



EURECOM  
Department of Networking and Security  
2229, route des Crêtes  
B.P. 193  
06904 Sophia-Antipolis  
FRANCE

Research Report RR-12-263

## **On the Relationship Between QoS and QoE for Web Sessions**

January 4<sup>th</sup>, 2012

Heng Cui, Ernst Biersack

Tel : (+33) 4 93 00 81 94  
Fax : (+33) 4 93 00 82 00  
Email : {firstname.lastname}@eurecom.fr

---

<sup>1</sup>EURECOM's research is partially supported by its industrial members: BMW Group, Cisco, Monaco Telecom, Orange, SAP, SFR, Sharp, STEricsson, Swisscom, Symantec, Thales.



# On the Relationship Between QoS and QoE for Web Sessions

Heng Cui, Ernst Biersack

## Abstract

Web browsing is a very common way of using the Internet to, among others, read news, do on-line shopping, or search for user generated content such as YouTube or Dailymotion. Traditional evaluations of web surfing focus on objectively measured Quality of Service (QoS) metrics such as loss rate or round-trip times; however, little is known how these QoS metrics relate to the user satisfaction, referred to as Quality of Experience (QoE). In this paper, we propose to use K-means clustering to discover the relationship between the subjective QoE and the objective QoS: Each Web session is described by a so called '*signature*' that consists of a set QoS metrics and the number of elements the Web page is composed of. In addition, we use a browser plugin to measure the time it takes to render the entire Web page (full load time) and ask the user to express via a feedback button its (dis-)satisfaction with the speed at which the Web page was rendered.

Clustering the Web sessions of multiple users based on their signatures allows to discover and explain the performance differences among users and identify the relationship between the QoS measured and the QoE experienced: User dis-satisfaction is often related to large round-trip delays, high loss rates, or Web pages with a large number of elements. We also see that there is a strong correlation between the full load time and the QoE: a full load time of ten seconds or more is typically not acceptable for the users.

## Index Terms

Web Browsing, Home Networks Measurement, Quality of Experiences, Quality of Service



# Contents

<b>1</b>	<b>Introduction</b>	<b>1</b>
<b>2</b>	<b>Measurement Set-up</b>	<b>1</b>
2.1	Architecture . . . . .	1
2.2	Metrics . . . . .	1
2.3	Methodology . . . . .	3
<b>3</b>	<b>Controlled Experiment</b>	<b>4</b>
3.1	Global Analysis . . . . .	4
3.2	Per-Page Analysis . . . . .	6
<b>4</b>	<b>Home Users Surfing in the ‘Wild’</b>	<b>8</b>
4.1	Global Analysis . . . . .	8
4.2	Per-Client Analysis . . . . .	10
4.3	Session Anomaly Analysis . . . . .	10
<b>5</b>	<b>Related Work</b>	<b>12</b>
<b>6</b>	<b>Conclusion and Future Work</b>	<b>13</b>

## List of Figures

1	Metrics Related to Download One Element In a Web Page . . . . .	2
2	Choice of Cluster Numbers (100 independent runs for each number setting) . . . . .	4
3	Clustering of All Ctrl. Sessions in 3 Homes . . . . .	5
4	Clustering of Sessions from Cluster 2 and 3 of Fig.3 . . . . .	6
5	All Controlled 'Google' Sessions in 3 Homes . . . . .	7
6	Clustering of All <b>Wild</b> user Sessions in <b>4</b> Clusters . . . . .	9
7	Performance Comparisons between 'Good' and 'Poor' Sessions for Different Users . . . . .	11
8	Session Anomaly Analysis for <b>Wild</b> Users . . . . .	12

# 1 Introduction

Web browsing is a very common way of using the Internet access as it allows to access to a wealth of information. Since there is a “human in the loop”, the time it takes to render a Web page should be small in order to assure a good user experience (QoE).

Traditional evaluations of the web surfing mainly use Quality of Service(QoS) metrics that are easy to measure such as packet loss rate or round trip times. However, in recent times the measurement community tried to “move up the stack” and try to capture the real user experience, referred to as Quality of Experiences(QoE). In 2011, SIGCOMM even organized a special workshop discussing the problems in measuring user experience<sup>1</sup>.

In this paper, we propose a methodology to evaluate user QoE during web surfing and its relationship to QoS, based on clustering of QoS data. We also demonstrate that this clustering is suitable to compare and explain the experience among different clients and Web pages.

## 2 Measurement Set-up

### 2.1 Architecture

Based on our previous work [1], we set up an experimental platform that uses a firefox plugin and normal packet capture to measure both, QoE related information such as user satisfaction and QoS related information such as full page load time, RTT, or packet loss rate.

- The firefox plugin, among others, enables the user to express his **dis-satisfaction** with the time it took to render a Web page by clicking on an icon embedded in the browser. We also use the plugin to bind each HTTP query initiated by the browser to its associated firefox window/tab, which makes it easy to associate queries with the web sessions of the users and to measure the full load time for that Web page.
- We use packet capture (libpcap format) to obtain raw packet traces that are loaded into a database for post-processing. For details about the architecture and how we combine the measured records, we refer the readers to our previous work [1].

