Understanding the Impact of the Access Technology: the Case of Web Search Services

Aymen Hafsaoui† ⋆, Guillaume Urvoy-Keller‡, Denis Collange§, Matti Siekkinen¶, and Taoufik En-Najjary§

†Eurecom, Sophia-Antipolis, France
‡Laboratoire I3S CNRS, Université de Nice, Sophia Antipolis, France
§Orange Labs, France
¶Aalto University School of Science and Technology, Finland

Abstract. In this paper, we address the problem of comparing the performance perceived by end users when they use different technologies to access the Internet. We focus on three key technologies: Cellular, ADSL and FTTH. Users primarily interact with the network through the networking applications they use. We tackle the comparison task by focusing on Web search services, which are arguably a key service for end users. We first demonstrate that RTT and packet loss alone are not enough to fully understand the observed differences or similarities of performance between the different access technologies. We then present an approach based on a fine-grained profiling of the data time of transfers that sheds light on the interplay between service, access and usage, for the client and server side. We use a clustering approach to identify groups of connections experiencing similar performance over the different access technologies. This technique allows to attribute performance differences perceived by the client separately to the specific characteristics of the access technology, behavior of the server, and behavior of the client.

Keywords: TCP, Performances, Web search, User behaviors, Access Impact .

1 Introduction

Telecommunication operators offer several technologies to their clients for accessing the Internet. We have observed an increase in the offering of cellular and Fiber-To-The-Home (FTTH) accesses, which now compete with the older ADSL and cable modem technologies. However, until now it is unclear what are the exact implications of the significantly different properties of these access technologies on the quality of service observed by clients.

Our main objective in this paper is to devise a methodology to compare the performance of a given service over different access technologies. We consider three popular technologies to access the Internet: Cellular, ADSL, and FTTH. We use traces of end users traffic collected over these three types of access networks under the control of a

⋆ Corresponding author.

major European ISP. We focus on an arguably key service for users: Web search engines, esp. Google and Yahoo.

In this paper, we present a methodology to separately account for the impact of access, service usage, and application on top. The methodology is based on breaking down the duration of an entire Web transaction into sub-components which can be attributed to network or either of the end points. This kind of approach is vital because the typical performance metrics such as average latency, average throughput, and packet loss only give an overview of the performance but do not say much about what the origins are.

Our methodology can be applied in different ways depending on the objectives of the study. For example, a service provider might only want to analyze the performance contribution of the server, while an ISP could be more interested in the (access) network's contribution. In both cases, the focus of the study could be the performance observed by the majority of clients or, alternatively, troubleshooting through identification of performance anomalies. We exemplify various use cases for Yahoo and Google Web search services.

2 Related Work

Comparing the relative merits of different access technologies has been the subject of a number of studies recently. In [1], the authors analyze passive traffic measurements from ADSL and FTTH commercial networks under the control of the same ISP. They demonstrate that only a minority of clients and flows really take advantage of the high capacity of FTTH access. The main reason is the predominance of p2p protocols that do not exploit locality and high transmission capacities of other FTTH clients.

In [2], the authors investigate the benefits and optimizations of TCP splitting for accelerating cloud services, using web search as an exemplary case study and through an experimental system deployed in a production environment. They report that a typical response to an average search query takes between 0.6 and 1.0 second (between the TCP SYN and the last HTTP packet). The RTT between the client and the data-center during the measurement period was around 100 milliseconds. Search time within the data-center ranges almost uniformly between 50 and 400 msec[1]. Four TCP windows are required to transfer the result page to the client when there is no packet loss. The total time taken in this case is 5RTT + search time.

In [3], the authors present results from a measurement campaign for GPRS, EDGE, cellular, and HSDPA radio access, to evaluate the performance of web transfers with and without caching. Results were compared with the ones of a standard ADSL line (down:1Mb/s; up:256kb/s). Benchmarks reveal that there is a visible gain introduced by proxies within the technologies: HSDPA is often close to ADSL but does not outperform it; In EDGE, the proxy achieves the strongest improvement, bringing it close to HSDPA performance.

In [4], the authors quantify the improvement provided by a 3G access compared to 2G access in terms of delays and throughput. They demonstrate that for wired access networks (ADSL and FTTH) the average number of servers accessed per subscriber is

[1] We observe a significant fraction of values outside of this range in Section 6.

one order of magnitude lower on the mobile trace, esp. because of the absence of P2P traffic. Focusing on the user experience when viewing multimedia content, they show how their behavior differs and how the radio access type influences their performance.

