An Empirical Study of Web Quality: Measuring the Web from Wroclaw University of Technology Campus

Leszek Borzemski and Ziemowit Nowak

Institute of Control and Systems Engineering
Wroclaw University of Technology
Wybrzeze Wyspianskiego 27
50-370 Wroclaw, Poland
e-mail: {leszek; znowak}@ists.pwr.wroc.pl

Abstract. This work presents an empirical study on Web quality measurement. We evaluate the performance and reliability of Web as perceived by the end users located at the Wroclaw University of Technology (WUT) campus. The active measurements are performed periodically for a set of Web servers mirroring the same data and localized in different parts of the Internet. We report the results of a series of experiments performed by means of the Wing measurement infrastructure. Wing system has been developed by us for probing, visualization and performance analysis of Web site from the user perspective. It uses a real Web browser, contrary to other measurement systems that use their own browsing mechanisms. Therefore, the measurement results made by Wing are realistic. The measurements presented in this paper were performed using MS Explorer. Based on the measurements that we have analyzed so far, it is inconclusive to say that the round-trip time can be a good predictor of HTTP throughput in general. The distribution of HTTP throughput versus TCP round-trip time as seen from Wroclaw site can be described using power law of the form $y=kx^{\alpha}$ with k and α determined experimentally: k=46456 and α=-0.8805.

1 Introduction

End users perceive good Web quality in the context of good performance, availability, security and accessibility. Various factors impact the performance and reliability of individual Web service such as: network solutions, Web site solutions and infrastructure solutions (DNS resolution, caching, traffic shaping, content distribution networks, load balancing, etc.). Web quality is extremely difficult to study in an integrated way. It has never been easy to determine whether bad performance or non-availability of service is due to either network problems or end-system problems on both sides, i.e. user and server sides, or both. Moreover, because most of these performance problems are transient and very complex in the relationships between different factors that may influence each other, we cannot exactly diagnose and isolate their key sources. These factors may affect ultimate performance and reliability of Web page downloading. Generally, end users require reliable and efficient Web service and they are interested in fast downloading of entire pages. Therefore, the users perceive Web quality mostly by latency and throughput. Almost 60% latency, as perceived by end-users at their microscopic level while accessing the Web server by the browser, refers to the network latency that is the delay between sending the request for data and receiving (the first bit of) the reply [7]. The lower the latency, the faster we can do low-data activities. The other key element of network performance, throughput, also affects Web applications. Throughput is the “network bandwidth” metric which tells about the actual number of bytes transferred over a network path during a fixed amount of time. Throughput determines the "speed" of a network as perceived by the end user. The higher the throughput of Internet connection, the faster user can surf the Internet.

This paper presents a methodology and an empirical study of Web quality. The main goal of our work is to answer for a question whether is it possible to develop a model describing general Internet performance for the users surfing the Web from some site. Our model takes into account TCP round-trip time and HTTP throughput. We investigate the correlation between a TCP connection’s RTT and HTTP throughput to examine whether connections with shorter RTTs tend to transfer more data. We measure the HTTP throughput and TCP RTT from the Wroclaw University of Technology campus to the set of worldwide Web sites. Sufficient data were gathered. We pooled 83 Web sites on the Internet over a period of several weeks and then created the aggregate performance characteristics of Web as seen from the perspective of local users (the throughput v. round-trip time function with experimentally determined parameters). We also evaluate Web quality indicators in the field of the reliability of Web transactions and availability of Web servers. We determine the transaction reliability ratios for Web servers and the “mortality” rate of observed URL links.

We measured speed and latency using active measurements from our site towards the precisely defined set of Web servers. We decided to download periodically the specific file that has been found in several non-commercial sites that usually have had non-overloaded Web servers. The measurements are made at the WUT side. Generally, throughput and latency can be defined and measured in different ways. Usually, the latency is measured by ping and traceroute tools. They measure latency by determining the time it takes a given small ICMP packet to travel from source to destination and back, the so-called round-trip time (RTT). RTT is not the only way to specify latency, but it is the most common. Unfortunately ping-based technique is not very useful in Web as ICMP packets do not match usual Internet traffic. These packets can also be blocked by firewalls. The routers often provide different prioritizing for ICMP packets than for “normal” traffic, e.g. for TCP sessions in HTTP transfers. Here we use our approach to estimate RTT within TCP sessions used for HTTP transfers. Our RTT estimation technique is based on the measurements of time spacing between the SYN packet sent by the client and the SYN-ACK packet received in the reply.