### 2.2 Metrics

A typical web page can contain up to hundreds of elements. To fully render the Web page, the browser needs to load all these elements. The typical procedure for loading one element is shown in Fig.1. As is shown in Fig.1(a), for the download of each element, we extract from the packet trace the following metrics:

---

<sup>1</sup><http://conferences.sigcomm.org/sigcomm/2011/workshops/W-MUST/>

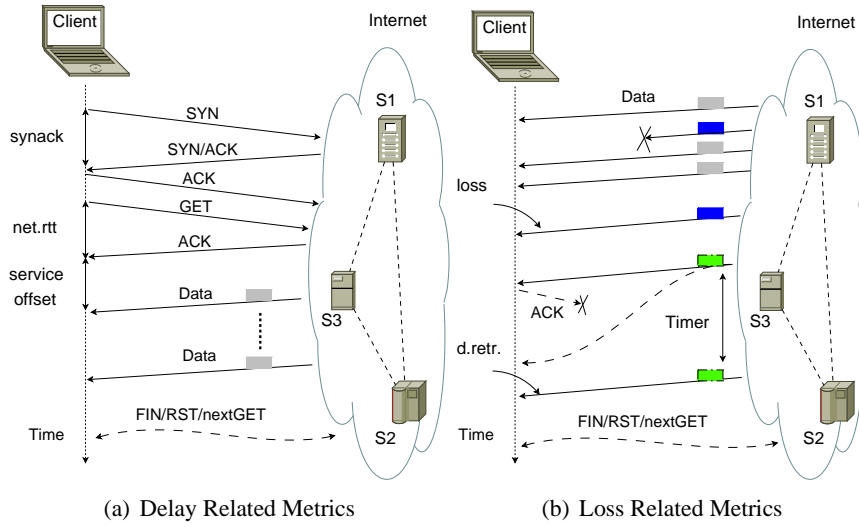


Figure 1: Metrics Related to Download One Element In a Web Page

‘**synack**’ is defined as time elapsed between the first client SYN packet and the corresponding ACK. ‘**net.rtt**’ is defined as time elapsed between the first GET query and the corresponding ACK. These two metrics can be considered as the TCP connecting and HTTP query delays observed by the end client. We also measure the TCP handshake **rtt** where retransmissions of the TCP handshake packets will be excluded, and it can be considered as the round-trip delay between the client and web servers <sup>2</sup>.

‘**service offset**’ is defined as the interval between the ACK of the GET request and the first data packet observed; in case that the ACK already carries a payload, ‘service offset’ will be set to zero. Besides the related delay metrics, we also compute the data loss and retransmissions from the server to the client as is shown in Fig.1(b). To estimate the retransmissions, we use method similar to [2], which is based on the TCP sequence number and the IPID field. The only difference is that, whenever we observe a retransmitted data packet, if such packet is not observed before, we consider it a **loss**; otherwise a **duplicate retransmission**.

To describe the QoS of a Web session, we use the metrics computed for each Web element and compute the **mean** over all the values of the given metric to obtain a **Key Performance Index(KPI)**, which consists of

$$[nr., SYNACK, NET.RTT, SERV.OFF., RTT, LOSS, D.RETR.]$$

Note that *nr.* is the number of distinct GET requests during a complete web session, which provides an estimation of the size of the Web page size in terms of

<sup>2</sup>In case of a proxy terminating the connection requests of the client, the measured rtt will only refer as round-trip time between the client and the proxy, and not between the client and the server. However, in this paper, we do not discuss the proxy case studies.



the number of elements. We use upper case letter to denote means and lower case to denote individual sample values.

We label the average rate of such two types of retransmissions during a web session as **LOSS** and **D.RETR** in the signature. ‘LOSS’ can be used to estimate up-stream packet loss; since our measurement point is the client host, ‘D.RETR’ can be caused by reverse ACK loss, or too short retransmission timers at the server side, etc.

Meanwhile, for the KPI, we focus on metrics captured by TCP connections and ignore DNS queries. DNS pre-fetching is widely supported by recent browsers and this causes DNS queries to occur before a real web page surfing, which makes the lookup times less useful. However, we plan to study the effects of DNS (e.g. response time, response IP, TTL, etc.) on the web browsing experiences in the future.

### 2.3 Methodology

To process our results and compare results among different homes, we use clustering. Since some of the values in the KPI are in different units and have different value ranges, we normalize all into the range [0,1]. For each type of metric  $i$  in the signature, we use  $\frac{x_i - \min_i}{\max_i - \min_i}$ , where  $x_i$  is the raw value for metric  $i$  in the signature;  $\max_i$  and  $\min_i$  are *maximum* and *minimum* values for that metric  $i$  respectively.

To cluster the results, we choose the well known *kmeans* algorithm, which is an un-supervised classification algorithm that does not need any training. The tricky point in *kmeans* is how to set a-priori the number of clusters: Hafsaoui et al. [3] use the dimension-reduced (t-SNE) method to determine cluster numbers, while other papers [4] [5] propose different algorithms for automatically detecting the right number of clusters. However, these methods are either based on pure visual inspection or introduce some extra parameters.

In our case, we carry out a comparison of the results obtained for different number of clusters. We define as **error distance** the squared euclidean distance between each sample in a cluster and its corresponding centroid. Fig. 2 shows the average error over all samples for different numbers of clusters based on the data sets we will use in this paper. We see that for one or two clusters, the average error distances are relatively larger, and that the error distance rapidly decreases for three clusters or more. This observation indicates that a number of clusters which is between 3 and 5 is sufficient to achieve small clustering errors. Based on the above discussions, we use *four clusters throughout the paper*. Since the initial centroids are chosen randomly, in order to achieve locally optimal clustering results, we run the *kmeans* algorithm ten times and keep as result the one with smallest distance error.

