



RealTracer—Tools for Measuring the Performance of RealVideo on the Internet

YUBING WANG
EMC Corporation Hopkinton, MA 01748, USA

wang_yubing@emc.com

MARK CLAYPOOL
Computer Science Department, Worcester Polytechnic Institute, 100 Institute Road, Worcester, MA 01609, USA

claypool@cs.wpi.edu

Abstract. The increase in high-bandwidth connections and high-speed computers has spurred the growth of streaming media across the Internet. While there have been a number of studies measuring the performance of traditional Internet traffic, there have not been sufficient empirical measurements of video performance especially for commercial videos across the Internet. The lack of empirical work that measures streaming video traffic may arise from the lack of effective video performance measurement tools. In this paper, we present *RealTracer*, a set of tools for measuring the performance of RealNetworks Video. RealTracer includes *RealTracker*, a customized video player that plays RealNetworks Video from pre-selected playlist and records user-centric video performance information. RealTracer also includes *RealData*, a data analysis tool that helps manage, parse and analyze data captured by RealTracker. We describe the software architecture and usage of RealTracker and the usage of RealData, both publicly available for download. To illustrate the use of RealTracer, we present some results from a study that used RealTracker to measure RealVideo performance across the Internet. Using RealData, that study made several contributions to better understanding the performance of streaming video on the Internet. Typical RealVideos have high quality, achieving an average frame rate of 10 frames per second and very smooth playout, but very few videos achieve full-motion frame rates. Overall video performance is most influenced by the bandwidth of the end-user connection to the Internet, but high-bandwidth Internet connections are pushing the video performance bottleneck closer to the server.

Keywords: streaming video, RealPlayer, performance measurement

1. INTRODUCTION

The increase in high-bandwidth connections and high-speed computers has spurred the growth of streaming media across the Internet. Web sites, traditionally text and graphics only, are increasingly offering streaming videos such as news clips, concerts, taped presentations and sporting events. Applications to access popular streaming media, such as RealNetworks RealOne, Microsoft Windows Media Player or Apple Quicktime, are freely available for most operating systems platforms, providing the potential for nearly every Internet end-host to play streaming video.

The impact of streaming video on the Internet will be largely determined by the use of commercial streaming media products, which has increased dramatically [8]. RealNetworks RealPlayer is installed on over 90% of home PCs, Apple Quicktime claims more than 100 million copies distribute world-wide, and Microsoft Media Player currently provides 220 million players [2]. RealPlayer is the most popular streaming media player on the US

Internet, with over 45% of the commercial market share in 2001 and an increase of over 50% from the previous year [8].

Over the years, there have been a number of studies measuring the performance of Internet backbones and end-hosts [14, 21], as well as detailed studies on the performance of Web clients [9, 12]. However, there have not been sufficient empirical measurements of video performance across the Internet. While the existing studies have been valuable in helping understand Internet performance, they are not sufficient for characterizing streaming video performance since video has application requirements that are different than the majority of Internet traffic.

Unlike typical Internet traffic, streaming video is sensitive to delay and jitter, but can tolerate some data loss. In addition, streaming video typically prefers a steady data rate rather than the bursty data rate often associated with window-based network protocols. Recent research has proposed rate-based TCP-Friendly transport protocols in the hope that streaming media applications will use them [7, 17], but such protocols are not yet widely part of any operating system distribution. For these reasons, streaming video applications often use UDP as a transport protocol rather than TCP. Thus, previous Internet-wide studies that have captured primarily TCP data do not necessarily reflect the traffic characteristics of streaming media. A study of RTSP streaming-media sessions [5] from clients from a large university to servers in the Internet compared streaming-media workloads to traditional Web-object workloads, and explored the effectiveness of caching on streaming-media workloads. While their analysis provided additional information on session length and bandwidth usage, they did not provide application level video performance metrics such as frame rate or image size. Nor did they collect performance data from geographically diverse users. Results of a brief study examining the traffic emanating from one popular Internet audio service using RealAudio [13] were useful in identifying data protocols used and flow lengths, but they did not provide information on RealVideo, which potentially uses a much higher fraction of bandwidth than do RealAudio streams. Nor did they extend their study to wide-area servers. A study on low-bit-rate streamed video over dialup connections across all 50 states in the U.S. [11] analyzed network level statistics such as packet loss and round-trip time. While their work studied the impact of these parameters on streaming media traffic, they used a proprietary streaming media protocol that may not be representative of commercial video products.

Internet traffic is commonly measured using the `tcpdump`¹ utility that can be used to filter packets from a particular application based on protocol and port number. However, the port number for many streaming media servers is not fixed, but is instead negotiated during the initial connection using protocols such as RTSP [18]. Extensions to `tcpdump` [5, 12] can parse `tcpdump` data to determine the ephemeral port for a particular session. However, none of these tools have application level information, such as frame rate, data encoding rate, etc. Instead, application level performance must be inferred from network trace information, as in [13], or by inferring client performance from aggregate server side logs, as in [23], adding inaccuracy to application level measures of performance.

The lack of empirical work that measures streaming video traffic may largely stem from the lack of effective video performance measurement tools. Capturing streaming media traffic in general is challenging because streaming video applications use a variety

of protocols. Moreover, while there have been efforts to develop common standardized protocols, many commercial applications continue to use proprietary protocols. Although existing tools [22] can parse typical multimedia control protocols to provide both packet level data and embedded control information, they do not, in general, provide application level information such as frame rate, frame jitter, or video encoding rate. Nor do such tools facilitate distribution to other users for large, wide-area data collection.

Commercial media players, including RealNetworks RealOne and Microsoft Windows Media Player, provide a means of observing performance statistics, such as average bandwidth and encoding rate. However, the same commercial players include no mechanisms for recording data or controlling playout in an automated fashion. In addition, there is often additional performance statistics that can help analyze streaming video performance that are unavailable via the player itself, but can be accessed via an API to the player core or via additional system level measurements.

In this work, we present *RealTracer*, a publicly available² set of tools for measuring the video and network characteristics. RealTracer includes *RealTracker*, a customized video player that plays RealVideo from a pre-selected playlist. For each video played, RealTracker records user-centric video performance information, including frame rate, jitter and user ratings, and can either send the performance information by email or FTP to a server or save it locally to disk. RealTracer also includes *RealData*, a data analysis tool that helps manage, parse and analyze data captured by RealTracker.