In [5] authors analyze Web search clickstreams by extracting the HTTP headers and bodies from packet-level traffic. They found that most queries consist of only one keyword and make little use of search operators, users issue on average four search queries per session, of which most consecutive ones are distinct. Relying on a developed Markov model that captures the logical relationships of the accessed Web pages authors reported additional insights on users' Web search behavior.

In [6] Stamou and all studied how web information seekers pick the search keywords to describe their information needs and specifically examine whether query keyword specifications are influenced by the results the users reviewed for a previous search. Then, they propose a model that tries to capture the results' influence on the specification of the subsequent user queries.

3 Data Sets

We study three packet level traces of end users traffic from a major French ISP involving different access technologies: ADSL, cellular and FTTH. ADSL and FTTH traces correspond to all the traffic of an ADSL and FTTH Point-of-Presence (PoP) respectively, while the cellular trace is collected at a GGSN level, which is the interface between the mobile network and the Internet. The cellular corresponds to 2G and 3G/3G+ accesses as clients with 3G/3G+ subscriptions can be downgraded to 2G depending on the base station capability. Table 1 summarizes the main characteristics of each trace.

	cellular	FTTH	ADSL
Date	2008-11-22	2008-09-30	2008-02-04
Starting Capture	13:08:27	18:00:01	14:45:02:03
Duration	01:39:01	00:37:46	00:59:59
Number of Connections	1772683	574295	594169
Well-behaved connections	1236253	353715	381297
Volume Upload(GB)	11.2	51.3	4.4
Volume Download(GB)	50.6	74.9	16.4

Table 1. Traces Description

In the present work, our focus is on applications on top of TCP, which carries the vast majority of bytes in our 3 traces, and close to 100% for the cellular technology. We restrict our attention to the connections that correspond to presumably valid and complete transfers, that we term well-behaved connections. Well-behaved connections must fulfill the following conditions: (i) A complete three-way handshake; (ii) At least one TCP data segment in each direction; (iii) The connection must finish either with a FIN or RESET flag. Well-behaved connections carry between 20 and 125 GB of traffic in our traces (see Table 1).

4 Web search Traffic: A First Look

In this section, we focus on the traffic related to Google Web Search engine, which is the dominant Web Search engine in our traces. We focus here on the overall performance metrics before introducing our methodology for finer grained analysis in Section 5. We compare the Google and Yahoo cases in Section 6.

To identify traffic generated by the usage of Google search engine, we isolate connections that contain the string www.google.com/fr in their HTTP header. Relying simply on information at the IP and TCP layers would lead to incorporate in our data set other services offered by Google like gmail or Google map, which are serviced by the same IPs.

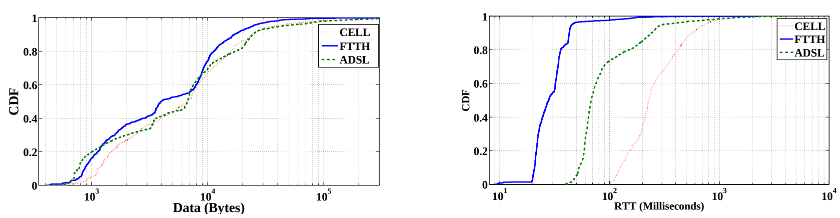
To identify Google search traffic for the upstream and downstream directions, we use TCP port numbers and remote address resolution. Table 2 summarizes the amount of Google search traffic we identified in our traces. We observed that FTTH includes the smallest number of such connections among the three traces, one explanation of this phenomenon was the short duration of the FTTH trace.

	Cellular	FTTH	ADSL
Well-behaved Connections	29874	1183	6022
Data Packets Upload	107201	2436	18168
Data Packets Download	495374	7699	139129
Volume Upload(MB)	74.472	1.66	11.39
Volume Download(MB)	507.747	8	165.79

Table 2. Google Search Traffic in the Traces

4.1 Connection Size

Figure 1(a) depicts the cumulative distribution of well-behaved Google search connection size in bytes. It appears that data transfer sizes are very similar for the three access technologies. This observation constitutes a good starting point since the performance of TCP depends on the actual transfer size. RTTs and losses also heavily influence TCP performance, as the various TCP throughput formulas indicate [7, 8]. Also, the available bandwidth plays a role. With respect to these metrics, we expect the performance of a service to be significantly influenced by the access technology since available bandwidth, RTTs[2] and losses are considerably different over ADSL, FTTH and Cellular. However, as we demonstrate in the remaining of this section, those metrics alone fail to fully explain the performance observed in our traces.



(a) Connections Size (b) RTT Estimation

Fig. 1. General Performances

[2] As noted in several studies on ADSL [9] and Cellular networks[10], the access technology often contributes to a significant fraction of overall RTT.