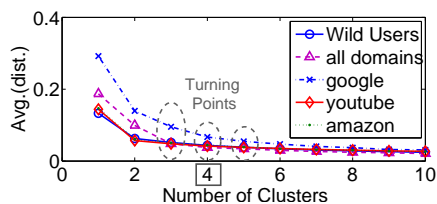


Figure 2: Choice of Cluster Numbers (100 independent runs for each number setting)

### 3 Controlled Experiment

In this section, we predefine a list of web pages, namely `www.google.fr(CDN)`, `www.amazon.fr(FR)`, `www.orange.fr(FR)`, `www.ebay.fr(FR)`, `www.youtube.com(CDN,US)`, `ccr.sigcomm.org/online(US)`, `www.sina.com(CN)`, `www.baidu.com(CN)`, `www.sohu.com(CN)`, `www.163.com(CN)`. These web pages are diverse in geographical location and web page type.

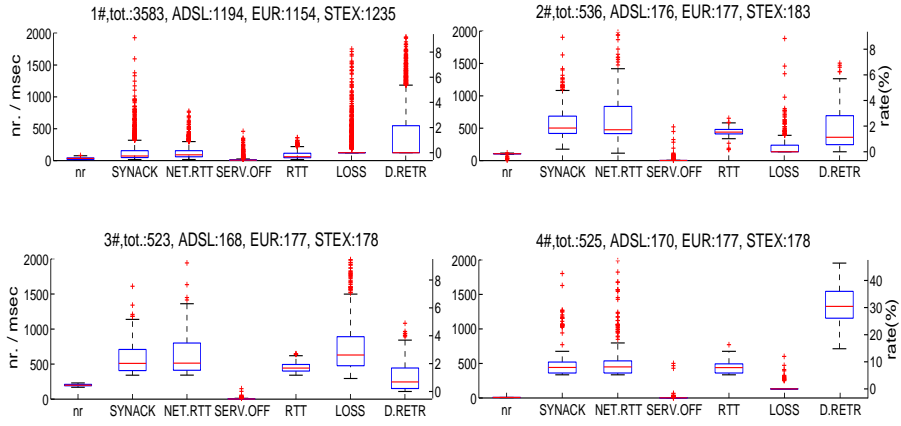
We emulate user browsing with three different clients that are (i) at home connected via an ADSL connection, (ii) in the office at Eurecom connected via a 100 Mb/s link to the Internet and (iii) in a student residence. All the machines are located in France. We name the ADSL home, the Eurecom office Ethernet and the student residence connections as ‘**ADSL**’, ‘**EUR**’, and ‘**STEX**’ respectively. The experiments are done during the same evening in three homes and lasted for around 8 hours each. Both, the ‘EUR’ and ‘STEX’ client computers are connected via a wired connection, while the ‘ADSL’ client computer is physically very close to the Access Point and uses a wireless connection. We clear the browser cache at the end of each Web session.

#### 3.1 Global Analysis

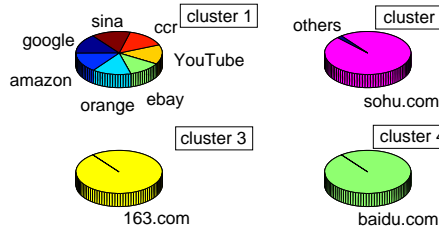
For all the emulated web sessions in three different locations we process and cluster the original data and present the results using boxplots<sup>3</sup>, which provide the median and lower/upper quartile of a given metric. Fig.3(a) shows the global clustering results of KPI values. Fig.3(b) shows the web page distribution as pie-charts, and Fig.3(c) shows the page load time of the different clusters. In Fig.3(a) (also following figures), we indicate the cluster ID number by a number followed by a ‘*sharp*’ sign in the title, and also indicate the number of sessions for a given location that are grouped in that cluster.

In Fig.3(c) we see that the Web pages in clusters 1 and 4 have page load times that are typically in the order of a few seconds or less, while the Web pages in clusters 2 and 3 have page load times of 10 seconds and more. Clusters 2 and 3

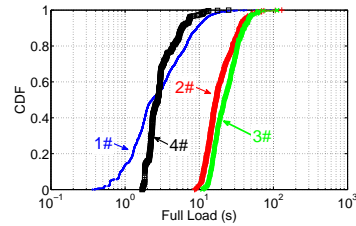
<sup>3</sup><http://www.mathworks.com/help/toolbox/stats/boxplot.html>