Researchers and practitioners alike can use RealTracer in at least two ways to evaluate RealVideo performance: (1) users can run RealTracker, saving the data locally, and then use RealData to analyze RealVideo performance from their own PCs; or (2) users can generate custom playlists and configuration files that have data sent to a central server, and distribute RealTracker to a group of distributed users, allowing analyzing RealVideo across a geographically diverse set of PCs.

In this paper, we describe the software architecture and usage of RealTracker and the usage of RealData. To illustrate the use of RealTracker, we present some results from a study that used RealTracker to measure RealVideo performance across the Internet from geographically diverse clients to geographically diverse servers. Using RealData, that study made several contributions to better the understanding of the performance and impact of streaming video on the Internet.

The rest of this paper is arranged as follows: Section 2 provides background needed to help understand the RealTracer tools and performance results presented here; Section 3 describes the RealTracer tools in detail; Section 4 presents some results we obtained in a wide-area study using the RealTracer tools; Section 5 summarizes our conclusions; and Section 6 presents possible future work.

2. Realvideo background

2.1. Connections and protocols

RealServer primarily uses Real Time Streaming Protocol (RTSP) [18] for session control and supports the Real-Time Protocol (RTP) [19] for framing and transporting data packets.

Earlier versions of RealServer used the Progressive Networks Audio (PNA) protocol and, for backward compatibility, newer real servers and players still support this protocol.

RealServer uses two network connections to communicate with each RealPlayer client for each stream: one for communicating control information with the client, and one for communicating the actual data. RealServer uses the control connection to request client configuration parameters and to send information such as clip titles, and clients use the control connection to send instructions such as fast-forward, pause, and stop. The video clips themselves, on the other hand, are actually streamed over the data connection.

At the transport layer, RealServer can use both TCP and UDP. The initial connection uses RTSP, with control information then being sent along a two-way TCP connection. The video data itself is sent using either TCP or UDP. The actual choice of transport protocol used is automatically determined by the RealPlayer and RealServer. This auto-configuration of protocols can be overridden by the user, but is the default and recommended setting for RealPlayer [16].

2.2. Buffering

For each video clip, RealPlayer keeps a jitter buffer to smooth out the video stream in case of changes in bandwidth, lost packets or variance in packet arrival rates (jitter). Data enters the buffer as it streams to RealPlayer, and leaves the buffer as RealPlayer plays the video clip. If network congestion reduces bandwidth for a few seconds, for example, RealPlayer can keep the clip playing with the buffered data. If the buffer empties completely, RealPlayer halts the clip playback for up to 5–30 seconds [1] while the buffer is filled again.

Typical inter-frame video compression algorithms used for video streaming exploit temporal correlation between frames to achieve extremely high compression gain. Or in other words, the frame rates vary with the video scene content. To avoid the fluctuations in the sending rate, normally the packet scheduler at the sender does a transformation from VBR (variable bit rate) to CBR (constant bit rate), before packets are sent into the network. At the receiver, a counterpart of transformation from CBR to VBR is performed. This requires a pre-decoder for smooth decoding and data rendering at the receiver side. In addition, to handle other network conditions like delay jitter, bandwidth fluctuation, packet loss and the related error recoveries, more buffers are needed. Thus, the actual implementation could either use two independent buffers, the pre-decoder buffer and the jitter buffer, or an integrated buffer.

2.3. Bandwidth characteristics

RealSystem uses a technology called *SureStream* in which one RealVideo clip is encoded for multiple bandwidths [15]. A RealPlayer connects to a single video URL and the RealServer determines which stream to use based on the RealPlayer's specified minimum and average bandwidths. The initial playback data rate of the video stream is based on the maximum client bit rate (a RealPlayer configuration parameter) and other video settings. The actual video stream served can be varied in mid-playout, with the server switching to a lower bandwidth stream during network congestion and then back to a higher bandwidth stream when congestion clears. Unlike previous versions of RealPlayer, RealOne does not need

to re-buffer data during this switching. If packets are lost during video delivery, special packets that correct errors are sent to reconstruct the lost data.

Most RealVideo streams are created with a *Scalable Video Technology* option that allows RealServer to automatically adjust the video stream according to the clients' connection and computer processing speed [15]. If the clip is unable to play at the encoded frame rate on a client machine, it will gradually reduce the frame rate in a controlled fashion to maintain smooth video. The initial size of the video stream is based on the maximum client bit rate (a RealPlayer configuration parameter) and other video settings.

3. Realtracer tools

3.1. Performance metrics

As described in Section 2.3, RealSystem uses SureStream to adjust the actual bandwidth of a video stream served in response to network congestion. It is hence useful to record the variations in streaming bandwidth during the clip playout to better understand how frame rate and jitter change. In addition, network throughput may have great impact on frame jitter [3] and therefore on the quality of the video as perceived by the user. RealTracker measures stream bandwidth every second while a clip is playing.

A basic unit of video performance is the rate at which frames are played. The higher the frame rate, the smoother the video. Very low frame rates are perceived more like a slideshow of still images than of a streaming video. RealTracker records the frame rate every second while a clip is playing. For reference purposes, RealTracker records the encoded frame rate for each clip as well. However, because of media scaling or frame interpolation, the playout frame rate may be higher or lower than this encoded frame rate.

However, even a high frame rate can appear choppy if the frames are not played out at even intervals. In previous work [4], we found that variance, or *jitter*, in frame playout intervals can degrade perceived quality nearly as much as does frame loss. RealTracker measures jitter as the standard deviation of the inter-frame playout time over the entire video clip playout time (1 minute long by default).

Even measures of frame rate and jitter alone are not always sufficient to determine the quality of the video as perceived by the user. During encoding, RealVideo adjusts the frame rate by keeping the frame rate up in high-action scenes, and reducing it in low-action scenes. Thus, an encoded video clip will intentionally not have just one frame rate, but a mix of frame rates that vary with the video scene content. In addition, our previous work [4, 20] shows that the temporal aspect of a streaming video clip has an impact on the effects of reduced frame rate and jitter on perceptual quality. Therefore in addition to the jitter, frame rate and bandwidth, RealTracker records the ratings (from 0–10) of videos watched and rated by users to provide additional performance data beyond measures of jitter and frame rate.