In frequent HTTP throughput tests (e.g. stress tests) multiple clients send simultaneous HTTP requests to a Web server. Our approach is quite different since, as we want to estimate the HTTP throughput at the Transport Layer. Then we can evaluate Web transfer speed in more detailed manner without browser and processing overhead. The use of TCP connections in browsing can have performance implications (e.g. persistent connections in HTTP/1.1 usually improve Web transfer speeds).

The throughput network tests commonly use connectionless traffic like IP or UDP packets. Considering that at least 90% of Internet traffic uses TCP protocol (which is generated mostly by HTTP clients and servers), this is a rather large oversight in the context of Internet throughput testing and measuring [24]. To estimate the actual transfer rate of Web object in the TCP connection used by getting this object we measure time spacing between the first byte packet and the last byte packet of the object received by browser that use that connection. Transfer rate is calculated by dividing a number of bytes transferred by that time. The throughput measured by the test is the amount of traffic available at the application level, i.e. IP, TCP and HTTP headers are not included into the measurement.

The measurement infrastructure is built around the Wing system which has been developed by us for Web probing, visualization and performance analysis [4].

The remainder of this paper is organized as follows. In section 2 we introduce the goals of our project and review the current state of Internet and Web measuring. Section 3 contains a description of the methodology and measurement infrastructure used in our study. In section 4 we present and discuss the experimental results. Finally, concluding remarks are given in Section 5.

2 Related Work and Background

We measured the data to deal with the following problems:

1) Analysis of Internet performance characteristics (indicated by the relationships between HTTP throughput and round-trip time) – we show how the Web is perceived from the perspective of the end-users localized within the WUT local area network.
2) Analysis of the reliability of Web and the “mortality” of Web pages perceived from the perspective of the end-users localized within the WUT local area network.
3) Analysis of communication network behavior and prediction of the browser-to-Web server throughput using data mining techniques.
4) Performance estimation of Web service which performance is not directly measured.

In this paper, we study only problems (1) and (2). Results from research (3) and (4) are to be reported elsewhere. We used the data mining approach in the Internet performance analysis but in the case of other measured data set concerning host-to-host performance characteristics at the IP protocol layer (we used Traceroute). Our general strategy involves discovering knowledge that may characterize performance behavior of Internet paths, and then making use of this knowledge to guide future user network usage [3]. Data mining and knowledge discovery in computer networks are of great topical interest of early works [9], [11], [19].

The main measurement and analysis problem is that the Web is big, distributed and volatile. Currently, regular measurements are mainly performed within large Internet projects by means of complex measurement infrastructure, especially in the core of Internet [1], [2], [5], [8], [15], [16], [17], [19], [21], [24], [23], [29], [30]. They are primarily focused on the analysis of end-to-end behavior of the IP and BGP communication networks. The behavior of Internet paths is analyzed in the context of path lengths, path asymmetries, out-of-order packet delivery, packet corruption, available bandwidth, latency, and loss of links. To reduce the size of the problem these projects handle the limited numbers of Internet servers. Passive and active measurement techniques can be used. Most of active probing projects are based on the round-trip time (RTT) evaluation performed using ping or similar measurement tools. Such measurements can present some "network weather" conditions for the Internet related to the IP protocol. For example, the ping utility (which is ICMP-based) determines if the host is reachable and measures the round-trip time and packet loss on the network path from current host to the target host. The RTT measurements made by ping can be used to estimate the "pipe" capacity between both hosts at the IP layer. Such projects do not directly address Web. Some projects include measurement tools for discovering the network characteristics at the TCP layer [13]. Such measurements can be utilized in the evaluation of Internet at the Web (HTTP) layer, because Web basically uses TCP protocol.

Passive measurements are mainly based on network analyzers commonly called sniffers that capture and analyze network traffic. Passive monitoring that originated with TCPdump protocol packet capture program [12], [25] allows recording of all network traffic that flows through a link. Passive monitors can collect traces on an individual link or in a network. They are increasingly used by network administrators, not only for the need of performance monitoring but for observation of abnormal network usage. They are very universal. We can use collected traces in further analysis to examine different network analysis goals. In addition, passive measurements do not introduce overhead traffic. Unfortunately traces can be extremely large or may not include all packets needed for the analysis due to route changes or multipath forwarding. This way of performance measurements is popular and several popular sniffers are available (e.g. Ethereal, TCPdump, Snoop, EtherPeek, WinDump, Tcptrace, Analyzer). They can produce several different types of output containing information collected on each TCP connection, such as elapsed time, bytes and segments sent and received, retransmissions, round-trip times, throughput, and more. Sniffers usually may work at all network layers, including the Web layer.