(a) Ctrl. Exp. Clustering Signatures



(b) Web Page Distributions in the Clusters

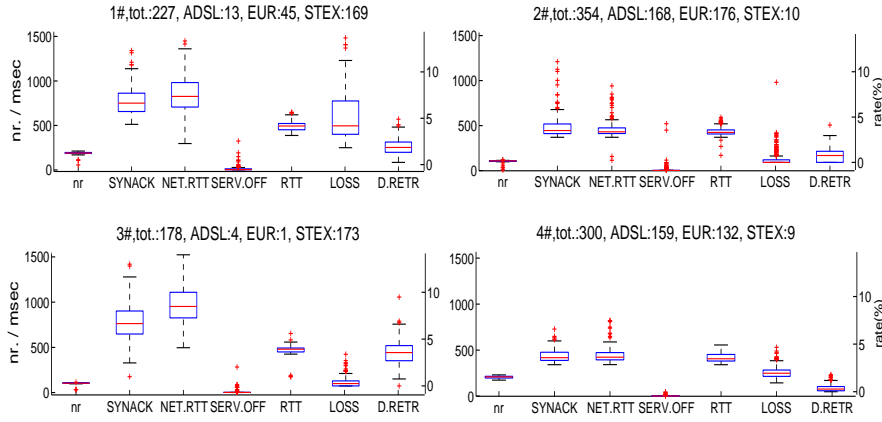


(c) Page Load Time

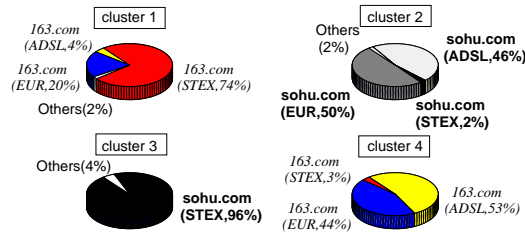
Figure 3: Clustering of All Ctrl. Sessions in 3 Homes

contain almost exclusively requests to a single Web page (see Fig.3(b) ), namely `sohu.com` and `163.com`, which are both in China. The RTTs in these two clusters are normally around 500ms and transfers in cluster 3 experience higher packet loss than in cluster 2; also the TCP handshake and HTTP query delays are large (see From Fig.3(a)).

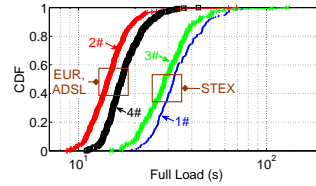
In cluster 4, the delay metrics for RTT, SYNACK and NET.RTT take values around 500ms, while there is almost no loss; all the requests go again to a single Web page in China (`www.baidu.com`). However, data transfers in this cluster experience an un-usual large number of duplicated retransmissions, which are as high as 30%. Duplicated retransmissions can be caused by reverse ACK loss, congestion, or a too short retransmission timeout at the server. A more detailed investigation reveals that the time interval between two successive data packets with the same sequence number is around 200ms, while the RTT is around 500ms, i.e. the Web servers of `www.baidu.com` use too short retransmission timeouts, which results in a lot of spurious retransmissions. However, these premature timeouts do not seem to have a negative incidence on the full load time, which is four seconds or less for 80% of the Web sessions. Note also that the number of elements for this



(a) Ctrl. Exp. Clustering Signatures



(b) Web Page Distributions in the Clusters



(c) Page Load Time

Figure 4: Clustering of Sessions from Cluster 2 and 3 of Fig.3

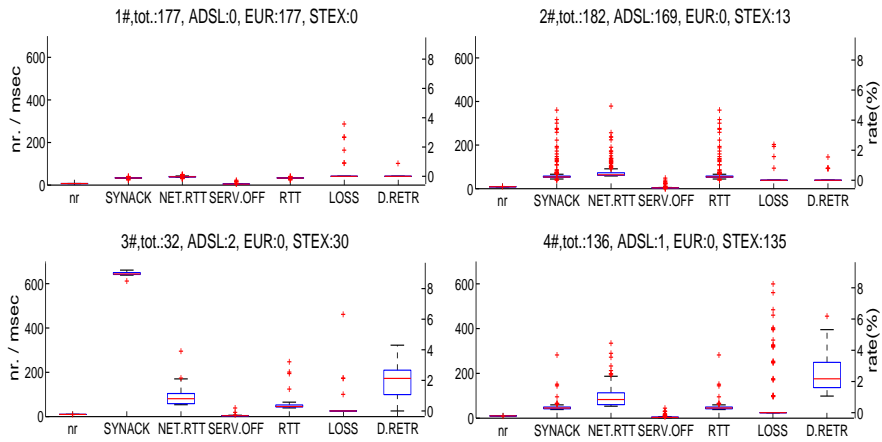
Web page is very small.

Cluster 1 contains all the other Web pages such as `google`, `amazon`, `ebay`, `orange`, `sina.com`. Web pages in this cluster normally have a small number of elements and their delay and loss metrics for this cluster are low. The whole page can be fully loaded within 10 seconds in 90% of the cases.

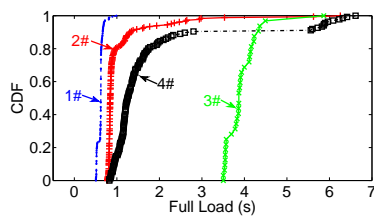
When we use a different number of clusters such as 3 or 5, the clustering still succeeds in isolating the Web pages with large full load times of 10 seconds or larger in a separate cluster. We do not show the details for space reasons here.