In order to help determine whether the bottleneck to streaming video performance is actually in the end-user's PC itself, RealTracker records the CPU utilization while the client plays the clip. RealTracker also records user information (country, state), and system information (CPU type, memory size, and network configuration) to further help identify

any performance bottlenecks. RealTracker also records the transport protocol. As mentioned in Section 2.3, at the transport layer, RealServer uses both TCP and UDP for sending data, which may result in different behaviors during network congestion.

3.2. Design, implementation and operation of RealTracker

3.2.1. Design and implementation. RealTracker is designed using the RealSystem Software Development Kit³ (SDK) to provide a customized interface to the RealPlayer core. RealTracker has three layers in terms of software structure, as illustrated by figure 1. The first layer includes four modules that provide the RealTracker user interface: the `user information collection` module records the user information entered by the user and the system information detected by RealTracker; the `RealVideo statistics collection` module collects and displays RealVideo statistics through the interfaces provided by layer two; the `data delivery` module delivers the statistics to the locations specified by the user via either email or ftp; the `RealVideo display` module plays out the streamed video through the interfaces provided by layer two.

The second layer acts as a bridge between the first layer and the third layer. It includes two RealSystem interfaces: `IRMAPlayer`⁴ and `IRMAClientAdviseSink`. `IRMAPlayer` lets the top-level client control the presentation playback by beginning, pausing, stopping, or seeking in a presentation timeline. Through this interface, the RealSystem client can also gain access to the client engine, stream objects, and stream source objects. `IRMAClientAdviseSink` lets the top-level client receive notifications from the client core about changes in a presentation's playback status and the corresponding RealVideo statistics. The second layer of RealTracker also includes a module called `Jitter Analysis`, which measures the frame-level jitter, a statistic that is not provided by `IRMAClientAdviseSink`. The third layer is the RealVideo core, provided with `RealPalyer` that is the foundation of RealTracker.

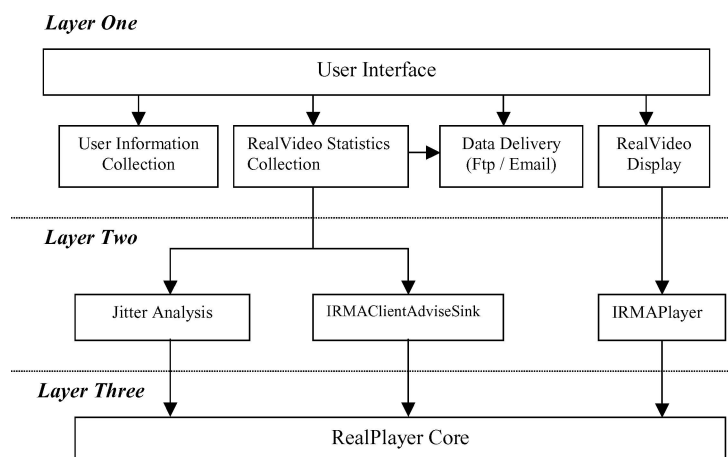


Figure 1. RealTracker structure.

3.2.2. Operation. There are two types of users of RealTracker. The first are users who want to conduct RealVideo performance studies by distributing RealTracker with specific configuration files and playlists as in [23], where RealTracker automatically delivers the RealVideo performance data back to the distributor (see Section 3.2.2.1). The second are users who collect RealVideo statistics by using RealTracker to display the RealVideo clips on the playlist created by themselves (see Section 3.2.2.2).

3.2.2.1. For the distributor. To plan a RealVideo performance study using RealTracker, the distributor needs to create a playlist and a configuration file. The playlist is a text file that contains a list of RealVideo clips to be played by each user. RealTracker randomizes the order of clips in the playlist. The configuration file lets the distributor specify two email addresses and two FTP sites to which the RealVideo statistics will be delivered.

3.2.2.2. For the user. Before users can run RealTracker, they must have RealPlayer or RealOne (either the free version or a subscribed version) installed and the RealTracker support files.⁵ Upon startup, RealTracker requests country, state, and network configuration information from the user as depicted in figure 2(a). In addition, RealTracker automatically detects the Operating System type, CPU type, available RAM and IP address. The main window, depicted in figure 2(b), provides a playlist for video clip selection and allows users to start and stop playing the selected clip. When each clip finishes playout, the user is solicited to assess the video quality by providing a numeric rating from 0–10 as depicted in figure 2(c).

While the video is playing, RealTracer gathers system statistics: encoded bandwidth, measured bandwidth, transport protocol, encoded frame rate, measured frame rate, playout jitter, frames dropped and CPU utilization. The user data and specific clip statistics are then sent via both email and FTP to a server. Users can also choose to save statistics locally to disk.

If so desired, the user can control the length of the clip playout and the requests for quality ratings using the “Options” button. The defaults are to play the clip for 1 minute and request a rating for each clip, proceeding to the next clip after 10 seconds if no rating is given.

3.3. Operation of RealData

RealData provides a file management structure called a *RealFolder* that is used to manage statistics gathered by RealTracker. Each RealData file is associated with a RealVideo clip that was played by RealTracker. All the RealData files and corresponding statistics data within one RealFolder can be exported to a comma delimited file, suitable for import into most spreadsheet programs, such as Microsoft Excel, for further analysis. RealData also provides a tool that can generate cumulative density data (used for drawing CDF graphs) for measured bandwidth, frame rate and jitter.

4. Results

In this section we present results obtained from our use of the RealTracer tools. The results presented are meant to show some of the possible usages of the RealTracer tools rather than only general results about the use of streaming media on the Internet.