"Pure" Web active measurements are focused on the continuous periodical observation of Web site performance characteristics through the benchmarking the Web and the measurement of query latency and transfer rate over a 24-hour period. URLs can be measured e.g. by loading the base page from the specific measurement sites. For example, the service provided by the MyKeynote [27] measures Web site's performance and availability from a word-wide network of measurement agents and visualizes the Web page downloading as perceived by particular agents. The results are presented on a Web page. MyKeynote can perform *ad hoc* and periodic full page measurements, as well as can store data for further off-line analysis. Unfortunately it has a significant weakness lying in the fact that it uses a specially developed Web browser. Additionally, it uses HTTP/1.0 protocol that is not still commonly used in the Internet. Leading browsers, such as MS Internet Explorer that is the most dominating browser used by more than 80% Internet users [30], employs more advanced and efficient HTTP/1.1 protocol [14]. But the problem is much deeper than the differences in supported versions of HTTP protocol. Different browsers handle page downloading in different ways, what own Web browser does as well. Therefore, there is the need for HTTP measurement tools based on most popular browsers. Nevertheless, MyKenote gives worthwhile results for understanding how the page can be loaded. The Patrick.net [28] is an example of similar but much simpler and non-commercial service for testing a Web page. Target page downloading is described as it is perceived in California, USA where this service is localized. Unfortunately it does not yet handle Javascript, Java, SSL and frames. Also periodic measurements and data base support are not available.

Another approach is presented by NAPA (Network Application Performance Analyzer) project [22]. In this open source project the user can use the NAPA analyzer, installed on user Windows workstation. NAPA can visualize Web page downloading timelines for all browsers that are installed in considered operation system. It is a simple and useful tool but unfortunately we experienced that it has some bugs and it is not stable. Moreover we cannot monitor Unix based browsers and store captured data for further off-line processing.

To handle the limitations of measurement systems mentioned above we develop a new measurement service called Wing [4]. Also we propose to use data mining techniques in the analysis of collected data.

3 Methodology and Measurement Infrastructure

Measurement of the Web is difficult however is essential if we are to gauge user perception of the Web. As it was discussed in the previous section two measurement approaches can be considered, namely active measurement based on injecting measurement data into the Web, and passive measurement based on observation of existing Web traffic. In our project we are using an active measurement approach based on our own probing, measurement and analysis infrastructure.

In order to truly measure Web traffic, which is almost entirely TCP/IP-based traffic, it is to probe using TCP/IP protocol rather than ICMP protocol. For that purpose we use the Wing system developed at our laboratory [4]. Wing is a network measurement tool that measures end-to-end network path characteristics at the HTTP layer. The entire measurement infrastructure is implemented at WUT side. The Wing works like a sonar-location system, sending GET requests for the Web object from the targeted Web site and waiting for the answer, i.e. for that object. Wing collects live HTTP trace data near a user workstation and distills key aspects of each Web transaction for all protocol (including DNS and UDP) that are used during browsing. Wing is unique because it uses a real browser running under user operating system for page downloading. Hence it perceives a Web page downloading in the same manner like a real browser. Moreover Wing exactly visualizes how a Web page is downloaded by a browser and stores all data about Web transaction in a data base for further analysis and processing. Wing can be freely programmed using scripts or may be used in ad hoc Web page diagnosis. Wing uses a SYN-ACK mechanism of TCP protocol for RTT estimation. Then it measures the time taken by the target host to respond with an ACK packet after issuing by a browser a SYN packet when starting a TCP connection. Due this estimation mechanism Wing avoids problems with ICMP-based network measurement (blocking, spoofing, rate limiting, etc.).Wing also estimates the network throughput at the Transport Layer. To estimate the average transfer rate of the TCP connection used by getting an object we measure time spacing between the first byte packet and the last byte packet of the object received by client using that connection. Transfer rate is calculated by dividing a number of bytes transferred by the amount of time taken to transfer them.

Fig. 1 shows the overall Wing architecture. Wing consists of the following distinct components: client workstation, sniffer, and data storing and processing server. The core executive software is implemented under Linux system. Because we want to observe MS IE activity the client is implemented under W2K operating system. It downloads a target Web page and issues consecutive GET requests under control of Wing executive module (partially developed under Linux and Windows). Linux-based sniffer adjacent to the Web client captures all LAN traffic related to current Web transaction. Wing controller monitors and time stamps of all browser's actions, determines the end of web page uploading, and preprocess gathered data into the format convenient for further statistical and data mining analysis, as well as for visualization. A database is used for storing collected data. In ad hoc mode Wing prepares the visualization of Web page downloading as perceived by the client. Then Wing returns a page with HTTP timeline chart and a number of detailed and aggregated data about Web page downloading and finishes the transaction. In a program mode under which our experiments were performed, Wing uses own scheduler when and where to submit Web transactions.