### 3.2 Per-Page Analysis

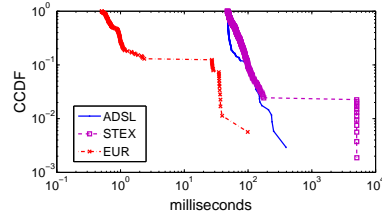
Since our experiment was done at three different locations, it allows to compare the performance achieved by the different clients accessing the same web page. We saw in Fig.3 that cluster 2 and 3 contain mostly requests to two Chinese Web pages, `sohu.com` and `163.com`. We now take all the sessions in these two clusters and apply once more clustering based on the KPIs for these Web pages in order to reveal the impact of the specific network access on the overall performance (See



(a) 'Google' Clustering Signatures



(b) 'Google' Page Load Time



(c) 'Google' DNS Response Time

Figure 5: All Controlled 'Google' Sessions in 3 Homes

Fig.4). The accesses to the `sohu.com` Web page are in clusters 2 and 3, while all the accesses to the `163.com` Web page are all in clusters 1 and 4. The two clusters with the highest full load times are cluster 1 and 3. The client who issued the Web requests grouped in these two clusters is predominately the 'STEX' client who shares a congested access link with the other inhabitants of the same student residence (Fig.4(b)).

As another illustration of how the comparison of the performance of the access to *the same Web page across different clients* helps identify the influence of problems specific to a client, we use the `Google` Web page and show the results in Fig.5. We know that Google works very hard to keep the page download times as low as possible by placing servers close to the clients and also by keeping the number of elements of its Web page low. Among the four clusters, requests from 'EUR' are grouped in cluster 1 and from 'ADSL' are grouped in cluster 2. On the other hand, the requests in clusters 3 and 4 are issued almost exclusively from the 'STEX' client. The fact that the 'STEX' client experiences higher delays and also loss in his local access can be seen in his KPIs. Cluster 3 is interesting because of its large SYNACK values; it turns out that for cluster 3 on average one out of

5 SYN requests to establish a TCP connection does not get answered and must be retransmitted. Since the retransmission timeout is 3 seconds, we get  $\text{SYNACK} = \frac{(50+50+50+50+3050)}{5} = 650(ms)$ .

The long tail for the page load times in cluster 4 is due to the long DNS response time for some of the sessions. As we have already discussed, the ‘STEX’ client experiences more packet loss than the other two clients. We can clearly identify in Fig.5(c) that around 3% of the DNS queries need to be retransmitted for the ‘STEX’ client.

## 4 Home Users Surfing in the ‘Wild’

So far, we have predefined a number of Web pages that were accessed from different locations. We now present a more “realistic” case, where clients are free to access whatever Web pages they want<sup>4</sup> and show how to apply clustering in this case.

Again, we pick three different locations, two of them are named ‘ADSL’ and ‘STEX’ which are the same clients from previous section, located in France. A third user, called ‘TO’ is located Torino, Italy. ‘ADSL’ client uses a WLAN in his home, while ‘STEX’ and ‘TO’ users have wired connections. These users randomly browse web pages with diverse geographical locations and are asked to click on the feedback button of the browser plugin whenever they are dis-satisfied with the time it took to render the Web page. We call a web session ‘**poor**’ when the users clicks on the ‘dis-satisfied’ feedback button and ‘**good**’ otherwise.

### 4.1 Global Analysis

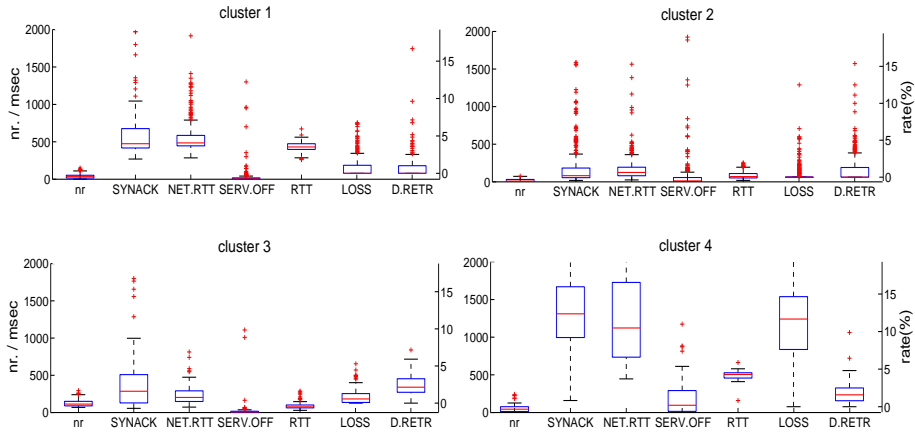
We first take all the web sessions of the ‘wild’ users and cluster them globally. Fig.6(a) shows the performance metrics for the different clusters. Since different users do not necessarily access the same Web pages, we provide geographical information to classify the Web pages. We use the country and organization database from maxmind<sup>5</sup>, and for each cluster, we group all the requests and add the country name based on the IP address of the Web server. Since for Google, maxmind always returns US as country, we prefer to identify Google separately labeled as ‘Google’. Fig.6(b) shows the geographical distribution via pie charts and 6(c) shows the page load times for the different clusters. Tab.1 also lists median values for the different metrics and the number of ‘poor’ and ‘good’ sessions.

The Chinese Web pages are all concentrated in cluster 1 and 4, while the others are in clusters 2 and 3.

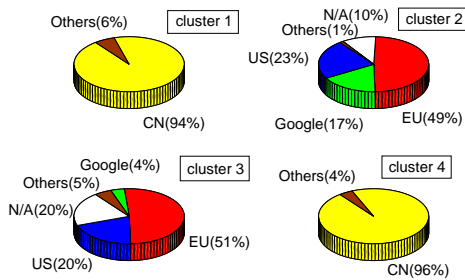
The Web pages in cluster 2 have typically short full load times which is for more than 80% of these sessions below 10 seconds. The KPI metrics related to

<sup>4</sup>We focus on traditional Web pages and not personalized ones such as facebook, gmail, etc.