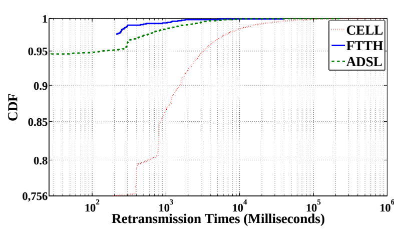
4.2 Latency

Several approaches have been proposed to accurately estimate the RTT from a single measurement point [11–15]. We considered two such techniques. The first method is based on the observation of the TCP 3-way handshake [12]: one first computes the time interval between the SYN and the SYN-ACK segment, and adds to the latter the time interval between the SYN-ACK and its corresponding ACK. The second method is similar but applied to TCP data and acknowledgement segments transferred in each direction [3]. One then takes the minimum over all samples as an estimate of the RTT. It is important to note that we take losses into account in our analysis (see next section).

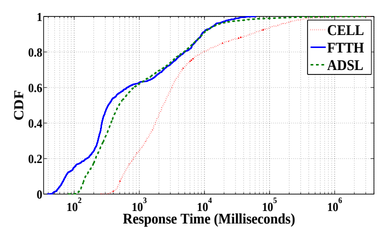
We observed that both estimation methods (SYN-/SYN-ACK and DATA-ACK) lead to the same estimates except for the case of cellular access because of a Performance Enhancing Proxy (PEP) which biased the results from the SYN-/SYN-ACK method, as the PEP responds to SYN packets from the clients on behalf of the servers. We thus rely on the DATA-ACK method to estimate RTTs over the 3 technologies. Figure 1(b) depicts the resulting RTT estimations for the 3 traces (for Google Web search service only). It clearly highlights the impact of the access technology on the RTT. FTTH access offer very low RTT in general – less than 50 ms for more than 96% of connections. This finding is in line with the characteristics generally advertised for FTTH access technology. In contrast, RTTs on the Cellular technology are notably longer than under ADSL and FTTH.

4.3 Packet Loss

To assess the impact of TCP loss retransmission times on the performance of Google Web search traffic, we developed an algorithm to detect retransmitted data packets, which happen between the capture point and the server or between the capture point and the client. This algorithm [4] is similar to the one developed in [11].



(a) Retransmission Time per Connection



(b) Transfer Time

Fig. 2. Immediate Access Impacts

If ever the loss happens after the observation point, we observed the initial packet and its retransmission. In this case, the retransmission time is simply the duration be-

[3] Keep in mind that we focus on well-behaved transfers for which there is at least one data packet in each direction. Hence, we can apply the second method.

[4] The used loss' detection algorithm is available on http://intrabase.eurecom.fr/tmp/papers.html. People are invited to check the correctness of our algorithm to detect losses

tween those two epochs[5]. When the packet is lost before the probe, we infer the epoch at which it should have been observed, based on the sequence numbers of packets. We try to separate real retransmission from network out of sequence events by eliminating durations smaller than the RTT of the connection.

Figure 2(a) depicts the cumulative distribution of retransmission time per connection for each trace. Retransmissions are clearly more frequent for the cellular access with more than 25% of transfers experiencing losses compared to less than 6% for ADSL and FTTH accesses. From previous works, we noticed that several factors explain high loss ratio for cellular access. In fact, in [16] authors recommend to use a loss detection algorithm, which uses dumps of each peer of the connection (this algorithm is not adapted for our case because our measurements have been collected at a GGSN level) to avoid spurious Retransmission Timeouts in TCP. In addition, authors report in [10] that spurious retransmission ratio, for SYN and ACK packets, in cellular networks is more higher for Google servers than other ones, due to short implemented Timeouts.

Most of the transfers are very short in terms of number of packets and we know that for such transfers, packet loss has a detrimental impact to the performance[17]. Thus, the performance of these transfers are dominated by the packet loss. In Sections 5.3 and 6, we analyze all connections, including the ones that experience losses by first removing recovery times from their total duration.

4.4 Application Level Performance

Our study of the two key factors that influence the throughput of TCP transfers , namely loss rate and RTT, suggest that, since Google Web search transfers have a similar profile on the 3 access technologies, the performance of this service over FTTH should significantly outperform the one of ADSL, which should in turn outperform the one of Cellular. It turns out that reality is slightly more complex as can be seen from Figure 2(b) where we report the distribution of transfer times (the figure for throughput is qualitatively similar but we prefer to report transfer times since Web search is an interactive service). Indeed, while the Cellular technology offers significantly longer response time, in line with RTT and loss factors, FTTH and ADSL have much closer performance than RTT and loss were suggesting.