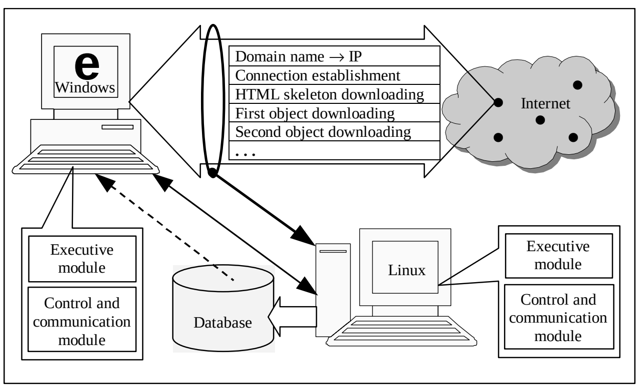

Fig. 1. The Wing architecture as implemented for MS IE

The measurements analyzed in this paper were performed between 21 September 2002 and 28 July 2003. We choose the rfc1945.txt file as a probe Web resource to be downloaded. This approach follows the idea of the ping utility idea where a standardized ICMP packet is send through the network to the target host to evaluate network performance. Similarly, at the Web layer we will submit the same HTTP request to several different Web servers to evaluate their response to the same ("standardized") request. It can be easily found on several non-commercial sites that have usually non-overloaded Web servers. Our resource is large enough (the original size is 137582 bytes) to estimate average transfer rate, and yet not too large to overload Internet link and Web server. The target servers were chosen randomly by the Google search machine. Among a few hundred links found by Google we have chosen 209 direct links to this document. After preliminary tests we have decided to use only 83 active Web servers in further measurements storing exactly the same file. Hence our experiments involved the repeated downloads of Web pages with rfc1945.txt file from 83 different

Web servers ten times a day over 24-hour period. After 47 weeks of measurements we have got the data base with information about 150.000 Web transactions.

The performance analysis presented in the section 4 is done for data from first twenty weeks, whereas all data (for all 47 weeks) is considered in reliability analysis. Longer observation period allows for better identifying of reliability quality indicators. Table 1 lists the servers used in our experiment. The geographic localization (longitude, latitude, country, and city) of target server is determined using our host localization service which was developed based on the NetGeo CAIDA's service [26]. The distance is the geographical distance between the target server's city and Wroclaw.

Table 1. Target servers

#	WEB SERVER	COUNTRY	CITY	DISTANCE [km]
2	199.125.85.46	US	Manchester	6350
4	www.fiuba6662.com.ar	NL	-	788
5	ftp.univie.ac.at	AT	Vienna	323
9	ironbark.bendigo.latrobe.edu.au	AU	Bendigo	15617
14	cs.anu.edu.au	AU	Canberra	15844
16	files.ruca.ua.ac.be	BE	Antwerp	876
19	www.deadly.ca	CA	Calgary	7754
21	www.munet.mun.ca	CA	-	6286
24	tecfa.unige.ch	CH	Geneve	962
27	www.embed.com.cn	CN	Baoan	8552
33	www.cgisecurity.com	US	Manchester	6350
37	www.gordano.com	UK	Bristol	1358
40	www.networksorcery.com	US	Herdon	7017
41	www.q-linux.com	PH	Makati	9697
43	www.sashanet.com	US	Provo	8702
44	docs.securepoint.com	US	Lanexa	7109
47	www.soldierx.com	US	Panguitch	8970
51	salsero.ibp.cz	CZ	Brno	210
52	bbs1.biz-worms.de	DE	Worms	634
54	www.deadlyzone.de	DE	Karlsruhe	654
57	www.gmd.de	DE	Sankt Augustin	688
61	www.netzmafia.de	AU	-	14074
62	www.robsite.de	DE	Karlsruhe	654
64	www.dbg.rt.bw.schule.de	DE	Stuttgart	614
69	www.teco.uni-karlsruhe.de	DE	Karlsruhe	654
71	www.vorlesungen.uni-osnabrueck.de	DE	Osnabruck	630
72	www-pu.informatik.uni-tuebingen.de	DE	Tubingen	637
74	jungle.brock.dk	NL	Amsterdam	844
77	hea-www.harvard.edu	US	Cambridge	6375
78	www.isi.edu	US	Marina Del Rey	9599
81	Web.mit.edu	US	Cambridge	6374
87	www.teco.edu	DE	Karlsruhe	654
88	www.ics.uci.edu	US	Irvine	9601