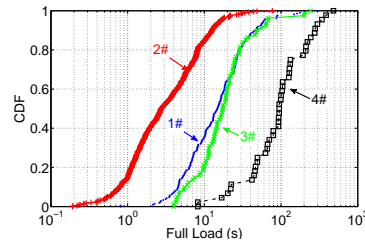
<sup>5</sup><http://www.maxmind.com/>



(a) Wild User Clustering Signatures



(b) GeoIP in Different Clusters



(c) Page Load Time in Clusters

Figure 6: Clustering of All **Wild** user Sessions in 4 Clusters

delay and loss are all quite low. In only 6% of the cases the users were dis-satisfied with the time it took to render the page, which is the lowest number of all clusters.

On the other extreme, users were dis-satisfied in 64% of the cases with the rendering time of Web pages in cluster 4, which is not surprising, since 90% of these Web sessions need more than 10 seconds to fully load, and 50% of the sessions need even more than 100 seconds. The KPI metrics related to delay and loss are very high: The median delay for TCP handshake and HTTP query are over 1 second and the median loss rate is above 10%. After checking the details of the web sessions in this cluster, we find most of them access the domain ‘\*.mop.com’ (e.g. www.mop.com, game.mop.com, etc.)

Access to Web pages in Cluster 1 and 3 exhibits similar full page load times and are rated as poor 21% and 42% of the time, respectively. However, the characteristics of the Web pages and the KPIs for these two clusters are different: Web pages in cluster 3 normally have more elements than in cluster 1. From Tab.1 we also see that the median RTT in cluster 3 is around 70ms while for cluster 1 it is normally

Table 1: Statistics of the four Clusters (*TOT* and *poor%* refer as total number of web sessions and the percentage of ‘poor’ sessions in that cluster, respectively).

clusterID	Median values (units for delay metrics are ‘milliseconds’)							TOT	poor%
	nr.	SYNACK	NET.RTT	SERV.OFF	RTT	LOSS	D.RETR		
Cluster 1	34	474	485	3	431	0.0%	0.0%	209	21%
Cluster 2	10	82	124	12	70	0.0%	0.0%	399	6%
Cluster 3	113	285	202	8	76	0.6%	2.1%	81	42%
Cluster 4	44	1310	1122	96	505	11.7%	1.6%	44	64%

around 400ms, which is not surprising since almost all the Web pages in cluster 1 are in China(CN). The median of packet loss and duplicated retransmission rates in cluster 3 are a bit higher than in cluster 1.

## 4.2 Per-Client Analysis

Since user dis-satisfaction is to a certain extent subjective, *we want to compare the KPIs of good and poor Web sessions on a per-user basis to see if we can identify commonalities among the different users.* The results are shown in Fig.7:

For the ‘TO’ client, the performance differences between ‘poor’ and ‘good’ sessions are obvious: median SYNACK, NET.RTT, and RTTs for ‘poor’ sessions are around 500ms, while for ‘good’ sessions, these values range from tens of milliseconds to 500ms for the upper quartile. Loss and duplicated retransmission rates are slightly higher for ‘poor’ sessions than ‘good’ sessions.

For the ‘STEX’ client, the situation is similar to the ‘TO’ client: all the delay related metrics are much higher for the ‘poor’ sessions than for the ‘good’ ones, the same holds for the loss rates. If we compare the number of elements a Web page is composed of, we see that Web sessions rated as ‘poor’ have a higher number of elements.

The ‘ADSL’ user never expressed any dis-satisfaction, which can be expected if we look at its KPIs, which are largely comparable and often even better than the ones for the ‘good’ Web sessions of the ‘STEX’ and ‘TO’ client.

Finally, if we look at the full load times (c.f. Fig.7(d)) we see that the CDFs of the ‘poor’ and the ‘good’ session for both, the ‘TO’ and the ‘STEX’ client are very similar. Also around 90% of the Web sessions rated ‘poor’ take over *10 seconds* to fully, while for all three clients between 70% and 80% of the session rated as ‘good’ take less than 10 seconds to get fully loaded.

## 4.3 Session Anomaly Analysis

So far we used clustering to study the correlation between KPIs of a web session and the user subjective experience. We found a strong correlation between high values for some of the KPI metrics and user dis-satisfaction. In this subsection, we focus on the **anomalies** in the collected ‘wild’ user web sessions. For each of the KPI metrics, we use the 80-th percentile as a threshold; if the value of



