ISSN 2348-1196 (print)
International Journal of Computer Science and Information Technology Research ISSN 2348-120X (online)
Vol. 3, Issue 3, pp: (101-106), Month: July - September 2015, Available at: www.researchpublish.com

Commercial Peer-To-Peer Video Streaming To Avoid Delay Transmission

S.Karthika[1], R.Anitha[2]

[1]Research Scholar, [2] Associate Professor
[1, 2] Sengamala Thayaar Educational Trust Womens College, Thiruvarur, India – 614001

Abstract: **A number of commercial peer-to-peer (P2P) systems for live streaming have been introduced in recent years. The behaviour of these popular systems has been extensively studied in several measurement projects. Due to the proprietary nature of these commercial systems, however, these studies have to rely on a "black-box" approach, where packet traces are collected from a single or a limited number of measurement points, to infer various properties of traffic on the control and data planes. Although such studies are useful to compare different systems from the end-user's perspective, it is difficult to intuitively understand the observed properties without fully reverse-engineering the underlying systems. In this project, we describe the network architecture of Zattoo, one of the largest production live streaming providers in Europe at the time of writing, and present a large-scale measurement study of Zattoo using data collected by the provider. To highlight, we found that even when the Zattoo system was heavily loaded with as high as 20 000 concurrent users on a single overlay, the median channel join delay remained less than 2–5 s, and that, for a majority of users, the streamed signal lags over-the-air broadcast signal by no more than 3 s.**

Keywords: **Live streaming, network architecture, peer-to peer (P2P) system.**

I. INTRODUCTION

There is an emerging market for IPTV. Numerous commercial systems now offer services over the Internet that are similar to traditional over-the-air, cable, or satellite TV. Live television, time-shifted programming, and content-on-demand are all presently available over the Internet. Increased broadband speed, growth of broadband subscription base, and improved video compression technologies have contributed to the emergence of these IPTV services. We draw a distinction between three uses of peer-to-peer (P2P) networks: delay-tolerant file download of archival material, delay-sensitive progressive download (or streaming) of archival material, and real-time live streaming. In the first case, the completion of download is elastic, depending on available bandwidth in the P2P network. The application buffer receives data as it trickles in and informs the user upon the completion of download. The user can then start playing back the file for viewing in the case of a video file. Bittorrent and variants are examples of delay-tolerant file download systems. In the second case, video playback starts as soon as the application assesses it has sufficient data buffered that, given the estimated download rate and the playback rate, it will not deplete the buffer before the end of file. If this assessment is wrong, the application would have to either pause playback and rebuffer or slow down playback. While users would like playback to start as soon as possible, the application has some degree of freedom in trading off playback start time against estimated network capacity. Most video-on-demand systems are examples of delay-sensitive progressive-download application. The third case, real-time live streaming, has the most stringent delay requirement. While progressive download may tolerate initial buffering of tens of seconds or even minutes, live streaming generally cannot tolerate more than a few seconds of buffering. Taking into account the delay introduced by signal ingest and encoding, and network transmission and propagation, the live streaming system can introduce only a few seconds of buffering time end-to-end and still be considered "live" . The Zattoo peer-to-peer live streaming system was a free-to-use network serving over 3 million registered users in eight European countries at the time of study, with a maximum of over 60 000 concurrent users on a single channel. The system delivers live streams using a *receiver-based, peer-division multiplexing* scheme as described in Section II. To ensure real-time performance when peer uplink capacity is below requirement, Zattoo subsidizes the network's bandwidth requirement, as described in Section III. After delving into Zattoo's architecture in detail, we study in Sections IV and V large-scale measurements collected during the live broadcast of the UEFA European Football Championship, one of the most popular one-time events in Europe, in June 2008. During the course of that month, Zattoo served more than 35 million sessions to more than 1 million distinct users. Drawing from these measurements, we report on the operational scalability of Zattoo's live streaming system along several key issues.