In the next section, we propose a new analysis method that uncovers the impact of specific factors like the application and the interaction with user, and thus informs the comparison of access technology.

5 Interplay Between Application, Usage and Access

The analysis method that we use consists in two steps. In the first step, the transfer time of each TCP connection is broken down into several factors that we can attribute to different causes, e.g., the application or the end-to-end path. In a second step, we use a clustering approach to uncover the major trends within the different data sets under study.

[5] Those epochs are computed at the sender side by shifting the time series according to our RTT estimate.

5.1 Step 1: Data Time Break-down

For this first step, we introduce a methodology that has been initially proposed in [17]. The objective is to reveal the impact of each layer that contributes to the overall data transfer time, namely the application, the transport, and the end-to-end path (network layer and layers below) between the client and the server.

The starting point is that the vast majority of transfers consist of dialogues between the two sides of a connection, where each party talks in turn. This means that application instances rarely talk simultaneously on the same TCP connection [17]. We call the sentences of these dialogues *trains*.

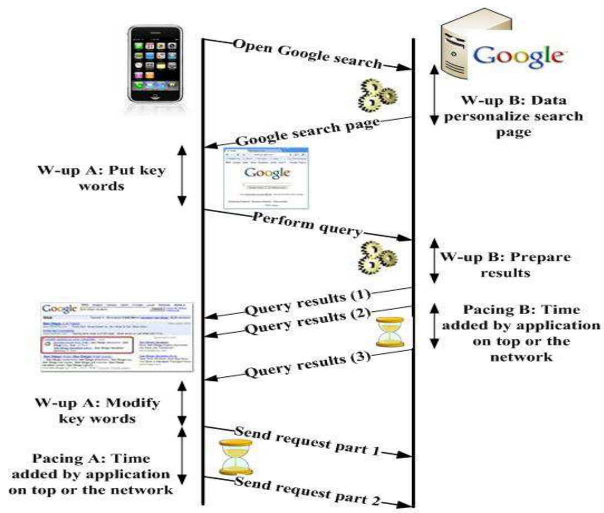


Fig. 3. Data Time Break-Down

We term A and B the two parties involved in the transfer (A is the initiator of the transfer) and we break down the data transfer into three components: warm-up time, theoretical time and pacing time. Figure 3 illustrates this break down in the case of a Google search where A is a client of the ISP and B is a Google server.

A warm-up corresponds to the time taken by A or B before answering to the other party. It includes durations such as thinking time at the user side or data preparation at the server side. For our use case, a warm-up of A corresponds to the time spent by the client to type a query and to browse through the results before issuing the next query (if any) or clicking on a link, whereas a warm-up of B corresponds to the time spent by the Google server to prepare the appropriate answer to the request.

Theoretical time is the duration that an ideal TCP transfer would take to transfer an amount of packets from A to B (or from B to A) equal to the total amount of packets exchanged during the complete transfer. Theoretical time can be seen as the total transfer time of this ideal TCP connection that would have all the data available right at the beginning of the transfer. For this ideal transfer, we further assume that the capacity of the path is infinite and an RTT equal to RTT_{A-B} (or RTT_{B-A}).

Once warm-up and theoretical times have been substracted from the total transfer time, some additional time may remain. We term that remaining time pacing time. While theoretical time can be attributed to characteristics of the path and warm-up time to applications and/or user, pacing factors effects due either to the access link or some mechanism higher up in the protocol stack. Indeed, as we assume in the computation of theoretical time that A and B have infinite access bandwidth, we in fact assume that we can pack as many MSS size packets within an RTT as needed, which is not necessarily true due to a limited access bandwidth. In this case, the extra time will be factored in the pacing time. Similarly, if the application or some middle-boxes are throttling the transmission rate, this will also be included in the pacing time. A contextual interpretation that accounts for the access and application characteristics is thus needed to uncover the cause behind observed pacing time. The above breakdown of total transfer time is computed for each side A and B separately.