User Information

Personal

Name: Jack Smith

E-Mail Address: Jack_Smith@foo.com

Location

Country: USA

State/Province: MA

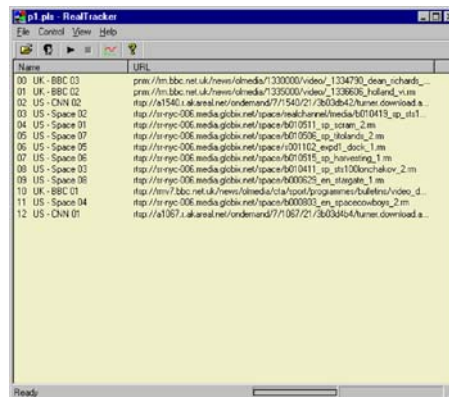
System

Cpu Type: Intel Pentium II

Network Configuration: DSL/Cable

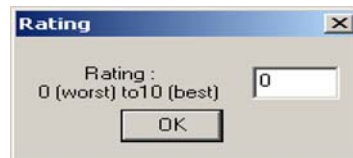
OK Cancel

(a)



Name	URL
00 UK - BBC 03	http://m.bbc.net.uk/news/1media/1300000/video/1334790_dean_schard...
01 UK - BBC 02	http://m.bbc.net.uk/news/1media/1335000/video/1336606_holland_v...
02 US - CNN 02	http://a1540.akamaized.net/on-demand/71540/21/7803db42/amer.download.s...
03 US - Space 02	http://rr-ny0-006.media.globalsite.net/pace/4070413_sp_st1...
04 US - Space 01	http://rr-ny0-006.media.globalsite.net/pace/4070511_sp_scream_2.m...
05 US - Space 07	http://rr-ny0-006.media.globalsite.net/pace/4070506_sp_islands_2.m...
06 US - Space 05	http://rr-ny0-006.media.globalsite.net/pace/4071022_esp01_block_1.m...
07 US - Space 06	http://rr-ny0-006.media.globalsite.net/pace/4070515_sp_hairstyng_1.m...
08 US - Space 03	http://rr-ny0-006.media.globalsite.net/pace/4070411_sp_sts100nchakov_2.m...
09 US - Space 06	http://rr-ny0-006.media.globalsite.net/pace/4000925_em_01sp04_1.m...
10 UK - BBC 01	http://m.bbc.net.uk/news/1media/1314000/program/bullefmu/video_d...
11 US - Space 04	http://rr-ny0-006.media.globalsite.net/pace/4000803_em_spaceshow_2.m...
12 US - CNN 01	http://a1067.akamaized.net/on-demand/71067/21/7803db42/amer.download.s...

(b)



Rating

Rating : 0 (worst) to 10 (best)

0

OK

(c)

Figure 2. (a) RealTracker user information window. (b) RealTracker main window. (c) RealTracker clip rating window.

4.1. Approach

In order to empirically evaluate the performance of RealVideo across the Internet using RealTracker, we employed the following methodology:

- Set up a Web site to allow users to download RealTracker.
- Select RealVideo servers from geographically diverse Web sites and choose diverse video clips from those sites.



Figure 3. Geographic depiction of RealServers and users.

- Solicit users to run RealTracker and gather data.
- Analyze the results using RealData.

In order to let users easily download RealTracker, we set up a Web site for the RealTracker software and corresponding support files, along with the detailed usage instructions.⁶ We also configured RealTracker so that the data could be sent via both email and FTP to a server at Worcester Polytechnic Institute.

We chose RealServers accessible through Web pages from 6 geographic regions: Asia, Australia, Europe, Japan, North America, and South America. Within each region, popular RealNetworks sites were chosen from: Australia, Brazil, Canada, China, Italy, Japan, the United Kingdom, and the United States. Figure 3 depicts a geographic representation of the RealServer sites chosen. From each site, we selected a variety of video content among all the videos that were offered.

Once the servers and videos were selected, we solicited friends, family and colleagues from various parts of the world to help in the study. Since it was fairly easy for us to obtain data points from Massachusetts, we asked friends and colleagues on campus and at work to solicit help from people they knew outside of Massachusetts. We also posted messages asking for help to the `rec.video` newsgroup and `end2end-interest` mailing list.

We then gathered data from users running RealTracker for an 11 day period from June 4, 2001 to June 15, 2001. Figure 3 depicts a geographic representation of the locations of users that ran RealTracker.

4.2. Results and analysis

As described in Section 3, the system statistics gathered by RealTracker while a clip is being played includes: encoded bandwidth, measured bandwidth, transport protocol, encoded frame rate, measured frame rate, playout jitter, frames dropped, CPU utilization and perceptual quality ratings. Here, our analysis of these statistics using RealData focuses on frame rate and bandwidth.

A total of 63 users from 12 different countries participated in the study, playing a total of 2855 clips, among which 2100 were from the US and 352 were from Europe.

We first analyze the recorded frame rates using RealData. The key frame rate targets we observe are [15]:

- The standard frame rate for full-motion video is 24 to 30 frames per second (fps). At this speed, the human eye perceives movement as continuous, without seeing individual frames.
- A common frame rate for computer video that approximates full-motion video is 15 fps. To most people, a 15 fps video flows smoothly, although for some videos, it will not appear quite as fluid as it would at a higher frame rate.
- Below 15 fps, a video looks choppy.
- Below 7 fps, a video looks very choppy.
- Below 3 fps, a video essentially becomes a series of still pictures.

4.2.1. Frame rate & network connection. In our analysis, we concentrate on frame rates of 3, 15 and 25 frames per second. Figure 4 shows a CDF of the frame rate for all the video clips played. The mean frame rate is 10 fps, above the range of really choppy video but well short of very fluid video. Approximately 25% of all videos played are under the minimum acceptable 3 fps, while the same number (25%) of videos are played at the approximate full-motion video rate, 15 fps. Only a very small fraction, less than 1%, of all videos achieve true full-motion video frame rates.

With the increase in high-speed Internet connections for home users, we may see more bottlenecks to performance in the server and not in the end-host network. Figure 5 depicts a CDF of frame rate for different end-user network configurations. The frame rates afforded

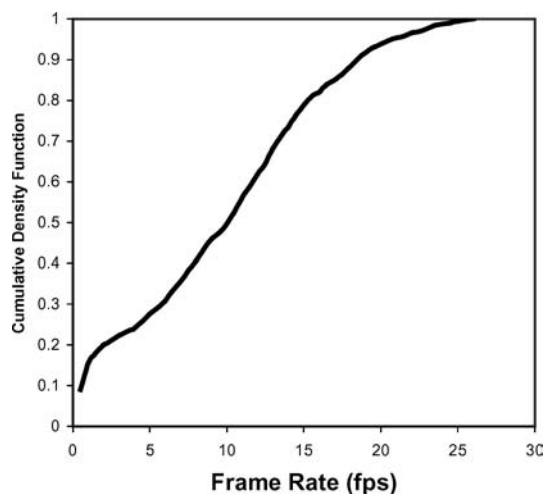


Figure 4. CDF of frame rate for all video clips.

