son RTT, la qualité obtenue est indépendante de ces considérations de QoS.

Tout ceci justifie que l'on se focalise sur des métriques de QoE plutôt qu'à des métriques de QoS.



(a) S-V **(b)** F-V

FIGURE 11.4: Valeur du ping en milli-secondes vers les principaux sites de serveurs vidéo de YouTube observés dans une mesure contrôlée en décembre 2011

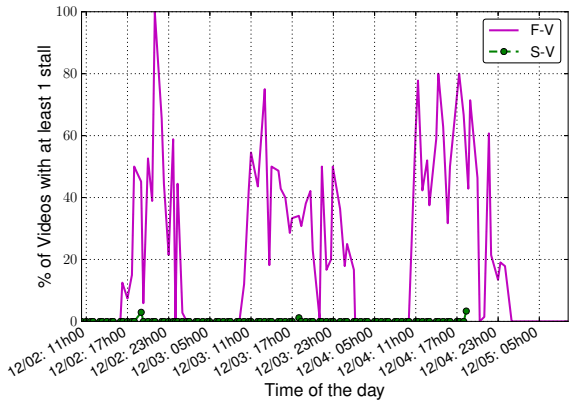


FIGURE 11.5: Évolution du pourcentage des vidéos avec au moins une interruption au cours du temps (par période de 60 minutes) pour 2 FAI dans une mesure contrôlée en décembre 2011

11.2.3.3 Sélection des serveurs vidéo aux États-Unis

Dans la Tab. 11.9, nous référençons les sites de serveurs vidéos avec leurs URLs, leur localisation ainsi que le nombre d'échantillons obtenus et le RTT moyen. Tout d'abord, nous avons 14 sites de serveurs ce qui est beaucoup plus que ce nous avons observé depuis la France. La plupart de ces sites sont localisés dans l'Ouest ou le Mid-Ouest des États-Unis. Le site le plus utilisé est celui de Washington même s'il est deux fois plus distant (en termes de RTT) que celui de Houston. Certaines vidéos sont obtenues depuis des sites très distants comme Chicago ou Miami.

Dans la Fig. 11.6, nous avons une représentation de la Tab. 11.9 sur une carte des États-Unis. Kansas-City est représenté par une marque et est l'endroit d'où sont réalisées les mesures. Chaque cercle représente les sites de serveurs vidéos. La taille des cercles indique le nombre de vidéos qui ont été servies par ce site, tandis que la couleur du cercle représente le RTT moyen vers ce site depuis Kansas-City.

TABLE 11.9: Valeur de ping en fonction des sites de serveurs vidéos pour les mesures de Kansas-City

URL Regexp	Localisation¶	Nb. d'échantillons	Ping moyen
WEST			
http ://o-o.preferred.iad09g05.v[1-24].lscache[1-8].c.youtube.com	Washington DC	1439	97
http ://o-o.preferred.sjc07s11.v[1-24].lscache[1-8].c.youtube.com	San Jose	446	73
http ://o-o.preferred.lax04s12.v[1-24].lscache[1-8].c.youtube.com	Los Angeles	147	75
http ://o-o.preferred.iad09s12.v[1-24].lscache[1-8].c.youtube.com	Washington DC	44	60
http ://o-o.preferred.sjc07s15.v[1-24].lscache[1-8].c.youtube.com	San Jose	10	61
MID-WEST			
http ://o-o.preferred.comcast-dfw1.v[1-24].lscache[1-8].c.youtube.com	Houston	719	50
http ://o-o.preferred.dfw06g01.v[1-24].lscache[1-8].c.youtube.com	Houston	308	59
http ://o-o.preferred.dfw06s08.v[1-24].lscache[1-8].c.youtube.com	Houston	190	24
http ://o-o.preferred.mna-mci1.v[1-24].lscache[1-8].c.youtube.com	Kansas City	71	184
http ://o-o.preferred.ord12s01.v[1-24].lscache[1-8].c.youtube.com	Chicago	64	1105
http ://o-o.preferred.kanren-lwc1.v[1-24].lscache[1-8].c.youtube.com	Lawrence	50	38
EAST			
http ://o-o.preferred.mia05s05.v[1-24].lscache[1-8].c.youtube.com	Miami	660	261
http ://o-o.preferred.lga15s20.v[1-24].lscache[1-8].c.youtube.com	New York	89	53

¶ la ville correspond au code d'aéroport dans l'URL

Ces informations confirment que le choix d'un site de serveurs vidéo dépend de beaucoup de paramètres et que des métriques comme la proximité géographique n'est pas le principal facteur. Ceci est d'autant plus valide aux États-Unis où YouTube a déployé beaucoup plus de sites qu'en Europe.

