

# Source Traffic Characterization for Thin Client Based Office Applications

Barbara Emmert\*, Andreas Binzenhöfer, Daniel Schlosser, and Markus Weiß

University of Würzburg, Institute of Computer Science, Würzburg Germany

**Abstract.** A thin client is a small network computer which is used as a remote screen visualizing the output of software applications running on a central server. To provide a seamless service to thin client users the network connection between the client and the server must be dimensioned properly. In this paper we therefore characterize the traffic generated by different types of thin client users when working with popular office applications like Microsoft Word, Excel, or PowerPoint. We analyze the traffic patterns measured in our testbed environment and thus provide a basis for subsequent research studies as well as for administrators to estimate the amount and the characteristics of user generated traffic.

## 1 Introduction

There are different ways for a company to provide their employees with the software they need to fulfill their jobs. The most common solutions are to install the applications locally on each client or to provide the software over network file systems. However, both approaches require that every desktop system has enough resources to run the application. An alternative is to run all applications on a central server and merely show the results on thin clients. While this solution is slightly less comfortable for the end-user, it reduces the initial expenditures as well as the current energy and maintenance costs. As a side-effect it also becomes easier for an administrator to manage the software suite as well as to install updates and patches. For the same reasons, the concept of application service providing (ASP) has become popular, especially among small and mid sized organizations. One of the services an ASP offers to its customers is to host desktop-type applications on a server farm and to provide access, administration, and support.

Originally, the thin client architecture was designed for local area networks (LAN). However, an increasing number of clients is connected via wide area networks (WAN) or over leased lines. The problem of this development is that the network connection (possibly shared by all clients) can easily become a bottleneck and affect the service quality experienced by the end-user. In order to properly dimension the company network and the lines rented for home office workers, the traffic generated by a thin client user must be well understood.

---

\* Corresponding author: [emmert@informatik.uni-wuerzburg.de](mailto:emmert@informatik.uni-wuerzburg.de)

In this paper we therefore characterize the traffic flow between a typical thin client and the widely used Citrix Presentation Server 4.0 [1]. This traffic depends not only on the implementation of the underlying ICA protocol, but also on the application in use and the behavior of the end-user. On that account we measure the traffic generated by different types of users working with Word, Excel, and PowerPoint, the most popular applications from the Microsoft Office suite. We will compare different scenarios and applications in terms of important characteristics like used packet sizes, bandwidth consumption, and time between packets. Thereby we differentiate between the traffic produced by the server and the traffic produced by the thin client. The results from our measurements are useful for companies to properly dimension their networks according to the expected number of clients as well as for researchers conducting analytical or simulative studies in a thin client environment.

The remainder of this paper is organized as follows. Section 2 gives a brief overview of related work. The measurement setup as well as the measurement methodology are explained in Section 3. We discuss the different scenarios, the results, and their implications in Section 4, Section 5 finally concludes this work.

## 2 Related Work

As simulation is an important tool for networking research, it is vital to generate realistic traffic streams. Due to the overwhelming dominance of web traffic in contemporary WANs, most source traffic models focus on this fraction of the utilized bandwidth. So does the stochastic source traffic model for HTTP traffic proposed by Cao et al. [2]. The authors analyzed packet traces from two links connecting medium sized organizations to the Internet. A more general approach has been proposed by Staehle et al. [3], who established a source traffic model for realistic wireless simulations. For this purpose, they introduced a single user traffic model which considers Email, HTTP, FTP, and WAP traffic.

The number of ASPs hosting office applications for remote users is growing, but the percentage of bandwidth consumed by this type of service remains negligible. Therefore, little work has been dedicated to analyzing the characteristics of traffic caused by thin client based applications. One of the few studies in this area has been performed by the Tolly Group [4] who evaluated the usability of Microsoft PowerPoint via WAN. They investigated the consumed bandwidth and completion time of a common PowerPoint operation executed on a machine running Citrix MetaFrame XP client software accessing a server hosting the corresponding Presentation Server. They report “on machine” experience for high bandwidth links like Fast Ethernet, Ethernet, or WLAN and up to twice as long completion times for slower links. Tolia et al. [5] evaluated VNC performance, focusing on the operation response times for image processing, presentation creating, and text processing. Their findings show that highly interactive applications are more sensitive to a higher network round trip time than simpler applications. More general measurements are reported by Lai and Nieh [6], who evaluated the performance of thin client computing in a WAN.