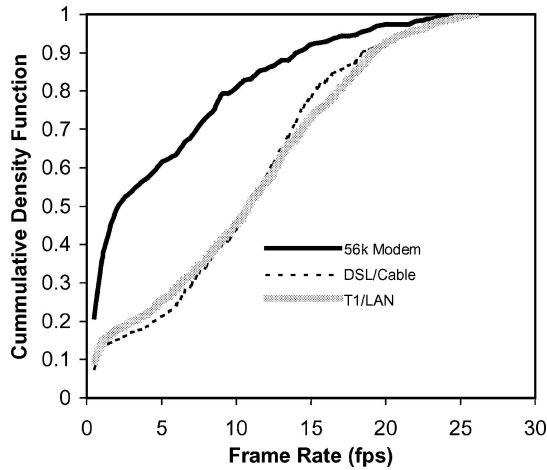


Figure 5. CDF of frame rate for different end-host network configurations.

by modem connections are clearly worse than the frame rates with higher speed connections. Over half of all videos streamed over modems play out at less than 3 fps, and less than 10% of videos streamed over modems achieve a smooth 15 fps. Contrast this to the higher speed connections in which only 20% of videos have frame rates less than 3 fps, while nearly 30% of videos play out at 15 fps. Also, high-speed home-Internet connections afforded by DSL and Cable modems provide nearly the same performance for streaming video, as do higher-speed T1/LAN connections. This suggests that video performance bottlenecks are increasingly less likely to be the end-user connection. Figure 6 depicts the bandwidth

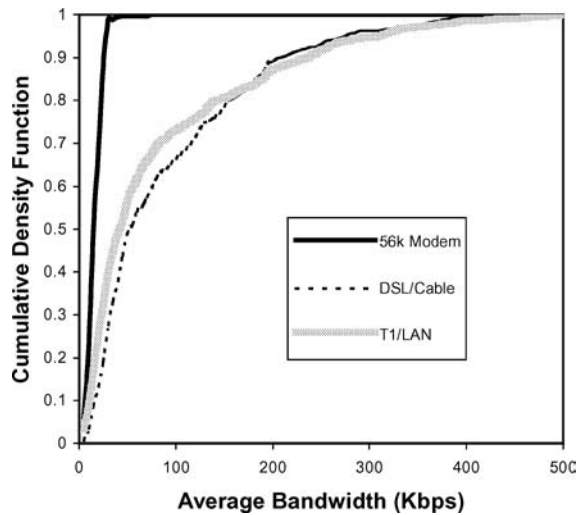


Figure 6. CDF of bandwidth for different end-host network configurations.

achieved by each class of end-host network configuration. Notice that DSL/Cable modems that can typically achieve throughputs from 256–512 Kbps or more, operate near full capacity less than 10% of the time. This further suggests that the bottleneck to video bandwidth is beyond the end-host connection. By comparing figure 5 with figure 6, it can be seen that modem connections get a proportionally higher frame rate for their network bandwidth than do higher-speed connections.

4.2.2. Frame rate & geography. It may be expected that servers in “wired” geographic areas, say North America, will provide better streaming video performance than other locations, say Brazil. Figure 7 depicts a Cumulative Density Function (CDF) of the frame rate for the servers used in our study, separated into 5 different geographic regions. The 5 regions all provide very similar frame rate distribution shapes, although the median of the best frame rate distribution is about 13 fps and the median of the worst frame rate distribution is about 8 fps. Australia and Europe have the best frame rate distributions, with Europe providing a larger percentage of frame rates above 20 fps. Asia provides the worst frame rates, but the differences at very low frame rates is small, and Asia servers actually have a larger percentage of frame rates above 15 fps than do North America servers.

While the peering richness of a client’s ISP may largely determine video performance, this is difficult to measure. Thus, we consider geographic region, which is easy to measure, in place of client ISP information. Similarly to servers, it may also be expected that users in well “wired” geographic areas will observe better frame rates than users in more technologically remote areas. Figure 8 depicts a CDF of frame rate for the users in our study, separated into 4 geographic regions. In this case, geographic region appears to more clearly differentiate

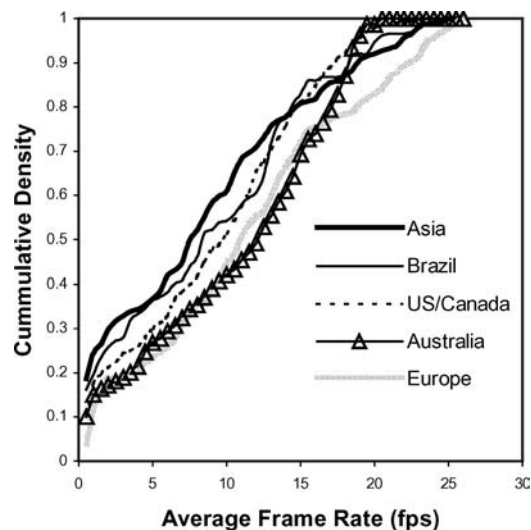


Figure 7. CDF of frame rate for real-servers in different geographic regions.

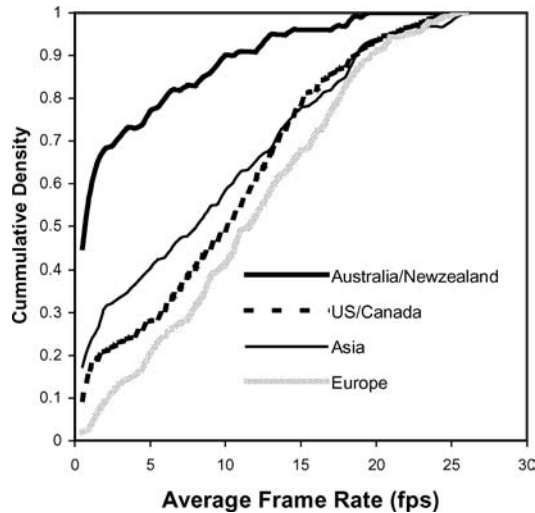


Figure 8. CDF of frame rate for users in different geographic regions.

streaming video performance than it did in the case of the servers. Australia/New Zealand provides the worst frame rates for all ranges, with 75% of videos having fewer than 3 fps and less than 10% of videos having more than 15 fps. Clips played in Europe have the best frame rates up to 15 fps, with only 15% of videos having less than 3 fps and 25% of videos getting more than 15 fps. North America is slightly better than Asia up to the 15 fps rate. Europe, North America and Asia all provide about the same percentage of videos with frame rates above 20 fps.