89	philby.ucsd.edu	US	La Jolla	9662
90	www.cs.uh.edu	US	Houston	8860
101	rfc.eunet.fi	FI	-	1116
103	kludge.tky.hut.fi	FI	Espoo	1116
104	www.tls.cena.fr	FR	Paris	1076
107	clauer.free.fr	FR	Paris	1076
109	www.loria.fr	FR	Nancy	817
110	abcdrfc.online.fr	FR	Paris	1076
111	Eurise.univ-st-etienne.fr	FR	-	1124
114	wigwam.sztaki.hu	HU	Budapest	427
117	www.crackinguniversity2000.it	IT	Al	1458
123	web.fis.unico.it	NL	Amsterdam	844
124	omega.di.unipi.it	IT	Pisa	958
125	www.mfn.unipmn.it	IT	Rome	1081
126	cesare.dsi.uniroma1.it	IT	Rome	1081
128	www.dsi.unive.it	IT	Venice	719
132	www.nendai.nagoya-u.ac.jp	JP	Tokyo	8848
135	gipwmc6.shinshu-u.ac.jp	JP	Nagano	8699
136	www.goice.co.jp	JP	Tokyo	8848
141	www.toyota.ne.jp	JP	Tokyo	8848
142	rfc.netvolante.jp	JP	Tokyo	8848
146	laplace.snu.ac.kr	KR	Seul	8040
154	www.freenic.net	US	New York	6677
158	www.potaroo.net	AU	Canberra	15844
160	www.cs.vu.nl	NL	Amsterdam	844
161	www.ii.uib.no	NO	Bergen	1260
163	www.alliedtelesyn.co.nz	NZ	-	17996
164	www.alternic.org	US	New York	6677
166	www.freesoft.org	US	Mountain View	9394
167	www.ietf.org	US	Natick	6395
168	ietfreport.isoc.org	AU	Canberra	15844
170	www.bw.kernel.org	ZA	Stellenbosch	9455
171	www.lousy.org	FI	Espoo	1116
173	www.os-omicron.org	JP	Tokyo	8848
174	memory.palace.org	US	Patulum	9329
177	www.theheap.org	US	Clifton Park	6501
179	free.vlsm.org	ID	Depok	9438
181	www.watersprings.org	JP	Tokyo	8848
182	yah.do.pl	PL	Wroclaw	0
184	katmel.eti.pg.gda.pl	PL	Lublin	385
187	www.ave.dee.isep.ipp.pt	PT	Porto	2242
188	www.fe.up.pt	PT	Porto	2242
191	ftp.korus.ru	RU	Yekaterinburg	2868
192	www.math.chalmers.se	SE	Goteborg	757
193	kst.fri.utc.sk	SK	Zilina	240
199	www.csie.nctu.edu.tw	TW	-	8911
200	dbWeb.csie.ncu.edu.tw	TW	-	8911
202	www-uxsup.csx.cam.ac.uk	GB	Cambridge	1168
205	www.ntmail.co.uk	UK	Bristol	1358
209	www.ietf.cnri.reston.va.us	US	Natick	6395

4 Summary of the Experimental Results

Due to the space limitations we present in this paper only the summary of the experimental results. First when doing active Internet measurements we need to evaluate the probing process itself. A sample (for the server #2) distribution of measurement time intervals is shown in Figs. 2 and 3. Fig. 2 depicts detailed information about system stoppage and shows time intervals between consecutive probing and measurements. Measurement time intervals greater than 5h20min were cut off. We have found that the intervals less than 2h40min were caused by the specific implementation of Wing's transaction scheduler. Time intervals greater than 2h40min were caused by system failures (including power off), lack of Internet access, lack of server reply due network congestion or server overloading. Fig. 3 presents a sample (for the same server #2) histogram of the measurement time intervals. Most of measurement time intervals (81%) are consistent with the assumptions (2h40min).

The characteristics showing the distribution of the HTTP average transfer rate for the Web server #21 is presented in Fig. 4. In the analysis we also investigate the correlation between a connection's RTT and transfer rate to examine whether connections with shorter RTT tend to transfer more data at the HTTP layer. The question is whether we can use RTT measurements for deriving HTTP throughput. Fig. 5 presents an answer by showing the distribution of median values of the average transfer rate (throughput) vs. RTT. Based on the measurements that we have analyzed so far, it is inconclusive to say that the RTT can be a good predictor of HTTP throughput in general. The distribution of HTTP throughput versus RTT can be described using power law of the form $y=kx^{\alpha}$ with k and α determined experimentally: k=46456 and α=-0.8805.

Wing precisely monitors each transaction and checks whether all embedded objects are downloaded by the browser. Wing can classify events related to transactions as: "OK", "PAGE_DEATH", "BROWSER_FAILURE" or "SERVER/INTERNET FAILURE". Using such event definitions we may compute the transaction reliability percentage rate as the percentage ratio of the sum of the number of events classified as "PAGE_DEATH" and "BROWSER_FAILURE" to total number of transactions. Fig. 6 shows the distribution of transaction reliability rate as well as the number of transaction issued for all target servers. The most reliable server has 98.2% transaction reliability ratio whereas the most unreliable server has 23.7% transaction reliability ratio only.