II. PROBLEM DEFINITION

A number of commercial peer-to-peer (P2P) systems for live streaming have been introduced in recent years. The behavior of these popular systems has been extensively studied in several measurement projects. Due to the proprietary nature of these commercial systems, however, these studies have to rely on a "black-box" approach, where packet traces are collected from a single or a limited number of measurement points, to infer various properties of traffic on the control and data planes. Although such studies are useful to compare different systems from the end-user's perspective, it is difficult to intuitively understand the observed properties without fully reverse-engineering the underlying systems. In this project, we describe the network architecture of Zattoo, one of the largest production live streaming providers in Europe at the time of writing, and present a large-scale measurement study of Zattoo using data collected by the provider. To highlight, we found that even when the Zattoo system was heavily loaded with as high as 20 000 concurrent users on a single overlay, the median channel join delay remained less than 2–5 s, and that, for a majority of users, the streamed signal lags over-the-air broadcast signal by no more than 3 s.

A. Problem Analysis:

Systems now offer services over the Internet that are similar to traditional over-the-air, cable, or satellite TV. Live television, time-shifted programming, and content-on-demand are all presently available over the Internet. Increased broadband speed, growth of broadband subscription base, have contributed to the emergence of these IPTV services. We draw a distinction between delay-tolerant file download of archival material, delay-sensitive progressive download (or streaming) of archival material, and real-time live streaming. Most video-on-demand systems are examples of delay-sensitive progressive-download application. live streaming generally cannot tolerate more fault tolerance due to increased encoding time over the data retrieval of stream servers Taking into account the delay introduced in the streaming affects channel ingest and encoding, and network transmission and propagation, This leads to the live streaming system can introduce only a few seconds of buffering time end-to-end for a an short streaming ratio with enhanced delay of buffering and channel data encoding

B. Problem Solution:

The ZATOO peer-to-peer live streaming system was a free-to-use network ,with a maximum of over 60 000 concurrent users on a single channel. The system delivers live streams using a receiver-based, peer-division multiplexing scheme To ensure real-time performance when peer uplink capacity is below requirement, we report on the operational scalability of Zattoo's live streaming system along several key issues. Scaleing of system in terms of overlay size and its effectiveness in utilizing peers' uplink bandwidth .enhanced packet retransmission scheme in allowing a peer to recover from transient congestion bye the multiplexing networking methodology.Delay over the data channel fault tolerance is enhanced over by the zatoo encoding scheme for enhancing data buffer size regardless of the rate of transmission with data loss .

The Zattoo system rebroadcasts live TV, captured from satellites, onto the Internet. The system carries each TV channel on a separate peer-to-peer delivery network and is not limited in the number of TV channels it can carry. Although a peer can freely switch from one TV channel to another, thereby departing and joining different peer-to-peer networks, it can only join one peer-to-peer network at any one time. We henceforth limit our description of the Zattoo delivery network as it pertains to carrying one TV channel. Fig. 1 shows a typical setup of a single TV channel carried on the Zattoo network. TV signal captured from satellite is encoded into H.264/AAC streams, encrypted, and sent onto the Zattoo network. The encoding server may be physically separated from the server delivering the encoded content onto the Zattoo network. For ease of exposition, we will consider the two as logically colocated on an Encoding Server. Users are required to register themselves at the Zattoo Web site to download a free copy of the Zattoo player application. To receive the signal of a channel, the user first authenticates itself to the Zattoo Authentication Server. Upon authentication, the user is granted a

ISSN 2348-1196 (print)
International Journal of Computer Science and Information Technology Research ISSN 2348-120X (online)
Vol. 3, Issue 3, pp: (101-106), Month: July - September 2015, Available at: www.researchpublish.com

ticket with limited lifetime. The user then presents this ticket, along with the identity of the TV channel of interest, to the Zattoo Rendezvous Server. If the ticket specifies that the user is authorized to receive signal of the said TV channel, the Rendezvous Server returns to the user a list of peers currently joined to the P2P network carrying the channel, together with a signed channel ticket. If the user is the first peer to join a channel, the list of peers it receives contain only the Encoding Server.