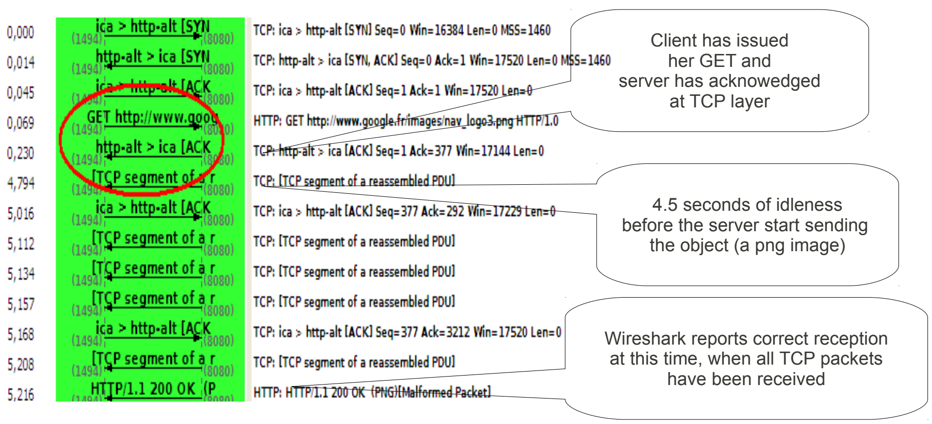


Fig. 4. Abnormal Long Response Time at The Server Side (Warm-up B value)

We report on Figure 4 an example of observed large warm-up time at the server side, for a client behind an ADSL access. We noticed that the acknowledgement received from the server indicates that the query (GET request) has been correctly received by the server, but it takes about 4.5 seconds before the client starts to receive the requested object (a png image in this case). As we can see next in Section 5.3, an easy identification of these extreme cases can be a useful application of our methodology.

5.2 Step 2: Data Clustering

The second analysis step is new as compared to our previous work [17]. For this second step, we use clustering approaches to obtain a global picture of the relation between the service, the access technology and the usage.

At the end of step 1, each well-behaved Google search connection is transformed into a point in a 6-dimensional space (pacing, theoretical and train time of the client and the server). To mine this data, we use a clustering technique to group connections with similar characteristics. We use an unsupervised clustering approach as we have no a priori knowledge of the characteristics of the data to be analyzed, e.g., a model of normal and abnormal traffic. We chose the popular Kmeans algorithm. A key issue when using Kmeans is the choice of the initial centroids and the number of clusters targeted. Concerning the choice of the centroids, we perform one hundred trials and take the best result, i.e., the one that minimizes the sum over all clusters of the distances between each point and its centroid.

To assess the number of clusters, we rely on a visual dimensionality reduction technique, t-SNE (t-Distributed Stochastic Neighbour Embedding)[18]. t-SNE projects multi-dimensional data on a plane while preserving the inner neighbouring characteristics of data. Application of t-SNE to our 6-dimensional data leads to the right plot of Figure 5(a). This figure indicates that a natural clustering exists within our data. In addition, a reasonable value for the number of clusters lies between 5 and 10. Last but not least the right plot of Figure 5(a) suggests that some clusters are dominated by a specific access technology while some others are mixed. We picked a value of 6 for the number of clusters in Kmeans. Note that we use the matlab implementation of Kmeans [19].

5.3 Results

Figure 5(b) depicts the 6 clusters obtained by application of Kmeans. We use boxplots[6] to obtain compact representations of the values corresponding to each dimension. We indicate, on top of each cluster, the number of samples in the cluster for each access technology. We use the same number of samples per access technology to prevent any bias in the clustering, which limits us to 1000 samples, due to the short duration of the FTTH trace. The ADSL and Cellular samples were chosen randomly among the ones in the respective traces. We also plot in Figure 6(b) the size of the transfers of each cluster and their throughput[7].

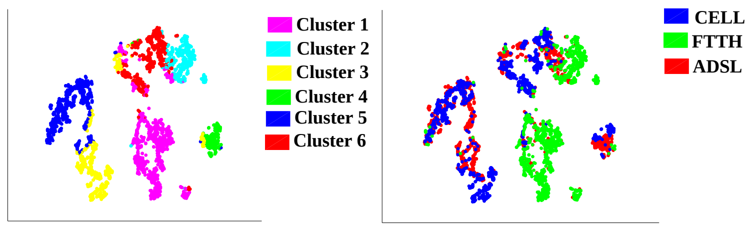
We first observe that the clusters obtained with Kmeans are in good agreement with the projection obtained by t-SNE as indicated in the left plot of Figure 5(a), where data samples are indexed using their cluster id in Kmeans.

Before delving into the interpretation of the individual clusters, we observe that three of them carry the majority of the bytes. Indeed, Figure 6(a) indicates that clusters 1 and 2 and 6 represent 83% of the bytes. Let us first focus on these dominant clusters.