Users often connect to local servers either explicitly or implicitly to get better performance. Content Delivery Networks (CDNs), in particular, attempt to make content more local for each user. While local access has been shown to be effective for improving the performance of Web content [10], the effects of local content on the performance of streaming media have not been studied thoroughly. We use RealData to analyze whether local access can achieve better video performance, too. Figure 9 depicts a CDF of frame rate for the combinations of RealServers and clients in different geographic regions. The access of US clients to US servers provides a similar frame rate distribution shape as the access of European clients to European servers. However, European clients to European servers provides slightly better frames rate than that of European clients to non-European servers. Overall, the 4 combinations all provide very similar frame rate distributions. This is consistent with our observation in figure 7, which suggests there is very little difference in streaming video that is served from different geographic regions. In contrast, the bandwidth distribution shows strong locality as shown by figure 10. The US clients to US servers achieve better bandwidth than that of US clients to non-US servers while the European clients to non-European servers have better bandwidth than that of European clients to European servers. This suggests media streaming may benefit from the effects of localized proxy caches and CDNs.

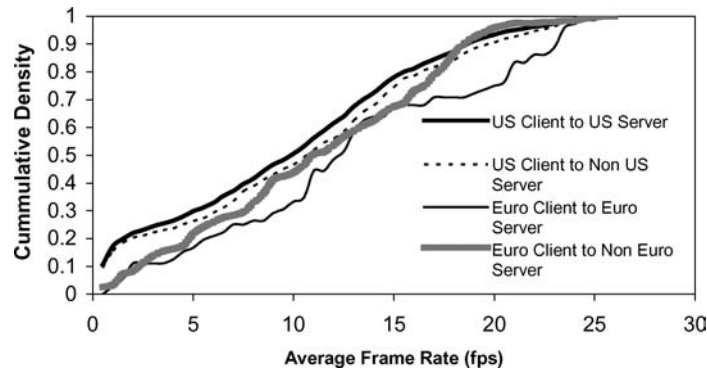


Figure 9. CDFs of frame rate for Users/RealServers in different geographic regions.

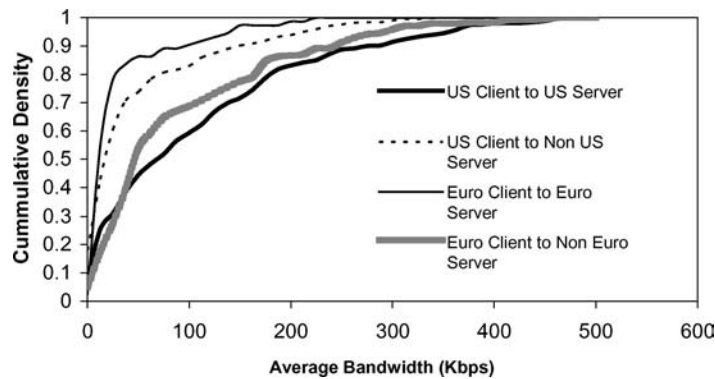


Figure 10. CDFs of bandwidth for Users/RealServers in different geographic regions.

It is well known that time of the day has significant impact on the quality of an Internet connection. However, our data analysis shows little correlation between our measurements of video performance and UTC time of the day. This may be because both clients and servers are distributed across multiple time zones.

4.2.3. Jitter. We next use RealData to examine jitter as recorded by RealTracker. As mentioned in 3.2.1, we measure the frame level jitter, i.e., the residual jitter. Since human perception of delay for interactive applications is around 100 ms, we focus on the percentage of videos that have 50 ms of jitter or less.⁷ In addition, jitter events that are larger than the average inter-frame playout are most noticeable by users, so we also focus on the percentage of videos that have a standard deviations of about 300 ms (about the average inter-frame playout time for the minimum acceptable 3 fps rate) or greater, as this may be a reasonable upper bound on an acceptable amount jitter. Figure 11 depicts a CDF of jitter (standard deviation of inter-frame playout times) for all the video clips played. Just over 50% of all

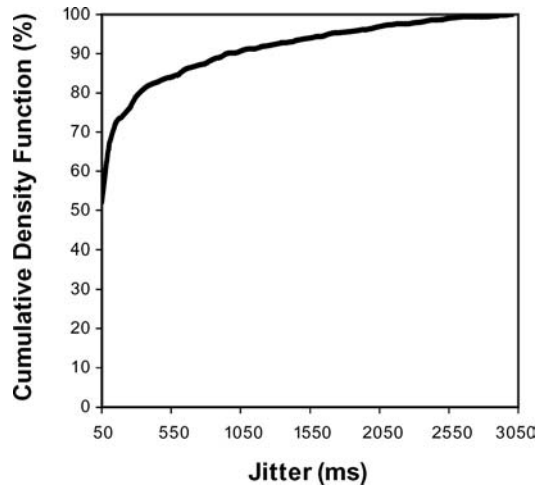


Figure 11. CDF of overall jitter.

videos play with very little perceptible jitter. This high percentage of smooth videos is most likely due to the buffers in the RealPlayer core when the video connection is first made. Only about 15% of all videos play out with a potentially unacceptable 300 ms or more of jitter.

Our expectation is that the frame rate results for different end-host network configurations should hold for jitter, as well. Based on [3], we expect high-speed Internet connections to have less jitter than slower Internet connections. Figure 12 depicts a CDF of jitter for different end-host network configurations. From the graph, jitter in video played out over

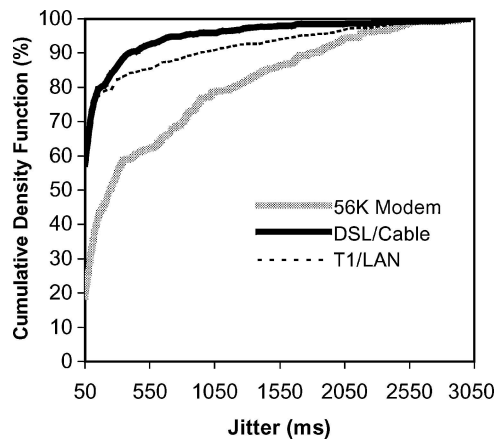


Figure 12. CDF of jitter for different network configurations.

a modem is typically much greater than jitter over a higher-speed connection. Video clips played over a modem have no perceptible jitter only about 10% of the time and have potentially unacceptable jitter nearly 45% of the time. DSL/Cable modems and T1/LAN connections have a nearly identical percentage of perceptually jitter-free streams, while DSL/Cable modems also have a smaller percentage of potentially unacceptable amounts of jitter (15% vs. 20%, respectively). Overall, DSL/Cable modems have better jitter distributions, possibly because users contend with fewer other users for bandwidth, causing less variance in bandwidth than occurs on corporate LANs.