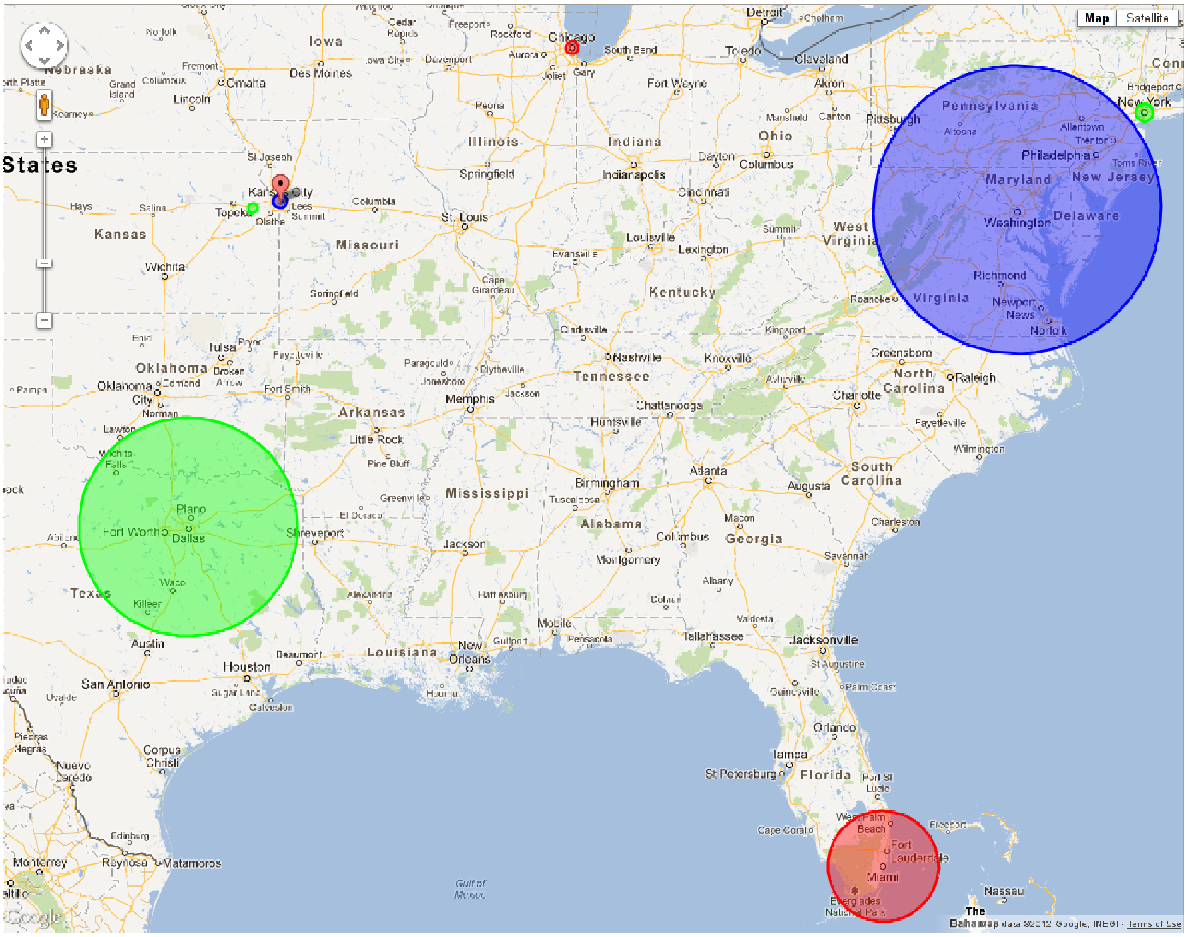


FIGURE 11.6: Carte indiquant la localisation des serveurs de vidéos, le nombre de requêtes obtenues sur chaque site (diamètre du cercle), et la distance (couleur du cercle : vert pour les ping $\leq 60\,ms$, bleu pour les ping $\geq 60\,ms$ et $\leq 200\,ms$, et rouge pour les ping $\geq 200\,ms$) des sites de serveurs vidéos pour YouTube pour les mesures de Kansas-City (marque).