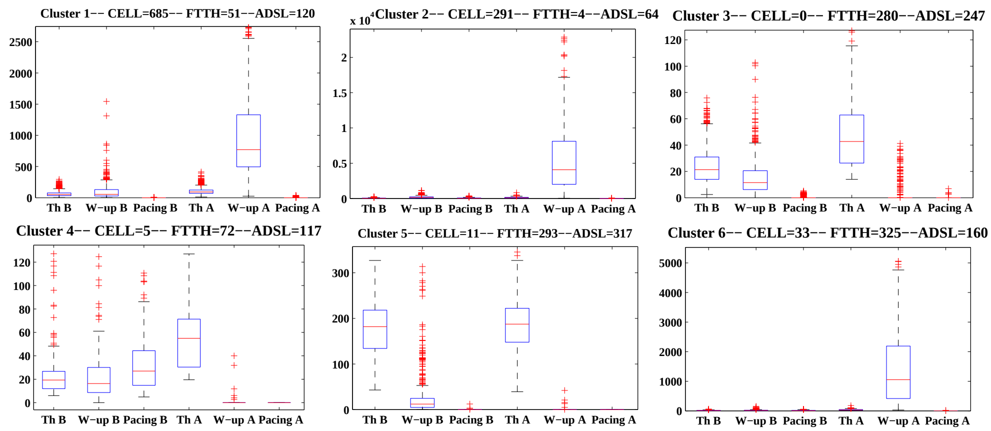
Clusters 1, 2 and 6 are characterized by large warm-up A values, i.e., long waiting time at the client side in between two consecutive requests. The warm-up A values are

[6] Boxplots are compact representations of distributions: the central line is the median and the upper and lower of the box the 25th and 75th quantiles. Extreme values -far from the waist of the distribution - are reported as crosses.

[7] We compute the throughput by excluding the tear down time, which is the time between the last data packet and the last packet of the connection. This specific metric that we term Application Level (AL) throughput offers a more accurate view of the user experience [17].

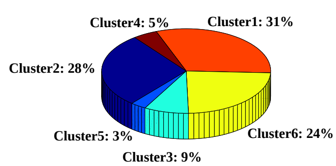


(a) T-SNE

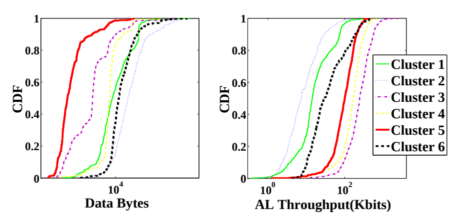


(b) K-means

Fig. 5. Google Search Engine Clusters



(a) Bytes Volume



(b) Clusters Parameters

Fig. 6. Google Search Engine Parameters

in the order of a few seconds, which are compatible with human actions. This behavior is in line with the typical use of search engines where the user first submits a query then analyzes the results before refining further her query or clicking on one of the links of the result page. Thus, the primary factor that influences observed throughputs in Google search traffic is the user behavior. In fact, identified values in clusters 1, 2 and 6 of Warm-up A are in line with results in [6] of the time between query submission and first click, where authors identified different users trends.

We can further observe that clusters 1 and 2 mostly consist of cellular connections while cluster 6 consists mostly of FTTH transfers. This means that the clustering algorithm first based its decision on the Warm-up A value; then, this is the access technology that impacts the clustering. As ADSL offers intermediate characteristics as compared to FTTH and Cellular, ADSL transfers with large Warm-up A values are scattered on the three clusters.

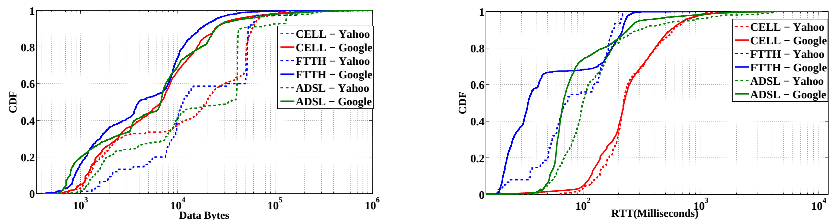
Let us now consider clusters 3, 4 and 5. Those clusters, while carrying a tiny fraction of traffic, feature several noticeable characteristics. First, we see almost no cellular connections in those clusters. Second, they total two thirds of the ADSL and FTTH connections, even though they are smaller than the ones in clusters 1, 2 and 6 – see Figure 6(b). Third, those clusters, in contrast to clusters 1, 2 and 6 have negligible Warm-up A values. From a technical viewpoint, Kmeans separates them based on the RTT as cluster 5 exhibits larger ThA and ThB values and also based on Pacing B values. After a further analysis of these clusters we observed that they corresponds to very short connection with an exchange of 2 HTTP frames, Google servers finish current connection after an idle period of 10 seconds. Moreover, cluster 3 presents cases when client opens Google web search page in their Internet browser without performing any search request, then after a time-out of 10 seconds Google server close the connection. In other hand, cluster 4 and 5 corresponds to Get request and HTTP OK response with an effective search, the main difference between cluster 4 and 5 were RTT and connection size.