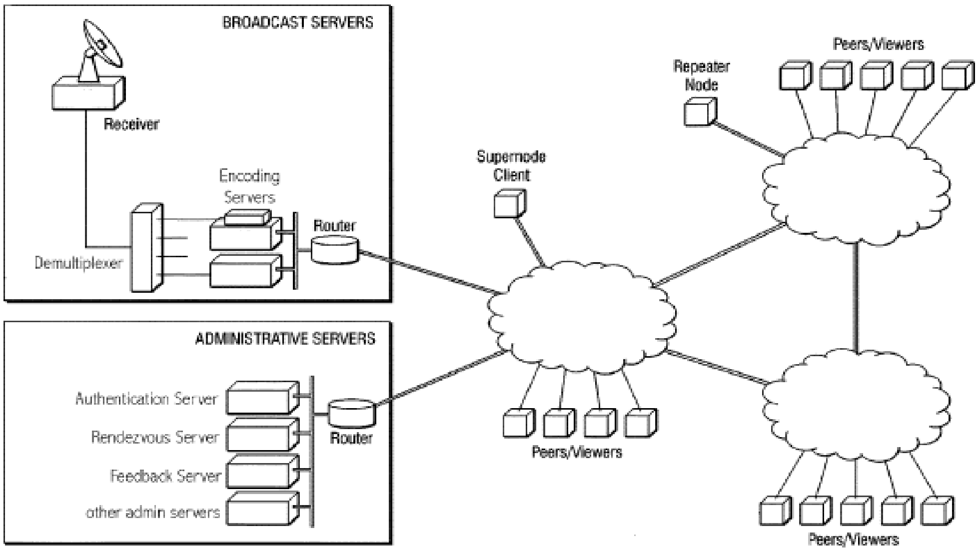
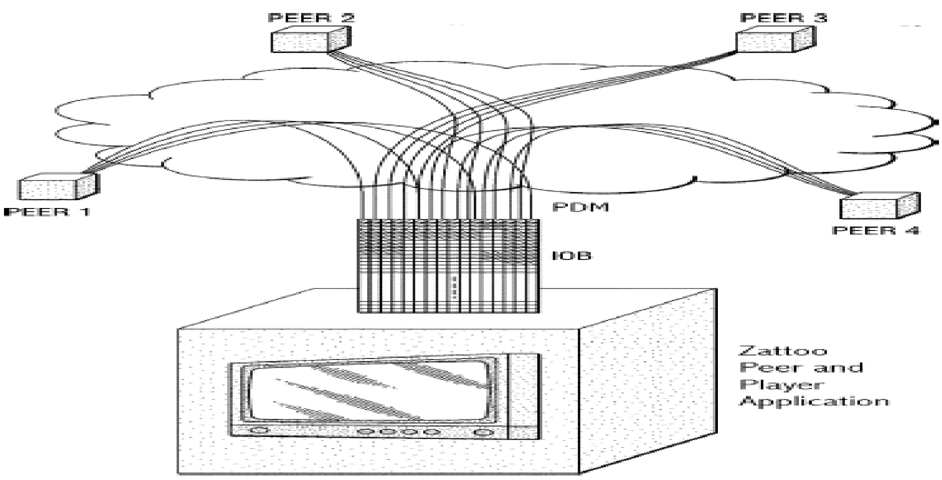


Figure 1 Zattoo delivery network architecture

III. PROCESS FLOW

A. Peer-Division Multiplexing:

The Server returns to the user a list of peers currently joined to the P2P network carrying the channel, together with a signed channel ticket.The user joins the channel by contacting the peers returned by the Rendezvous Server, presenting its channel To minimize per-packet processing time of a stream, the Zattoo protocol sets up a virtual circuit with multiple fan outs at each peer. When a peer joins a channel, it establishes a peer-division multiplexing (PDM) scheme among a set of neighboring peers by building a virtual channel to each of the neighboring peers. Baring departure or performance degradation of a neighbor peer, the virtual channel are maintained until the joining peer switches to another channel.This initiates minimizing inter domain traffic and thus saving on transit bandwidth cost, but also helps reducethe number of physical links and metro hops traversed in the overlay network, potentially resulting in enhanced user-perceived stream quality.



b. Encoding Server:

By the time the capacity metric percolates up to the Encoding Server, it contains the total download and upload rate aggregates of the whole streaming overlay. The Encoding Server then simply forwards the obtained to the Subsidy Server. to the Subsidy System. The keep-alive message tells the Subsidy System which channel the Repeater node is serving, plus its CPU and capacity utilization ratio.The channel utilization ratio ( ) is the ratio over based on the encoding scheme over took at the zatoo methodology of live streaming .we classify the capacity trend of each channel on the basis of encoding scheme.

• **Stable:** The ratio has remained within for the past reporting periods.

• **Exploding:** The ratio increased by at least between (e.g.,) reporting periods.

• **Increasing:** The ratio has steadily increased by () over the past reporting periods.