4.2.4. Perceptual quality. The end-host network configuration has one of the largest impacts on video frame rate and jitter. We expect the impact of end-host network configuration to hold for perceptual quality ratings, too. We first analyze the perceptual quality of all RealVideo clips across the Internet, depicted as a CDF in figure 13. The median perceptual quality rating is about 5 and the distribution is very uniform across all quality rating values. This suggests there may be a “normalization” of ratings that causes users to provide an average rating of 5 for the video clips they watch, regardless of the system conditions. Figure 14 depicts a CDF of quality rating for different end-user network configurations. The end-host network has a large impact on perceptual quality. The average video watched over a modem is only about half as good in perceived quality as the average video watched on a DSL/Cable modem. DSL/Cable modems have better perceptual quality distributions than do LAN/T1 connections. This difference was not evident in the frame rate CDF for network configuration (figure 6) but was evident in the jitter CDF for network configuration (figure 12), suggesting that jitter is differentiating the video quality between the two configurations. We also briefly examined if there is any relationship between perceptual quality and systems measures of video performance. Overall, we have found there to be very little

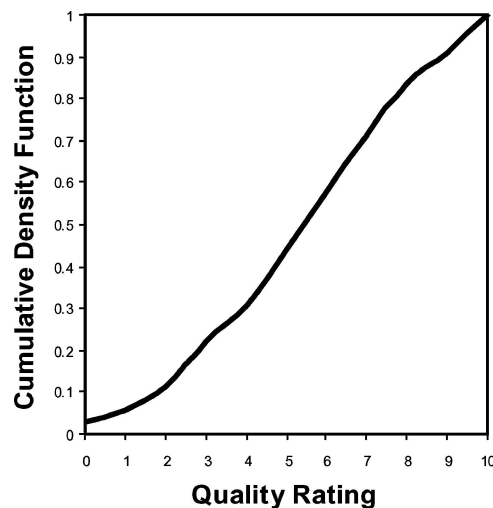


Figure 13. CDF of overall quality.

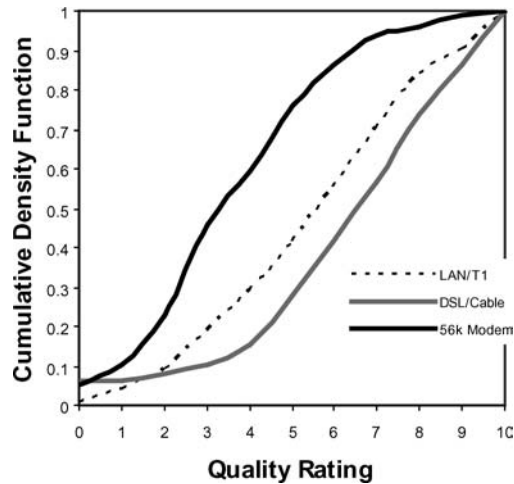


Figure 14. CDF of quality for different end host network configurations.

visual correlation between system measurements and configuration and perceptual quality when taken over all users.

5. Conclusions

In recent years, the use of commercial streaming products has increased dramatically due to the growth of high-bandwidth connections and high-speed computers. The future impact of streaming video on the Internet will be largely impacted by the role of commercial streaming media products. However, to date there have not been sufficient wide-scale empirical measurements of video performance across the Internet, which may largely stem from the lack of effective video performance measurement tools.

In this paper, we have presented the design, implementation and operation of a new tool suite, called RealTracer, for measuring and analyzing RealNetworks Video (RealVideo) performance. RealTracer includes RealTracker, a customized player that can play streaming RealVideo clips, record system performance statistics as well as user perceptual quality ratings, along with RealData, a tool that helps analyze the statistical data collected by RealTracker. RealTracker is implemented using the RealSystem RealOne Software Development Kit to control RealVideo playback and gather the corresponding statistics. With RealTracker, users can create their own playlist and specify the means of gathering statistics. With RealData, users can sort the RealVideo statistics, generate cumulative density data, and export the statistical data.

We have successfully used RealTracer to conduct a wide-scale empirical study of RealVideo traffic from several Internet servers to many geographically diverse users [24]. The RealTracer tools helped us find:

- The average RealVideo clip streamed over the Internet has good quality, playing out at 10 fps and, aided by a large, initial delay buffer, with nearly imperceptible amounts of inter-frame jitter. Users connecting to the Internet with modems and/or slow computers still have their PC or their network connection as the video performance bottleneck, while typical new computers connecting to the Internet via DSL or Cable modem achieve even slightly better performance than corporate network connections to the Internet. The fact that these higher speed links are under utilized for streaming video suggests that increasing broadband connections for home users are pushing the bottlenecks for video performance closer to the server.
- There is very little difference in streaming video that is served from different countries, but there are distinct performance differences from video that is received in different countries.
- While local access has been shown to be effective for Web content, our results suggest media streaming may also benefit from the effects of localized proxy caches and CDNs.

6. Future work

RealTracker only records user-centric video performance information. Our future work could seek to broaden the data set of both users and servers. In doing so, RealTracker could also gather statistics from RealServers. RealServer plug-ins can monitor resources in the RealServer Property Registry, which is a dynamic repository for a variety of server and client properties. These properties include such values as the number of clients currently connected, the total bandwidth being utilized, and a comprehensive set of statistics for each of these clients. A monitor plug-in can monitor any registry property, receiving notification when RealServer updates the property. Monitor plug-ins can also add their own properties to the registry and receive notifications when those properties change.

The major commercial competitor to RealNetwork's RealPlayer is Microsoft's MediaPlayer.⁸ Developing similar tools to RealTracer for Media Player, perhaps a *MediaTracer* tool suite, would enable an empirical study of more general video performance on the Internet.