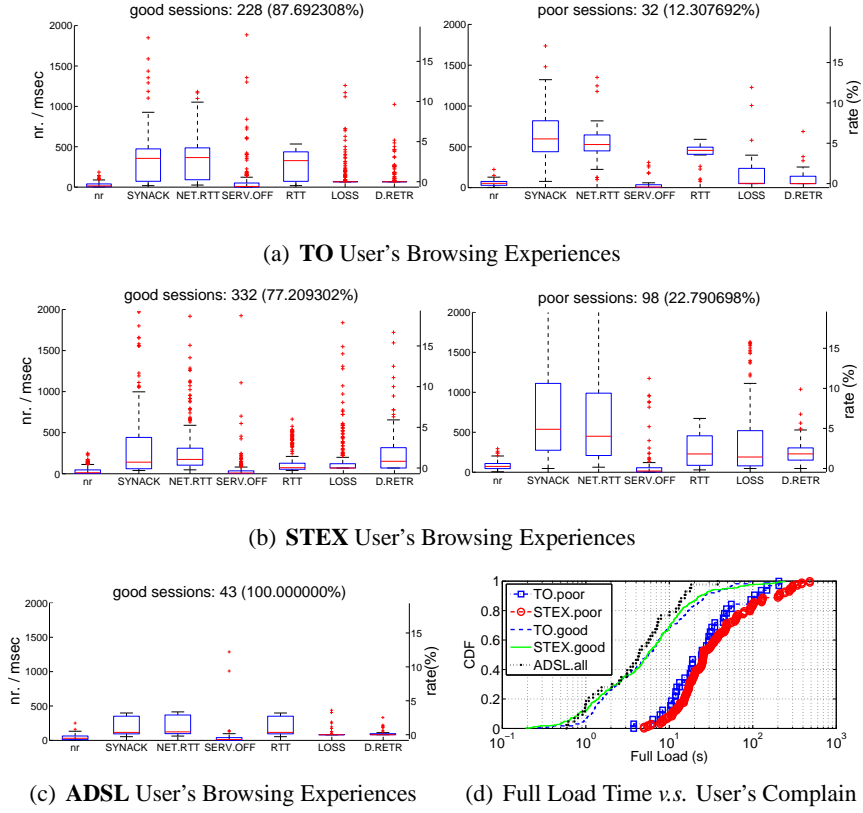


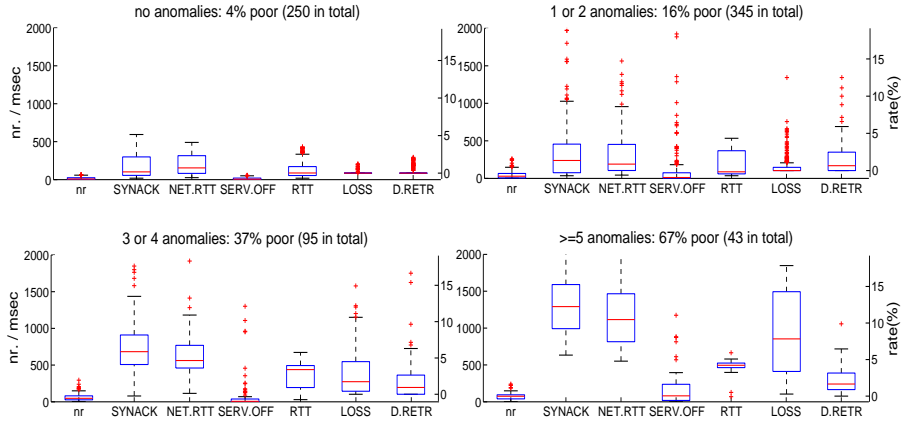
Figure 7: Performance Comparisons between ‘Good’ and ‘Poor’ Sessions for Different Users

a given metric is larger than its 80-th percentile, we consider that value as anomalous. For each web session in our ‘wild’ home user measurements, we count the number of anomalies in the KPI metrics and group the Web sessions by the number of anomalies’ into: ‘no anomalies’, ‘1 or 2 anomalies’, ‘3 or 4 anomalies’ and ‘more than 4 anomalies’ to study the relationship between the number of anomalies and user dis-satisfaction. Fig.8(a) presents the KPIs as boxplot, Fig.8(b) gives the geographical distribution of the web pages in each group and Fig.8(c) shows the page load times.

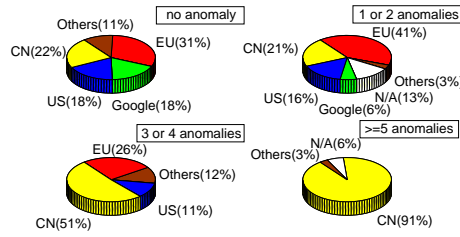
We can see that Web sessions with three and more anomalies have full load times that are typically larger than 10 sec. resulting very frequently in user dis-satisfaction. These Web sessions are also characterized by high values for the KPIs related to delay and loss and most of these Web pages are hosted in China.

For web sessions with no anomalies, we can see that the Web pages are normally small in terms of number of elements, also KPI metrics such as delays and loss are small. Page load times for these sessions are in nearly 90% of the cases below 10 seconds, resulting a very low percentage of user dis-satisfaction.

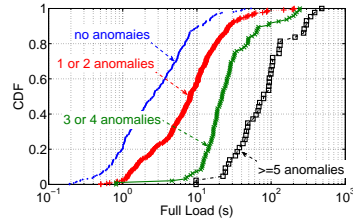
While the percentile-based method is complementary to the clustering approach,



(a) Session Signature in Different Anomalous Groups



(b) GeoIP in Different Anomalous Groups



(c) Page Load Time

Figure 8: Session Anomaly Analysis for **Wild** Users

our goal remains the same, namely to group web sessions according to their KPI values and to correlate user satisfaction with the values for the KPIs. Moreover, the clustering technique and percentile-based method are related. Tab.2 shows the distribution of the number of anomalous KPIs for the different clusters presented previously in Fig.6. We can clearly see that the Web sessions in cluster 2, which have the lowest percentage of dis-satisfaction, experience no or very few anomalies. A similar observation is true for the Web sessions in cluster 4, which have the largest percentage of poor Web sessions, and also the largest number of anomalies.