• **Falling:** The ratio has decreased by over the past reporting periods.statistics from individual peers,

c. Stream Management:

We represent a peer as a packet buffer,A local media player server if one is running.A local file server if recording is supported and potentially other peers.Zattoo player application with virtual server established to four peers. As packets from each substream arrive at the peer, they are stored in the reassembly to reconstruct the full stream. Portions of the stream that have been reconstructed at the encoding server are then played back to the user. The input pointer points to the slot in the peer next server where the next incoming packet with sequence number higher than the highest sequence number received so far will be stored. The repair pointer always points one slot beyond the last packet received in order.

IV. EXPERIMENTAL RESULTS

A. Implementation:

In addition to achieving lossless recording, we use retransmission to let a peer recover from transient network congestion. A peer sends out a retransmission request when the distance between the repair pointer and the input pointer has reached a threshold of packet slots, usually spanning multiple segments. A retransmission request consists of an -bit packet mask, with each bit representing a packet, and the sequence number of the packet corresponding to the first bit. Marked bits in the packet mask indicate that the corresponding packets need to be retransmitted. When a packet loss is detected, it could be caused by congestion on the virtual circuits forming the current PDM or congestion on the path beyond the neighboring peers. In either case, current neighbor peers will not be good sources of retransmitted packets. Hence, we send our retransmission requests to random peers that are not neighbor peers. A peer receiving a retransmission request will honor the request only if the requested packets are still in its IOB and it has sufficient leftover capacity, after serving its current peers, to transmit all the requested packets. Once a retransmission request is accepted, the peer will retransmit all the requested packets to completion.

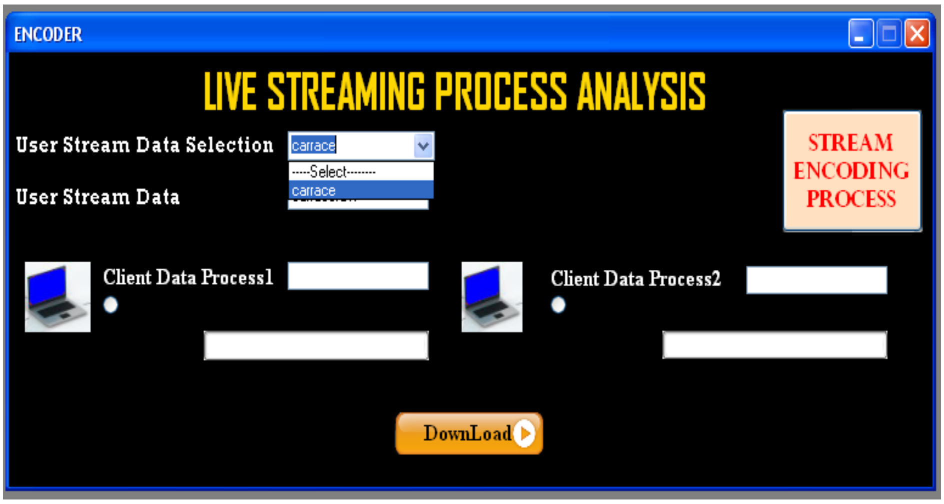


Figure 2 video upload from server to zatto

The amount of available uplink bandwidth at a peer is initially estimated by the peer sending a pair of probe packets to Zattoo's Bandwidth Estimation Server. Once a peer starts forwarding substreams to other peers, it will receive from those peers quality-of-service feedbacks that inform its update of available uplink bandwidth estimate. A peer sends quality-of-service feedback only if the quality of a substream drops below a certain threshold.2 Upon receiving quality feedback from multiple peers, a peer first determines if the identified substreams are arriving in low quality. If so, the low quality of service may not be caused by limit on its own available uplink bandwidth—in which case, it ignores the low quality feedbacks. Otherwise, the peer decrements its estimate of available uplink bandwidth. If the new estimate is below the bandwidth needed to support existing number of virtual circuits, the peer closes a virtual circuit. To reduce the instability introduced into the network, a peer closes first the virtual circuit carrying the smallest number of substreams.