CHAPITRE 12

Conclusion

Dans cette thèse, nous avons traité le problème de la mesure de l'Internet du point de vue d'un Fournisseur d'Accès Internet (FAI). Nous avons utilisé à la fois des traces de trafic passives et actives pour déterminer les principales caractéristiques du trafic Internet résidentiel, mais aussi pour comprendre le comportement des utilisateurs. Du point de vue d'un opérateur, savoir et comprendre le trafic de ses clients est très important pour planifier l'évolution de son réseau en fonction des besoins de ses usagers.

Nous avons tout d'abord utilisé des mesures actives pour inférer le comportement des utilisateurs à partir de plateformes locales d'un FAI. Nous avons montré qu'une grande part du trafic résidentiel est généré par un faible nombre d'usagers, et nous avons aussi proposé diverses techniques pour mieux gérer le trafic d'une plateforme locale. Nous nous sommes focalisés sur le vidéo Streaming non seulement car cette application est responsable de la majorité du trafic, mais aussi parce qu'elle est utilisé par la plupart des clients Internet. La réduction des pics de trafic sans perturber les utilisateurs a été traitée en considérant les applications et leur interactivité. Pour le vidéo Streaming, l'influence des caractéristiques de trafic et de la performance sur son usage a été étudiée et quantifiée.

Nous avons aussi développé un nouvel outil de mesure active pour explorer les mécanismes utilisés pour la distribution des vidéos et leur répartition sur les différents sites de serveurs. La connaissance du trafic vidéo Streaming gagnée dans la première partie de la thèse a profondément influencé le développement de cet outil. Nous avons donc montré comment le DNS et le FAI de l'utilisateur impactent la politique de distribution de la vidéo ainsi que la qualité d'expérience obtenue. Cette évaluation précise de la qualité de la vidéo nous a permis de comprendre les principaux facteurs dans la perception du service de vidéo Streaming. Nous avons donc crée et validé des méthodes efficaces pour mesurer la qualité du vidéo Streaming pour pouvoir évaluer à grande échelle et de manière non intrusive sur des captures passives de la part d'un opérateur.

Perspectives Ce travail nous donne plusieurs indications pour nos futures études. Tout d'abord, la nécessité d'une infrastructure de mesure permanente est démontrée, ainsi que la création de nouveaux indicateurs déterminés par les usages et les attentes des clients. Une des méthodes importantes développée dans cette thèse est de se

concentrer sur un nombre restreint d'utilisateurs générant la majorité du trafic. Ceci dans le but de bien comprendre les usages de l'Internet. Une méthode de regroupement des clients basés sur leurs usages a montré que même en se focalisant sur une courte période de temps, nous pouvons obtenir des résultats significatifs. L'étude d'un groupe restreint mais représentatif permet aussi d'utiliser des techniques d'analyse plus précises (comme le DPI).

Nous avons aussi développé des outils efficaces pour l'analyse des vidéos Streaming, et nous comptons suivre les évolutions de technologies pour garder la précision de cet outil. Le ratio du débit moyen sur le débit d'encodage donne une bonne approximation de la qualité d'expérience, et nous prévoyons d'utiliser cet indicateur pour suivre la qualité du vidéo Streaming. Néanmoins, un ensemble d'indicateurs est nécessaire pour prendre en compte les services les plus populaires pour les utilisateurs de l'Internet. Avec un ensemble complet d'indicateurs, nous volons non seulement évaluer l'état d'une plateforme, mais aussi être capable de déterminer la cause des éventuels problèmes. En effet, nous avons vu que des problèmes de performance peuvent influencer l'usage des services. Nous aimerions généraliser ces méthodes à plus de services pour signaler des problèmes de performances non détectés par les indicateurs réseau classiques.

Finalement, nous avons montré comment les mesures actives peuvent révéler la dynamique des observations des outils de surveillance. Nous prévoyons donc d'étendre notre outil à d'autres services, de sorte à pouvoir suivre activement et en direct les évolutions de l'usage du vidéo Streaming. Le développement d'outils similaires pour d'autres services important est aussi à prévoir.