For their work, they used both measurements in the Internet2 and simulations in a testbed. One of the investigated characteristics was the completion time of typical office tasks like typing, scrolling, or image downloading. To analyze the difference between the considered platforms, they compared the amount of data transferred before the completion of each test, the required bandwidth, and the test duration. The fact that the typing task was completed faster, and required less bandwidth than scrolling already gave a first insight into the structure of traffic caused by office applications. In this work, however, we intend to specify the characteristics of the emerging traffic in more detail. In particular, we distinguish between the client and the server side, different user behavior as well as different office applications.

### 3 Measurement Setup and Methodology

In order to measure and characterize the traffic of thin client applications, we set up a typical thin client environment as depicted in Figure 1. It consists of a

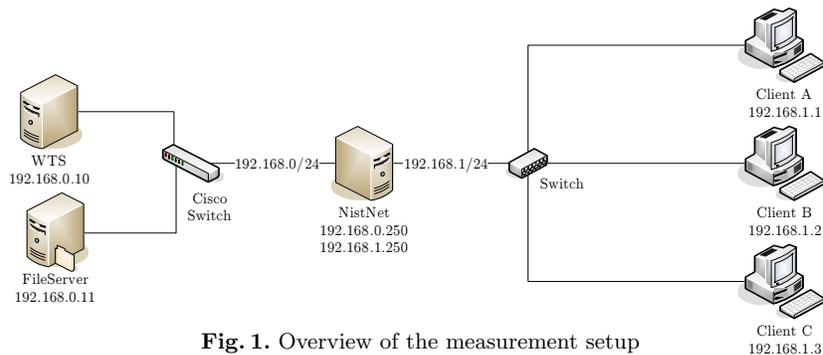


Fig. 1. Overview of the measurement setup

server farm which is accessed by three clients and a network emulator running NistNet [7], which enables us to alter delay, jitter, and packet loss of the end-to-end connection. The Windows Terminal Server as well as the corresponding file server are Intel Xenon 3.4Ghz machines with 3.5 GB RAM running Windows 2003 Server in the standard edition with service pack 1. The clients are Pentium III 2.6Ghz machines with 1 GB RAM running on Windows XP Professional with service pack 2. The Windows Terminal server hosts the entire Microsoft Office 2003 product family and runs Citrix Presentation Server 4.0 to make these applications available for remote users. The clients use version 9.237 of the ICA client to access the applications hosted on the server. This setup enables us to generate and measure thin client caused traffic in a controlled environment.

Our emulation experiments are set up as follows. At first the client opens a Citrix session and starts the corresponding application. It then performs different tasks which are typical for office users. To automate this procedure, we used AutoHotkey [8], which is able to carry out keystrokes as well as mouse movements and clicks according to a simple macro language. This way, we are, e.g., able

to emulate different typing speeds in Word or the insertion of pictures into a PowerPoint slide. To capture the corresponding processes on network level, we recorded packet traces using Windump [9] on both the client and the server machines, which could then be analyzed offline.

## 4 Measurements and Results

The traffic generated by a thin-client on packet level depends on both the application and the behavior of the user. Intuitively, the activity of the user directly corresponds to the amount of consumed network bandwidth. To verify this assumption, we will therefore regard different office applications and characterize the source traffic created by different types of users of which we distinguish light, normal, and power users, in the following.

### 4.1 Analysis of Typical Word Tasks

To obtain an initial qualitative understanding of thin client traffic, we emulated typical tasks performed by a Microsoft Word user like typing, scrolling, or selecting menu entries. During the typing test, the user continuously types text and thereby corrects some misspelled words. He then scrolls the document using the scroll bar and selects some entries from the menu bar. Figure 2(a) confirms that the different tasks require different amounts of bandwidth on network level. Thereby scrolling consumes the most bandwidth as large parts of the screen need to be refreshed.