More generally, we expect that our method, when applied to profile other services, will lead to some clusters that can be easily related to the behavior of the service under study while some others will relate anomalous or unsual behaviors that might require further investigation. For the case of Google search engine, we do not believe cluster 3,4,5 are anomalies per se that affects the quality of experience of users since the large number of connections in those clusters would prevent the problem from flying below the radar. We found only very few cases where the server's impact to the performance was dominating and directly impacting the quality of experience of the end user. Observing many such cases would have indicated issues, e.g., with service implementation or provisioning.

6 Contrasting Web Search Engines

The main idea in this section is to contrast Google results with others Web search services. For the case of our traces, we observed that the second dominant Web Search engine is Yahoo, though with an order of magnitude less connections. This low number of samples somehow limits the applicability of our clustering approach as used in the Google case. We restrict our attention to the following questions: (i) Do the two services offer similar traffic profile? (ii) Are services provisioned in a similar manner? Architecture of Google and Yahoo data-centers are obviously different but they must both obey the constraint that the client must receive its answer to a query in a maximum amount of time that is in the order of a few hundreds of milliseconds [2]. We investigate the service provisioning by analyzing the Warm-up B values (data preparation time at server) offered by the two services.

6.1 Traffic Profiles



(a) Connection Size (b) RTTs

Fig. 7. Yahoo vs. Google Web search services

Figure 7(a) shows cdfs of data connections size for Cellular, FTTH and ADSL traces for both Google and Yahoo. We observe for our traces that Yahoo Web search connections are larger than Google ones. An intuitive explanation behind this observation is that Yahoo Web search pages contain, on average, more photos and banners than ordinary Google pages.

Figure 7(b) plots cdfs of RTTs. We can observe that RTT values on each access technology are similar for the two services, which suggests that the servers are located in France and that it is the latency of the first hop that dominates.

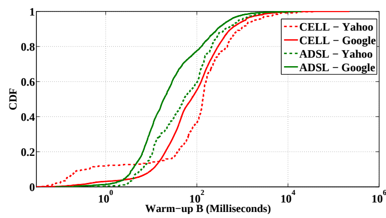
We do not present clustering results for Yahoo due to the small number of samples we have. However, a preliminary inspection of those results revealed the existence of clusters due to long Warm-up A values, i.e. long waiting times at the client side – in line with our observations with the Google Web search service. In the next section, we focus on the waiting time at the server side.

6.2 Data preparation time at the server side

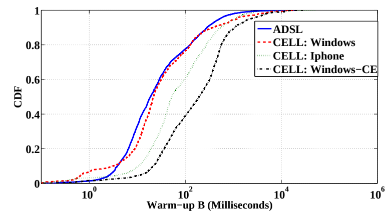
Figure 8(a) presents the cdf of warm-up B [8] values for both Yahoo and Google for the ADSL and Cellular technology (we do not have enough samples on FTTH for Yahoo to present them). We observe an interesting result: for both Yahoo and Google, the time to craft the answer is longer for cellular than for the ADSL technology. It suggests that both services adapt the content to cellular clients. A simple way to detect that the remote client is behind a wired or wireless access is to check its Web browser-User Agent as reported in the HTTP header. This is apparently what Google does as Figure 8(b) reveals (again, due to a low number of samples on Yahoo, we are not able to report a similar breakdown). Indeed, cellular clients featuring a laptop/desktop Windows operating system (Vista/XP/2000) experience similar warm-up B as ADSL clients while clients using Iphones or a Windows-CE operating system experience way higher warm-up B. As the latter category (esp. Iphones: more than 66% of Google connections) dominates in our

[8] We have one total warm-up B value per connection, which is the total observed warm-up B for each train.

dataset they explain the overall cellular plot of Figure 8(a). Note that further investigations would be required to fully validate our hypothesis of content adaptation. We could think of alternative explanations like a different load on the servers at the capture time or some specific proxy in the network of the ISP. However, it is a merit of our approach to pinpoint those differences and attribute them to some specific components like the servers here.