Bibliography

[1] V. K. Adhikari, S. Jain, Y. Chen, and Z.-L. Zhang, "Vivisecting YouTube: An Active Measurement Study", In *IEEE INFOCOM 2012 Mini-conference*, 2012.

[2] 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.

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

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

[5] N. B. Azzouna and F. Guillemin, "Impact of Peer-to-Peer Applications on Wide Area Network Traffic: an Experimental Approach", In *IEEE Global Telecommunications Conference (GLOBECOM 2004)*, pp. 1544–1548, 2004.

[6] N. B. Azzouna, F. Clérot, C. Fricker, and F. Guillemin, "A flow-based approach to modeling ADSL traffic on an IP backbone link.", *Annales des Télécommunications*, 59(11-12), 2004.

[7] K. Bartos, M. Rehak, and V. Krmicek, "Optimizing flow sampling for network anomaly detection", In *Wireless Communications and Mobile Computing Conference (IWCMC), 2011 7th International*, pp. 1304 –1309, july 2011.

[8] beet.TV, "http://www.beet.tv/2010/09/dailymotiongrowth.html".

[9] A. Callado, C. Kamienski, G. Szabo, B. Gero, J. Kelner, S. Fernandes, and D. Sadok, "A Survey on Internet Traffic Identification", *Communications Surveys Tutorials, IEEE*, 11(3):37 –52, quarter 2009.

[10] 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 *IMC*, 2007.

[11] Y. Chabchoub, C. Fricker, F. Guillemin, and P. Robert, "Inference of flow statistics via packet sampling in the internet.", *IEEE Communications Letters*, 12(12):897–899, 2008.

[12] Y. Chabchoub, C. Fricker, F. Guillemin, and P. Robert, "On the statistical characterization of flows in Internet traffic with application to sampling", *Comput. Commun.*, 33(1), January 2010.

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

[14] comScore, "http://www.comscore.com/Press_Events/Press_Releases/ 2009/3/YouTube_Surpasses_100_Million_US_Viewers".

[15] M. Crovella and B. Krishnamurthy, *Internet Measurement: Infrastructure, Traffic and Applications*, John Wiley and Sons, Inc, 2006.

[16] N. Duffield, C. Lund, and M. Thorup, "Flow sampling under hard resource constraints", In *Proceedings of the joint international conference on Measurement and modeling of computer systems*, SIGMETRICS '04/Performance '04, pp. 85–96, New York, NY, USA, 2004, ACM.

[17] Endace, "http://www.endace.com/endace-dag-high-speed-packet-capture-cards.html".

[18] J. Erman, A. Gerber, M. T. Hajiaghayi, D. Pei, and O. Spatscheck, "Network-aware forward caching", In *WWW*, 2009.

[19] A. Finamore, M. Mellia, M. Munafo', and V. Gehlen, "Uncovering the Big Players of the Web", In *Traffic Monitoring and Analysis (TMA) Workshop*, 2012.

[20] A. Finamore, M. Mellia, M. Munafo, R. Torres, and S. Rao, "YouTube Everywhere: Impact of Device and Infrastructure Synergies on User Experience", In *IMC*, 2011.

[21] T. Fioreze, L. Granville, A. Pras, A. Sperotto, and R. Sadre, "Self-management of hybrid networks: Can we trust netflow data?", In *Integrated Network Management, 2009. IM '09. IFIP/IEEE International Symposium on*, pp. 577 –584, june 2009.

[22] V. Gehlen, A. Finamore, M. Mellia, and M. Munafo, "Uncovering the Big Players of the Web", In *TMA*, 2012.

[23] A. Gerber, J. Pang, O. Spatscheck, and S. Venkataraman, "Speed Testing without Speed Tests: Estimating Achievable Download Speed from Passive Measurements", In *Internet Measurement Conference (IMC)*, 2010.

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

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

[26] Google Public DNS, "http://code.google.com/speed/public-dns/docs/intro.html".

[27] J. Han, *Data Mining: Concepts and Techniques (Second Edition)*, Morgan Kaufmann Publishers Inc., San Francisco, CA, USA, 2006.

[28] T. Hoßfeld, R. Schatz, M. Seufert, M. Hirth, T. Zinner, and P. Tran-Gia, "Quantification of YouTube QoE via Crowdsourcing", In *IEEE International Workshop on Multimedia Quality of Experience - Modeling, Evaluation, and Directions (MQoE 2011)*, Dana Point, CA, USA, December 2011.