Figure 3 video decoded into xml

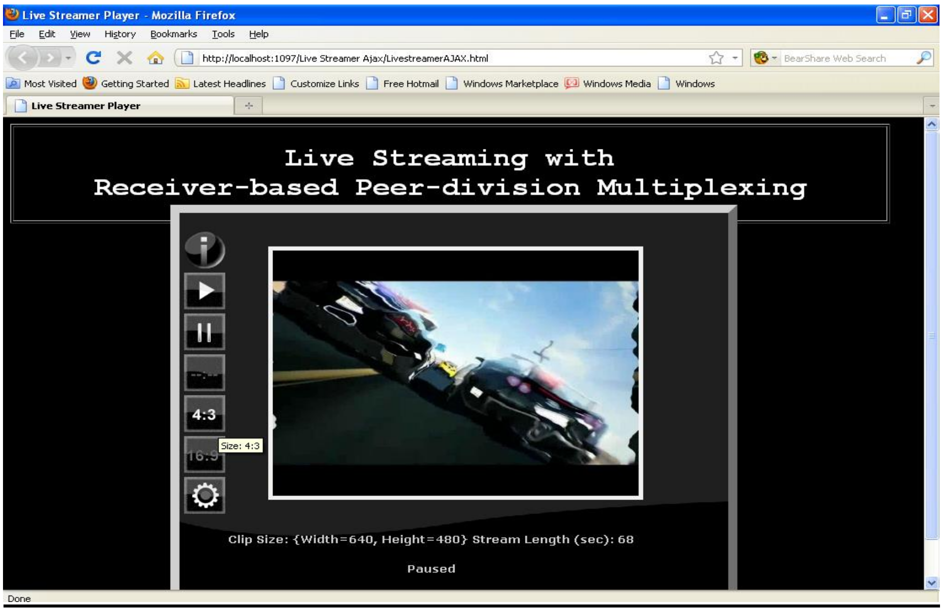


Figure 4 xml file paled as a video

V. CONCLUSION

We have presented a receiver-based, peer-division multiplexing engine to deliver live streaming content on a peer-to-peer network. The same engine can be used to transparently build a hybrid P2P delivery network., we have shown that the resulting network can scale to a large number of users and can take good advantage of available uplink bandwidth at peers. We have also shown that packet retransmission can help improve network stability by isolating packet losses and preventing transient congestion from resulting in PDM reconfigurations. We have further shown that the PDM schemes presented have small enough overhead to make our system competitive to digital satellite TV in terms of channel switch time, stream synchronization, and signal lag. We developed Networking Technology for P2P performance of data stream management systems. The main idea of these Policies is maximize the Data Packet arrival rate to streamer.

REFERENCES

[1] R. auf der Maur, "Die Weiterverbreitung von TV- und Radioprogrammenüber IP-basierte Netze," in Entertainment Law, F. d. Schweiz, Ed.,1st ed. Bern, Switzerland: Stämpfli Verlag, 2006.

[2] "Euro2008," UEFA [Online]. Available: http://www1.uefa.com/.

[3] S. Lin and D. J. Costello Jr., Error Control Coding, 2nd ed. Englewood Cliffs, NJ: Pearson Prentice-Hall, 2004.

[4] S. Xie, B. Li, G. Y. Keung, and X. Zhang, "CoolStreaming: Design, theory, and practice," IEEE Trans. Multimedia, vol. 9, no. 8, pp. 1661–1671, Dec. 2007.

[5] K. Shami et al., "Impacts of peer characteristics on P2PTV networks scalability," in Proc. IEEE INFOCOM, Apr. 2009, pp. 2736–2740.

[6] "Bandwidth test statistics across different countries," Bandwidth- Test.net [Online]. Available: http://www.bandwidth-test. net/stats/country/.

[7] X. Hei, C. Liang, J. Liang, Y. Liu, and K. W. Ross, "Insights into PPLive: A measurement study of a large-scale P2P IPTV system," in Proc. IPTV Workshop, Int. World Wide Web Conf., May 2006.

[8] B. Li et al., "An empirical study of flash crowd dynamics in a P2Pbased live video streaming system," in Proc. IEEE GLOBECOM, 2008, pp. 1–5.

[9] J. Rosenberg et al., "STUN—Simple traversal of User Datagram Protocol (UDP) through network address translators (NATs)," RFC 3489, 1993.

[10] A. Ali, A. Mathur, and H. Zhang, "Measurement of commercial peer-to-peer live video streaming," in Proc. Workshop Recent Adv. Peer-to-Peer Streaming, Aug. 2006.