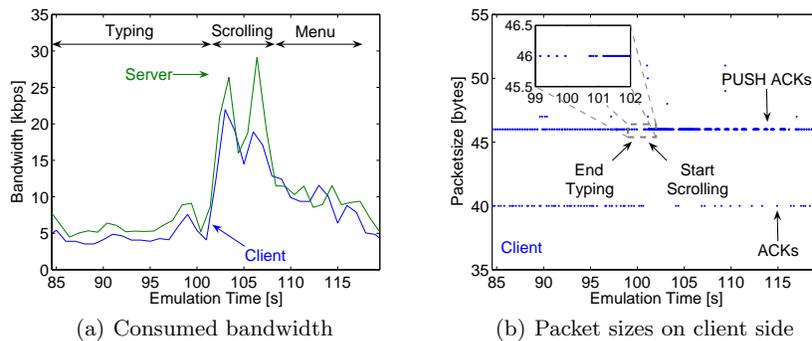


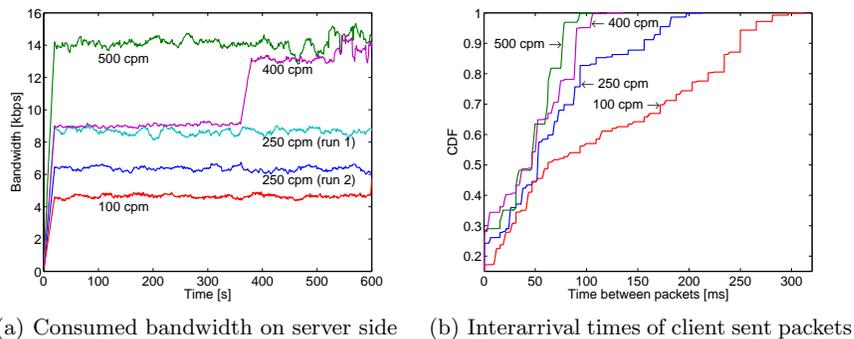
Fig. 2. Characteristics of typical Word tasks

More interesting observations, however, can be made when looking at the sizes of the packets which were sent by the client during the emulation as shown in Figure 2(b). Almost all packets sent by the client are either 40 byte TCP acknowledgments or 46 byte TCP push acknowledgments. That is, the Citrix client encodes all user input using 6 byte payload and setting the TCP push flag to enforce the immediate delivery of the data. Thereby, instead of sending packets of larger size, the ICA protocol prefers to reduce the time between packets in times of higher bandwidth utilization. This can nicely be seen in the zoomed area in Figure 2(b), which shows that the time between two sent packets is

significantly smaller when scrolling than while typing. Thus, in the design of the ICA protocol, responsiveness seems to have played a more important role than overhead.

#### 4.2 Impact of the User Behavior

Next, we analyzed the traffic caused by different types of typing users. The results shown in Figure 3 were obtained by emulating a user who types the beginning of Orwell’s ”War of the Worlds” for 10 minutes. We considered typing speeds of 100, 250, and 400 characters per minute (cpm) as to be typical for light, normal, and power users respectively. For comparison purposes, we also include results obtained for 500 cpm.



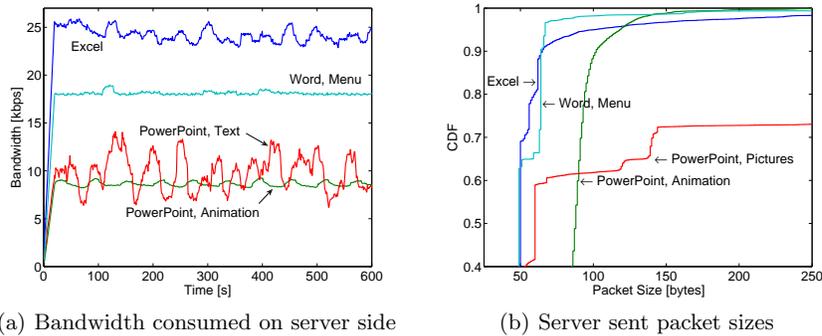
**Fig. 3.** Traffic characteristics varying with typing speed

When different types of users were emulated, the bandwidth consumed by the client varies between 2 and 7 kbps. In Figure 3(a) we show 20 s moving averages of the bandwidth consumed by the server for the same experiment. The abrupt rise in the curve representing the power user (400 cpm) in Figure 3(a) is illustrating our observations during the measurements: In some cases, the consumed bandwidth suddenly grew by a significant amount, but remained constant before and after this event. As this increase is too fast for being TCP triggered, we assume that an ICA-internal QoS mechanism is responsible for it. This appears also to be the reason, why the obtained mean values for a normal user (250 cpm) differ by more than 2 kbps for two different experiments (run 1 and run 2). However, the total of our measurements showed, that besides the mentioned increases, which did not occur on client side, a typing user consumes a nearly constant amount between 4 and 15 kbps of network bandwidth depending on the typing speed.