## 5 Related Work

This work builds on our previous work [1] where we presented the measurement architecture but did not carry out any systematic analysis of the measurements.

Different methodologies of collecting user feedback are proposed in the literature. Chen et al. [6] propose the ‘OneClick’ platform, which carefully evaluates the

Table 2: Relation Between Clusters in Fig. 6 and the Number of Anomalies

	Number of anomalies			
	0	1 or 2	3 or 4	$\geq 5$
cluster 1	22%	45%	28%	5%
cluster 2	51%	47%	2%	0%
cluster 3	0%	77%	21%	2%
cluster 4	0%	2%	30%	68%

correlation between the user click rate and some controlled network performance metrics such as packet loss etc. Joumblatt et al. [7] propose HostView, which collects user feedback via an ‘annoyed’ clicking icon and a feedback form; a follow up paper [8] discusses lessons learned and some preliminary results collected via HostView.

Compared to these works, our work focuses only on Web browsing, the properties on the web pages, and the relation between KPIs and user satisfaction. Our analysis methodology is also inspired by work of A. Hafsaoui [9], where methods similar to ours are used to analyze the performance of TCP connections.

## 6 Conclusion and Future Work

We have studied for the case of Web browsing the relationship between low level KPI metrics such as delay and loss and user satisfaction. We use both, controlled browsing and ‘wild’ browsing experiments. We show that (i) by sharing KPIs of web sessions among different ‘home clients’, clustering techniques allow to discover performance differences among Web sessions and the root causes for poor Web sessions; (ii) when focusing on poor Web sessions only, clustering or percentile-based methods allow to explain user dis-satisfaction by looking at the KPIs of the different clusters; (iii) by comparing the values of the KPIs for good and poor Web sessions we can identify the reasons for dis-satisfaction.

As a first extension, we need to solicit more users run our system in order to consolidate our results. We also need to check how the number of clusters for the kmeans algorithm is affected by the number of users that participate in the experiment.

As future work, we also plan to extend our system to work in real time and in a distributed fashion but making the agents in the different locations communicate and perform distributed computations such as distributed clustering.

We have seen that there is a strong correlation between large delays or high loss rates and user dis-satisfaction. However, *correlation does not imply causality*. There are, for instance, many reasons for experiencing large delays or high loss rates such as misconfigured DNS servers, a large buffer at the access link, an overloaded proxy between the client and server or simply the large geographical

distance between the client and the server. We plan to use the framework developed by J. Pearl [10] to study causal relationships.

## References

- [1] H. Cui and E. Biersack, “Trouble Shooting Interactive Web Sessions in a Home Environment,” in *ACM SIGCOMM Workshop on Home Networks*, Toronto, Canada, August 2011.
- [2] M. Siekkinen, G. Urvoy-Keller, E. W. Biersack, and D. Collange, “A Root Cause Analysis Toolkit for TCP,” *Computer Networks*, vol. 52, no. 9, pp. 1846–1858, 2008.
- [3] A. Hafsaoui, G. Urvoy-Keller, D. Collange, M. Siekkinen, and T. En-Najjary, “Understanding the impact of the access technology: the case of web search services,” in *TMA 2011, third International Workshop on Traffic Monitoring and Analysis*, Vienna, Austria, April 2011.
- [4] A. Finamore, M. Mellia, and M. Meo, “Mining unclassified traffic using automatic clustering techniques,” in *TMA 2011, third International Workshop on Traffic Monitoring and Analysis*, Vienna, Austria, April 2011.
- [5] D. Pelleg and A. Moore, “X-means: Extending k-means with efficient estimation of the number of clusters,” in *Proceedings of the Seventeenth International Conference on Machine Learning*. San Francisco, USA: Morgan Kaufmann, 2000, pp. 727–734.
- [6] K. Chen, C. Tu, and W. Xiao, “OneClick: A Framework for Measuring Network Quality of Experience,” in *Proceedings of IEEE INFOCOM 2009*, Rio de Janeiro, Brazil, April 2009, pp. 702–710.
- [7] D. Joumlatt, R. Teixeira, J. Chandrashekar, and N. Taft, “HostView: Annotating End-host Performance Measurements with User Feedback,” in *Hot-Metrics, ACM Sigmetrics Workshop*, New York, NY, USA, June 2010.
- [8] D. Joumlatt, R. Teixeira, J. Chandrashekar, and N. Taft, “Performance of Networked Applications: The Challenges in Capturing the User’s Perception,” in *ACM SIGCOMM Workshop on Measurements Up the Stack*, Toronto, Canada, August 2011.
- [9] A. Hafsaoui, “Application level performance in wired and wireless environments,” Ph.D. dissertation, TELECOM ParisTech, 2011. [Online]. Available: [http://www.eurecom.fr/~hafsaoui/these\\_aymen\\_hafsaoui.pdf](http://www.eurecom.fr/~hafsaoui/these_aymen_hafsaoui.pdf)
- [10] J. Pearl, *Causality Models, Reasoning and Inference 2nd Edition*. Cambridge University Press, 2009.