Another result is shown in Fig. 7 where the percentage availability of URLs is plotted versus day of observation. We can determine the "mortality" rate of observed URL links as -0.006, i.e. only about 80% of mirrored Web sites defined in the beginning were still available in the last phase of experiment.

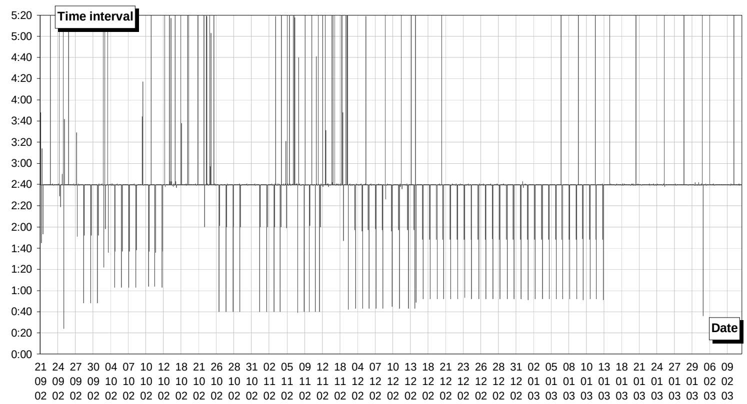

Fig. 2. Sample measurement time interval distribution (for the server #2)

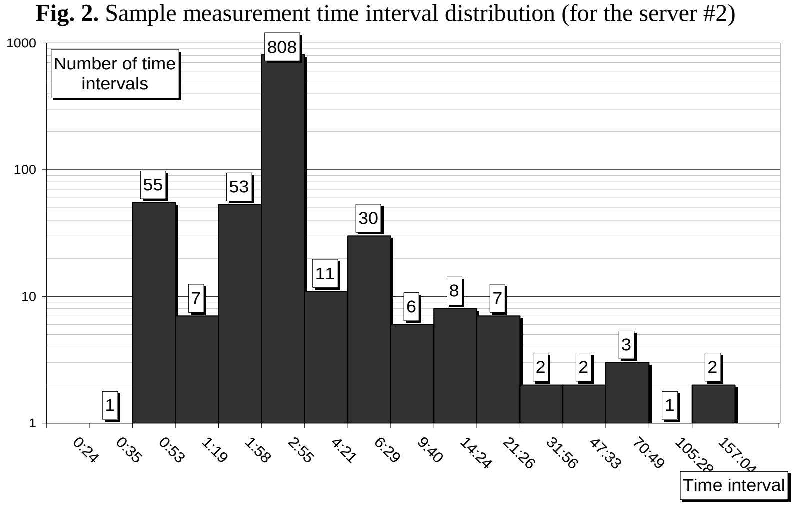

Fig. 3. Sample histogram of measurement time intervals (for the server #2)