(a) Yahoo vs. Google



(b) Google

Fig. 8. Warm-up B

7 Conclusion

In this paper, we tackled the issue of comparing networking applications over different access technology – FTTH, ADSL and Cellular. We focused on the specific case of Web search services. We showed that packet loss, latency, and the way clients interact with their mobile phones all have an impact on the performance metrics on the three technologies. We devised a technique that (i) automatically extracts the impact of each of these factors from passively observed TCP transfers and (ii) group together, with an appropriate clustering algorithm, the transfers that have experienced similar performance over the three access technology. Application of this technique to the Google Web search service demonstrated that it provides easily interpretable results. It enables for instance to pinpoint the impact of usage or of raw characteristics of the access technology. We further compared Yahoo and Google Web search traffic and provided evidences that they are likely to adapt content to the terminal capability for cellular clients which impacts the performance observed. As future work, we will apply our approach to the profiling of other network services, which should be straightforward since our approach is application agnostic (we did not make any hypothesis on Google Web search to profile it). We intend to profile, among others, applications which are more bandwidth demanding like HTTP streaming. We also would like to investigate the usefulness of the method at higher levels of granularity, such as session or client level.

Acknowledgments

This work has been partly supported by France Telecom, project CRE-46142809 and by European FP7 COST Action IC0703 Traffic Monitoring and Analysis (TMA).

References

1. G. Vu-Brugier. Analysis of the impact of early fiber access deployment on residential internet traffic. *International Teletraffic Congress, Paris*, June 2009.
2. A. Pathak, Y. Wang, C. Huang, A. Greenberg, Y. Hu, J. Li, and K. Ross. Measuring and evaluating tcp splitting for cloud services.
3. P. Svoboda, F. Ricciato, W. Keim, and M. Rupp. Measured web performance in gprs, edge, umts and hsdpa with and without caching. *IEEE International Symposium on a World of Wireless, Mobile and Multimedia Networks, Helsinki*, pages 1–6, June 2007.
4. L. Plissonneau and G. Vu-Brugier. Mobile data traffic analysis: How do you prefer watching videos? *ITC*, 2010.
5. N. Kammenhuber, J. Luxenburger, A. Feldmann, and G. Weikum. Web search clickstreams. In *Proceedings of the 6th ACM SIGCOMM conference on Internet measurement*, IMC '06, pages 245–250, New York, NY, USA, 2006. ACM.
6. S. Stamou and L. Kozanidis. Impact of search results on user queries. In *Proceeding of the eleventh international workshop on Web information and data management*, WIDM '09, pages 7–10, New York, NY, USA, 2009. ACM.
7. Jitendra Padhye, Victor Firoiu, Donald F. Towsley, and James F. Kurose. Modeling tcp throughput: A simple model and its empirical validation. In *SIGCOMM*, 1998.
8. François Baccelli and David R. McDonald. A stochastic model for the throughput of non-persistent tcp flows. *Perform. Eval.*, 65(6-7):512–530, 2008.
9. Gregor Maier, Anja Feldmann, Vern Paxson, and Mark Allman. On dominant characteristics of residential broadband internet traffic. In *Internet Measurement Conference*, pages 90–102, 2009.
10. P. Romirer-Maierhofer, F. Ricciato, A. D'Alconzo, R. Franzan, and W. Karner. Network-wide measurements of tcp rtt in 3g. *TMA '09: International Workshop on Traffic Monitoring and Analysis, Aachen*, pages 17–25, May 2009.
11. S. Jaiswal, G. Iannaccone, C. Diot, J. Kurose, and D. Towsley. Measurement and classification of out-of-sequence packets in a tier-1 ip backbone. *IEEE/ACM Trans., Piscataway*, 15(1):54–66, 2007.
12. H. Jiang and C. Dovrolis. Passive estimation of tcp round-trip times. *SIGCOMM Comput. Commun. Rev., Pittsburgh*, 32(3):75–88, 2002.
13. S. Shakkottai, R. Srikant, N. Brownlee, A. Broido, and KC. Claffy. The rtt distribution of tcp flows in the internet and its impact on tcp-based flow control. Technical report number tr-2004-02, CAIDA, January 2004.
14. B. Veal, K. Li, and D. K. Lowenthal. New methods for passive estimation of tcp round-trip times. *PAM, Boston*, pages 121–134, 2005.
15. Y. Zhang, L. Breslau, V. Paxson, and S. Shenker. On the characteristics and origins of internet flow rates. *SIGCOMM '02, Pittsburgh*, pages 309–322, 2002.
16. A. Barbuzzi, G. Boggia, and L.A. Grieco. Desrto: an effective algorithm for srto detection in tcp connections. TMA '10, 2010.
17. A. Hafsaoui, D. Collange, and G. Urvoy-Keller. Revisiting the performance of short tcp transfers. *8th International IFIP-TC 6 Networking Conference, Aachen*, pages 260–273, May 2009.
18. http://homepage.tudelft.nl/19j49/t-SNE.html.
19. http://www.mathworks.com/help/toolbox/stats/kmeans.html.