The comparison of interarrival times of client sent packets in Figure 3(b) shows results similar to the observations described in the last section: The cumulative density functions (CDFs) illustrate that the client increases the bandwidth by sending out packets faster in the case of a more active user. The server sent larger packets, while the size of most client packets was 46 byte. Thus the Citrix client adapts to the user behavior by varying the time between two packets instead of the packet size.

### 4.3 Comparison of Different Office Applications

Figure 4 visualizes the impact of the application type on the network traffic. For this experiment we compared the emulations of users searching and selecting menu entries under Word, inserting text, pictures as well as looking at animations under PowerPoint and selecting fields in an Excel data sheet.



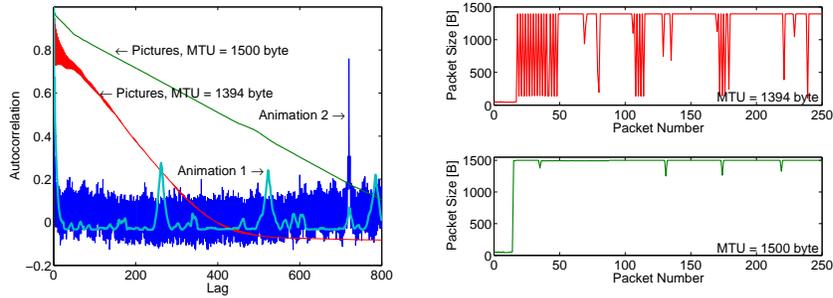
**Fig. 4.** Traffic characteristics varying with applications

First, we compared the consumed server side bandwidth. Again, the bandwidth consumed by the client was smaller and was only 2 kbps in the case where the animation was displayed. Note that in Figure 4(a) we do not show the bandwidth consumed for the insertion of pictures in a PowerPoint presentation, as the used files were roughly of size 2 MB. The amount of consumed bandwidth was thus significantly larger than for all other tasks. We were however surprised, that the compared 20s moving averages for the Excel test and for Word menu operations were significantly higher than for displaying animations, which lay roughly in the range of inserting text in a presentation. The variations in the bandwidth of the latter test may arise from the necessary creation of new slides, a task which requires bandwidth intense mouse movements. All in all, we saw again that while the consumed bandwidth differs quite strongly between the applications, it is nearly constant for each single task.

As in the earlier described cases, the size of packets sent by the client is almost constant. The size of the server sent packets, however, differs strongly between the measurements (see Figure 4(b)). Note that for the case of highly interactive mouse operations (Excel and Word menu), the size of more than 50 % of all packets is only 40 byte, i.e. those packets are acknowledgments. Our measurements showed that, similar to the typing tests analyzed earlier, the responsiveness seems to be improved by sending packets faster. This does however not hold for the case of animations and pictures under PowerPoint, where the bandwidth increase is done by sending larger packets. Another interesting fact is illustrated by the CDF in Figure 4(b) representing the insertion of pictures in a presentation: While all inserted pictures were stored on the client, the large packet sizes observed during the experiment indicate, that for presentation purposes the picture files still have to be transferred to the server.

#### 4.4 Characterization of Server Side Traffic

Our last experiment was dedicated to a deeper investigation of the server sent packet sizes. For this purpose, we analyzed traces collected during the insertion of pictures and the visualization of different animations under PowerPoint. A shorter and a longer animation were compared and the influence of the maximum transfer unit set at the server were investigated.