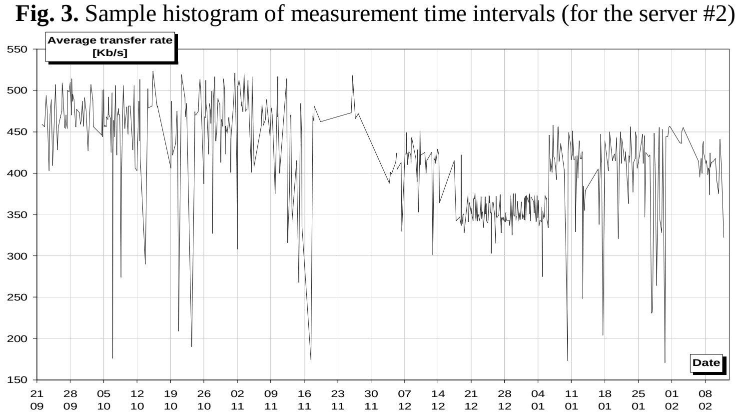

Fig. 4. HTTP average transfer rate distribution for the Web server #21

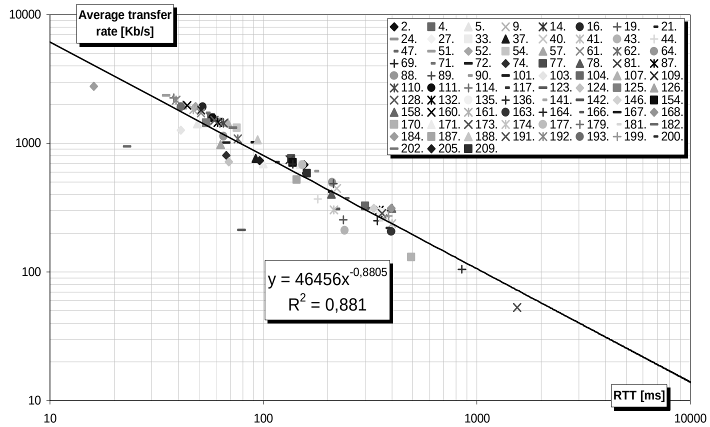

Fig. 5. Distribution of median values of the average transfer rate vs. RTT

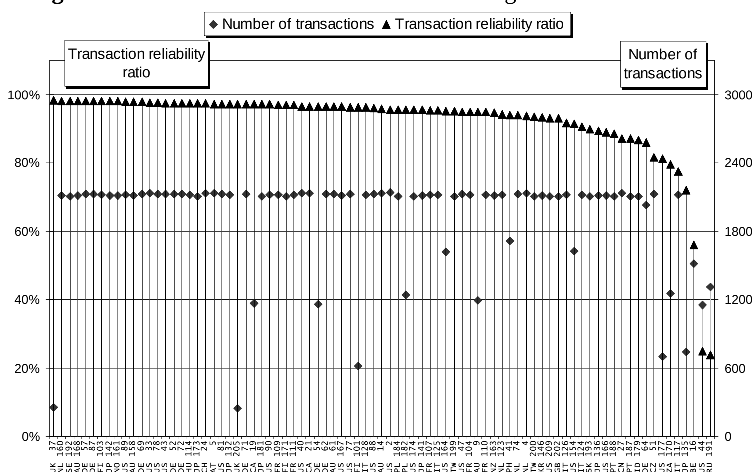

Fig. 6. Number of transactions and transaction reliability ratio

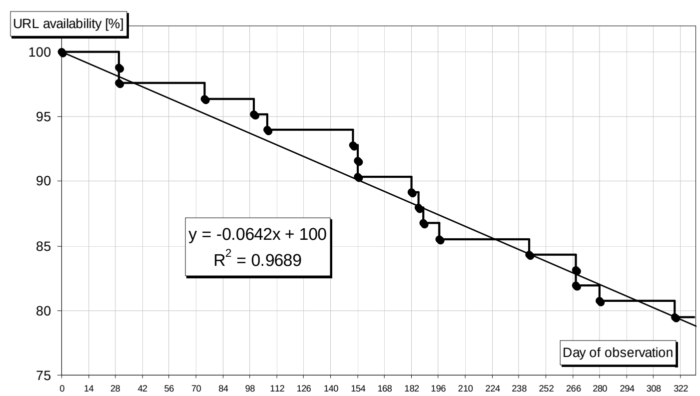

Fig. 7. URL availability

5 Conclusions

A number of experiments were performed to evaluate Web quality in the context of the performance and reliability of Internet access from the Wroclaw University of Technology campus network, Wroclaw, Poland. The experiments were conducted using Wing measurement infrastructure. There are several classes of performance and reliability problems that can be analyzed using Wing. We present main results, especially those which may characterize some aggregate view of Web performance and reliability as perceived from WUT local area network. Most valuable results that apply for our WUT location are presented in Fig. 5, 6 and 7.

We would like to use the result presented in Fig. 5 as the model describing the WUT Internet characteristic, especially for the estimation of the average Web throughput (in the sense of median value) on the basis of the TCP RTT measurement. An obvious question is how well this approach might generalize to other Web environments. We determined the throughput v. RTT model for WUT Wroclaw. But the experiments were conducted only at our local site. This experiment should be repeated in other locations to determine the parameters of the model specific to the particular site.

Obviously, Web users want to have knowledge about Web performance and reliability. Generally, it is hard to get this knowledge before clicking a particular Web site link and getting data. In many situations it is challenging to know *a priori* which of many Web servers from some set of servers under observation will have best Web quality. For example, when we are optimizing Web page downloading it's crucial to predict the performance of all mirrors under consideration. Another future work would be to related to computational grids. Such systems aggregate a wide variety of Internet resources including supercomputers, storage systems and data sources distributed all over the world, and use them as a single unified resource forming what is popularly known as Grids [10]. Currently, grids are also built using Web technology. Well-predicted Web performance and reliability is a key issue in such projects to decide which remote resources are to be used in the demanded time period.