[29] HTB, "http://luxik.cdi.cz/~devik/qos/htb/".

[30] P. Juluri, L. Plissonneau, and D. Medhi, "Pytomo: A Tool for Analyzing Playback Quality of YouTube Videos", In *Proc. 23th International Teletraffic Congress: ITC*, 2011.

[31] K. Kumar, J. Xu, J. Wang, O. Spatschek, and L. Li, "Space-code bloom filter for efficient per-flow traffic measurement", In *INFOCOM 2004. Twenty-third AnnualJoint Conference of the IEEE Computer and Communications Societies*, volume 3, pp. 1762 –1773 vol.3, march 2004.

[32] Libpcap, "http://sourceforge.net/projects/libpcap/".

[33] G. Maier, A. Feldmann, V. Paxson, and M. Allman, "On Dominant Characteristics of Residential Broadband Internet Traffic", In *Internet Measurement Conference (IMC)*, 2009.

[34] Open DNS, “http://www.opendns.com/”.

[35] V. Paxson, “Strategies for Sound Internet Measurement”, In *IMC ’04: Proceedings of the 4th ACM SIGCOMM conference on Internet measurement*, pp. 263–271, ACM, 2004.

[36] M. Pietrzyk, J.-L. Costeux, G. Urvoy-Keller, and T. En-Najjary, “Challenging statistical classification for operational usage: the ADSL case”, In *IMC*, 2009.

[37] 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.

[38] PlanetLab, “http://www.planet-lab.org”.

[39] L. Plissonneau and E. Biersack, “A Longitudinal View of HTTP Video Streaming Performance”, In *MMSys*, 2012.

[40] L. Plissonneau, J.-L. Costeux, and P. Brown, “Detailed Analysis of eDonkey Transfers on ADSL”, In *2nd Conference on Next Generation Internet Design and Engineering (NGI06)*, 2006.

[41] L. Plissonneau, T. En-Najjary, and G. Urvoy-Keller, “Revisiting Web Traffic from a DSL Provider Perspective: the Case of YouTube”, In *19th ITC Specialist Seminar on Network Usage and Traffic*, 2008.

[42] I. Poese, B. Frank, B. Ager, G. Smaragdakis, and A. Feldmann, “Improving Content Delivery using Provider-aided Distance Information”, In *Internet Measurement Conference (IMC)*, 2010.

[43] Pytomo, “http://code.google.com/p/pytomo/”.

[44] A. Rao, A. Legout, Y.-s. Lim, D. Towsley, C. Barakat, and W. Dabbous, “Network characteristics of video streaming traffic”, In *Proceedings of the Seventh COnference on emerging Networking EXperiments and Technologies*, CoNEXT ’11, 2011.

[45] R. N. S. Alcock, “Application Flow Control in YouTube Video Streams”, *ACM SIGCOMM Computer Communication Review*, 41(2), April 2011.

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

[47] M. Siekkinen, G. Urvoy-Keller, E. W. Biersack, and D. Collange, “A Root Cause Analysis Toolkit for TCP”, In *Computer Networks*, volume 52, pp. 1846–1858, 2008.

[48] B. Staehle, M. Hirth, R. Pries, F. Wamser, and D. Staehle, “YoMo: A YouTube Application Comfort Monitoring Tool”, In *New Dimensions in the Assessment and Support of Quality of Experience for Multimedia Applications*, Tampere, Finland, June 2010.

[49] S. Sundaresan, W. de Donato, N. Feamster, R. Teixeira, S. Crawford, and A. Pescape, “Broadband Internet Performance: A View From the Gateway”, In *ACM SIGCOMM*, 2011.

[50] B. I. D. System, “http://bro-ids.org”.

[51] T. Tanimoto, “An elementary mathematical theory of classification and prediction”, In *IBM Program IBCLF*, 1959.

[52] TcpDump, “http://www.tcpdump.org”.

[53] 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.

[54] Tstat, “http://tstat.tlc.polito.it/”.

[55] G. Vu-Brugier, “Analysis of the Impact of early Fiber Access Deployment on Residential Internet Traffic”, In *Proc. 21th International Teletraffic Congress: ITC*, 2009.

[56] Wireshark, “http://www.wireshark.org”.