7. RealTracer Availability

The RealTracer programs and source code are publicly available at: <http://perform.wpi.edu/real-tracer/>

Acknowledgments

We would especially like to thank Zheng Zuo for his help in implementing the RealTracer tools. We would also like to thank all the reviewers for reading this paper and providing valuable comments.

Notes

1. <http://www.tcpdump.org/>
2. Download at the Video Tracer web site: <http://perform.wpi.edu/real-tracer/>
3. <http://www.realnetworks.com/resources/sdk/index.html>
4. “IRMA” stands for “Interface RealMedia Architecture” and is named in the SDK.
5. See <http://perform.wpi.edu/real-tracer/>
6. See <http://mile.wpi.edu/real-tracer/#realtracer> for the specific instructions given to users.
7. Based on the empirical assumption that approximately 95% of the playout times are within two standard deviations of the mean [5].
8. <http://www.microsoft.com/windows/windowsmedia/en/default.asp>

References

1. D. Cunningham and N. Francis, “An introduction to streaming video, cultivate interactive,” European Commission’s Digital heritage and Cultural Content (DIGICULT), May 2001.
2. M. Claypool and J. Riedl, “The effects of high-speed networks on multimedia jitter,” in Proc. of SCS Euromedia Conference (COMTEC), Munich, Germany, April 1999.
3. M. Claypool and J. Tanner, “The effects of jitter on the perceptual quality of video,” in Proc. of the ACM Multimedia Conference, Orlando, Florida, USA, Vol. 2, November 1999.
4. M. Chesire, A. Wolman, G. Voelker, and H. Levy, “Measurement and analysis of a streaming-media workload,” in Proc. of the 3rd USENIX Symposium on Internet Technologies and Systems (USITS), March 2001.
5. A.C. Dalal and E. Perry, “A new architecture for measuring and assessing streaming media quality”, in Proc. of the Passive and Active Measurement Workshop (PAM), (Poster), La Jolla, California, April 2003
6. J. Devore and R. Peck, Statistics — The Exploration and Analysis of Data, 2nd edition, Wadsworth, Inc., 1993.
7. S. Floyd, M. Handley, J. Padhye and J. Widmer, “Equation-based congestion control for unicast applications,” in Proc. of ACM SIGCOMM Conference, 2000, pp. 45–58.
8. Jupiter Media Metrix, “Users of media player applications increased 33 percent since last year, press release,” April 2001, available at <http://www.jup.com/company/pressrelease-.jsp?doc=pr01040>
9. B. Krishnamurthy and C. Wills, “Analyzing factors that influence end-to-end web performance,” in Proc. of the Ninth International World Wide Web Conference, Amsterdam, Netherlands, May 2000.
10. B. Krishnamurthy, C. Wills, and Y. Zhang, “On the use and performance of content distribution networks,” in Proc. of the ACM SIGCOMM Internet Measurement Workshop, San Francisco, California, USA, November 2001.
11. D. Loguinov and H. Radha, “Measurement study of low-bitrate internet video streaming,” in Proc. of the ACM SIGCOMM Internet Measurement Workshop, San Francisco, California, USA, November 2001.
12. B. Mah, “An empirical model of HTTP network traffic,” in Proc. of ACM SIGCOMM Conference, 1997, pp. 301–313.
13. A. Mena and J. Heidemann, “An empirical study of real audio traffic,” in Proceedings of the IEEE Infocom, Tel-Aviv, Israel, March, 2000, pp. 101–110.
14. V. Paxson, “End-to-end Internet packet dynamics,” IEEE/ACM Transactions on Networking, Vol. 7, No. 3, pp. 277–292, 1999.
15. Real Networks Incorporated, RealProducer User’s Guide, copyright 2000, available at <http://www.service.real.com/help/library/guides/producerplus85/producer.htm>
16. Real Networks Incorporated, RealPlayer 8 User Manual, copyright 2000.
17. R. Rejaie, M. Handley, and D. Estrin, “RAP: An end-to-end rate-based congestion control mechanism for realtime streams in the internet,” in proc. of IEEE INFOCOM conference, 1999.
18. H. Schulzrinne, A. Rao, and R. Lanphier, “Real time streaming protocol (RTSP),” RFC 2326, April 1998, available at <http://www.rfc-editor.org/rfc/rfc2326.txt>
19. H. Schulzrinne, S. Casner, R. Frederick, and V. Jacobson, “RTP: A transport protocol for real-time applications,” RFC 1889, January 1996, available at <http://www.rfc-editor.org/rfc/rfc1889.txt>.

20. A. Tripathi and M. Claypool, "Improving multimedia streaming with content-aware video scaling," in Proceedings of the Second International Workshop on Intelligent Multimedia Computing and Networking (IMMCN), Durham, North Carolina, USA, March 8–12, 2002
21. K. Thompson, G. Miller and R. Wilder, "Wide-area internet traffic patterns and characteristics," IEEE Network, Nov/Dec 1997.
22. J.E. van der Merwe, R. Cáceres, Y.-H. Chu, and C.J. Sreenan, "mmdump—A tool for monitoring internet multimedia traffic," ACM Computer Communication Review, Vol. 30, No. 4, 2000.
23. E. Veloso, V. Almeida, W. Meira, A. Bestavros, and S. Jin, "A hierarchical characterization of a live streaming media workload," in Proc. of the SIGCOMM Internet Measurement Workshop, Marseille, France, Nov 2002.
24. Y. Wang, M. Claypool, and Z. Zuo, "An empirical study of RealVideo performance across the internet," in Proc. of the ACM SIGCOMM Internet Measurement Workshop, San Francisco, California, USA, November 2001.

Yubing Wang earned M.S. in Electrical Engineering from University of Science and Technology of China in 1993 and M.S. in Computer Science from WPI in 2001. He is currently working in EMC Corp. as a principal software engineer, participating in the developments of several EMC NAS products. He is also a Ph.D. candidate in Computer Science of WPI. His primary research interests include multimedia networking and distributed file system.



Mark Claypool earned M.S. and Ph.D. degrees from the University of Minnesota in 1993 and 1997, respectively. Dr. Claypool joined the Computer Science department of WPI in 1997, receiving tenure and promotion to Associate Professor in 2004. He is also the Director of the Interactive Media and Game Development major at WPI, a 4-year degree in the principles of interactive applications and computer-based game development. His primary research interests include multimedia networking, congestion control, and network games.