References

1. Andersen D., Balakrishnan H., Kaashoek F., Morris R.: Resilient Overlay Networks. In: Proc. of 18th ACM Symp. on Operating Systems Principles, Banff, Canada (2001) 131-145
2. Ballintijn G., Van Steen M., Tanenbaumn A. S.: Characterizing Internet Performance to Support Wide-Area Application Development. Operating Systems Review, 34 (4) (2000) 41-47
3. Borzemski L.: Data Mining in Evaluation of Internet Path Performance. In: Innovations in Applied Artificial Intelligence: 17th International Conference on Industrial & Engineering Applications of Artificial Intelligence & Expert Systems, IEA/AIE 2004, Ottawa, Canada, May 17-20, 2004, Proceedings. Lecture Notes in Artificial Intelligence, Vol. 3029, Springer-Verlag Berlin Heidelberg (2004) 643-652
4. Borzemski L., Nowak Z.: WING: A Web Probing, Visualization and Performance Analysis Service. In: Web Engineering: 4th International Conference, ICWE 2004, Munich Germany, July 26-30, 2004, Proceedings. Lecture Notes in Computer Science, Vol. 3140, Springer-Verlag Berlin Heidelberg (2004) 601-602

5. Brownlee N., Claffy K, Murray M., Nemeth E.: Methodology for Passive Analysis of a University Internet Link. In: Proc. of Workshop on Passive and Active Measurements PAM2001, Amsterdam, Holland (2001)
6. Brownlee N., Loosley C.: Fundamentals of Internet Measurement: A Tutorial. Keynote Systems, USA (2001)
7. Cardellini V., Casalicchio E., Colajanni M., Yu P.S.: The State of the Art in Locally Distributed Web-Server Systems. ACM Computing Surveys, Vol. 34, No. 2, June (2002) 263-311
8. claffy K., McCreary S.: Internet Measurement and Data Analysis: Passive and Active Measurement. University of California, CAIDA, USA (1999)
9. Faloutsos M., Faloutsos Ch.: Data-Mining the Internet: What We Know, What We Don't, and How We Can Learn More. Full day Tutorial ACM SIGCOMM 2002 Conference, Pittsburgh, PA (2002)
10. Foster I., Kesselman C. (Eds.): The Grid 2: Blueprint for a New Computing Infrastructure, Second Edition, Morgan Kaufmann, Elsevier, San Francisco (2004)
11. Garofalakis M., Rastogi R.: Data Mining Meets Network Management: The NEMESIS Project., Proc. of DMKD'2001, Santa Barbara, California (2001)
12. Jacobson V., TCPdump, the protocol packet capture and dumper program.
13. Luckie M. J., McGregor A. J., and Braun H.-W.,: Towards Improving Packet Probing Techniques, In: ACM SIGCOMM Internet Measurement Workshop, San Francisco, CA (2001)
14. Mogul J., Clarifying the Fundamentals of HTTP, In: Proc. of WWW11 Conference, Honolulu (2002) 444-457
15. Murray, M. claffy, k.: Measuring the Immeasurable: Global Internet Measurement Infrastructure. In: Proc. of Workshop on Passive and Active Measurements PAM2001, Amsterdam, Holland (2001) 159-167
16. Ng E. T. S., Zhang H.: Towards Global Network Positioning. In: ACM SIGCOMM Internet Measurement Workshop, San Francisco, CA (2001) 25-29
17. Padmanabhan V. N., Qiu L.: Network Tomography Using Passive End-to-End Measurements. In: DIMACS Workshop on Internet and WWW Measurement, Mapping and Modeling, Piscataway, NJ (2002)
18. Padmanabhan V. N., Qiu L., Wang H.: Server-Based Inference of Internet Performance. In: Proc of IEEE Infocom 2003, Vol. 1, San Francisco, CA (2003) 145- 155
19. Palmer Ch. R., Siganos G., Faloutsos M., Faloutsos Ch., Gibbons P.: The Connectivity and Fault-Tolerance of the Internet Topology. In: Proc. of Workshop on Network-Related Data Management (NRDM 2001), Santa Barbara, CA (2001)
20. Saroiu S., Gummadi K. P., Gribble S. D. King: Estimating Latency between Arbitrary Internet End Hosts, In: SIGCOMM Internet Measurement Workshop, Marseille, France (2002) 3-14
21. Wolski R. Dynamically Forecasting Network Performance Using the Network Weather Service. Technical Report TR-CS96-494, U.C. San Diego, CA (1996)
22. Yoder J.,: Better End User Visible Web Browning Performance, Intel Corp. (2002)
23. Zhang Y., Duffield N., Paxson V., Shenker S. On the Constancy of Internet Path Properties. In: ACM SIGCOMM Internet Measurement Workshop, San Francisco, CA (2001) 197-211
24. http://ipmon.sprint.com/
25. http://irg.cs.ohiou.edu/software/tcptrace/index.html
26. http://www.caida.org
27. http://www.mykeynote.com
28. http://www.patrick.net
29. http://www.slac.stanford.edu
30. http://www.w3schools.com