(a) Packet size autocorrelation (b) Snapshots of a picture transmissions

**Fig. 5.** Server sent packet size characteristics for different PowerPoint tasks

Figure 5(a) visualizes the autocorrelation of server sent packet sizes for a lag up to 800. Apart from two spikes in the case of Animation 1 and one spike in the case of Animation 2, the autocorrelation is not significant, but fluctuating. This indicates, that the traffic caused by animations is not bursty, but that quite often an acknowledgment packet seems to be transferred just after a larger data packet. The spikes correspond to the duration of the animation which was shorter in the first case.

The comparison of the two experiments of picture insertion reveals that, while the autocorrelation is fluctuating quite strongly in the case where the server MTU was set to 1394 byte, this is not the case, if the MTU is set to 1500 byte. Both autocorrelations are not significant any more after a given time which corresponds to the transmission of a picture. The reasons for the different shapes and slopes of the curves become apparent in Figure 5(b), where we show for each measurement a snapshot of 250 subsequent sent packets. In both cases, we chose the begin of a picture transmission. Observe that for the case, where the smaller maximum packet size is chosen, very often a small packet of around 200 byte is following a large packet of 1394 byte. This is however, not the case if the MTU is set to 1500 byte: Under this configuration, the most common packet size is corresponding to the maximal feasible value. Furthermore, smaller packet sizes are only used for acknowledgments which gives the autocorrelation a higher value than in the first case. These results indicate that the Presentation Server tries to send packets of 1500 byte in both cases, which are defragmented by the operation system under the first configuration. Thus, if large pictures or files have to be transmitted, the parameter settings of the operating system and the ICA protocol have to be chosen and adapted with care to avoid unnecessary transmission delays and bandwidth consumption.

## 5 Conclusion

Despite their growing popularity, little is known about the characteristics of traffic generated by thin client based applications. In this paper, we therefore analyzed different office applications, published by Citrix Presentation Server 4.0 and accessed by thin clients, by measurements on the network layer. During our experiments, we found that the user input, like keystrokes or mouse movements, is mainly encoded with a payload of 6 byte. Thereby higher bandwidth rates are mainly achieved by decreasing the time between two packets while keeping the size of the packets constant. This indicates that in the design of the underlying ICA protocol, responsiveness played a more important role than efficient bandwidth usage. We found moreover, that keyboard input causes less traffic than mouse movements or the transfer of large files, but that PowerPoint animations consume comparatively small amounts of bandwidth. Our measurements showed furthermore, that except for the transmissions of large files, the autocorrelation of the packet sizes is rather small, which indicates that the considered thin client traffic is not bursty.

The results presented in this work can be used by system administrators to understand ICA traffic and help in network bandwidth dimensioning when updating existing or designing new thin client architectures. However, special care should be taken when setting the size of the maximum transfer unit at the server: values smaller than 1500 byte (likely to be used in VPNs) may possibly increase bandwidth consumptions and transmission delays of large files, thus decreasing the user perceived quality of experience.

## Acknowledgments

The authors would like to thank Björn Boder for the insightful discussions as well as for providing the hardware necessary to conduct the studies.

## References

1. Citrix Systems, Inc. <http://www.citrix.com>
2. Cao, J., Cleveland, W., Gao, Y., Jeffay, K., Smith, F., Weigle, M.: Stochastic Models for Generating Synthetic HTTP Source Traffic. In: IEEE INFOCOM 2004, Hong Kong, China (March 2004)
3. Staehle, D., Leibnitz, K., Tran-Gia, P.: Source Traffic Modeling of Wireless Applications. Technical Report 261, University of Würzburg, Institute of Computer Science (June 2000)
4. The Tolly Group: Citrix Systems, Inc. MetaFrame XP Presentation Server Windows-based Application Access Performance and Functionality (2003)
5. Tolia, N., Andersen, D.G., Satyanarayanan, M.: Quantifying Interactive User Experience on Thin Clients. *IEEE Computer Society Press Computer* **39**(3) (2006)
6. Lai, A.M., Nieh, J.: On the Performance of Wide-Area Thin-Client Computing. *ACM Transactions on Computer Systems* **24**(2) (2006)
7. Carson, M., Santay, D.: NistNet: a Linux-based network emulation tool. *SIGCOMM Computer Communication Review* **33**(3) (2003)
8. AutoHotkey. <http://www.autohotkey.com/>
9. WinPcap. <http://www.winpcap.org/>