A Monitor Plane Component for Adaptive Video Streaming

Bjørn J. Villa and Poul E. Heegaard

Department of Telematics,
Norwegian Institute of Science and Technology
{bjorn.villa, poul.heegaard}@item.ntnu.no

Abstract

Video components have become an important part of many popular Internet services, and at an aggregated level they are now a dominant service type in terms of generated traffic volume. Many of these services are provided from content providers utilizing the best effort part of an Internet service. In this paper we provide an introduction to the technical evolutions in this area and also present the capabilities of the more recent enhancements in terms of adaptive streaming services. These services are put into the context of an established QoE optimization framework, and a candidate Monitor Plane (measurement and reporting) component is described. The potential use of this component in order to provide an objective QoE indicator and as input to a home gateway bandwidth broker is proposed.

1. Introduction

Video streaming over the Internet has experienced a tremendous growth the past decade. As an example of such a service YouTube may be considered. This service was established in February 2005. By early 2007 - YouTube alone represented about 20% of all HTTP traffic, or nearly 10% of all traffic on the Internet [4]. A recent forecast report [17] states that by 2012 more than 50% of all the traffic will be video based.

As business models for premium content delivery Over-The-Top (best-effort Internet based) are appearing, methods for assuring the quality levels even in this domain are becoming more important. There is a shift from focusing only on the Quality of Service (QoS) dimension to also investigate the Quality of Experience (QoE) dimension [2]. The latter would represent the actual user experience and is therefore considered as more suited for measuring the overall user satisfaction with a service.

A concept which has been introduced for streaming services with promising results is the use of adaptive streaming [21]. This type of streaming allows the client to dynamically change the video quality of an ongoing stream. This is done in order to adapt to changes in network conditions. However, it has been shown [1] that even this solution has its shortcomings as it does not address the situation where multiple service instances are competing. One way of addressing this problem is to seek the assistance of an external entity which is shared by the competing clients. In a home network environment, this would be the home gateway. In order to be able to do such operations, the home gateway must have a way of obtaining information about active video streams.

The structure of this paper is as follows. Section 2 introduces a framework for QoE optimization; Section 3 describes video streaming technologies; Section 4 describes methods for information collection; Section 5 presents a Monitor Plane component for the MS SilverLight framework; Section 6 presents the experimental implementation; Section 7 presents the potential use of the Monitor Plane reports; Section 8 presents the conclusion and Section 9 outlines future work.

This paper was presented at the NIK-2011 conference. For more information see http://www.nik.no/

2. QoE Optimization Framework

There have been several proposals for autonomic access networks being able to perform optimization tasks on both network and service level [6] [10] [19] [20]. These solutions mainly consist of three types of layers or components each serving different roles. These components are called Knowledge Plane (KP), Monitor Plane (MP) and Action Plane (AP).

The idea is that network components through information gathering in MP should be able to recognize, interpret and classify current traffic patterns. The reasoning process is done by the KP which allows the network to identify potential problems in a service mix. The KP should then be able to use a diverse set of tools to resolve the problem it has identified, and perform an optimization towards a specific objective. The actions chosen by the KP is realized by the AP which applies a change in the network configuration or make some adjustment to selected services [20]. In Figure 1 the placement of KP, AP and MP components re illustrated.

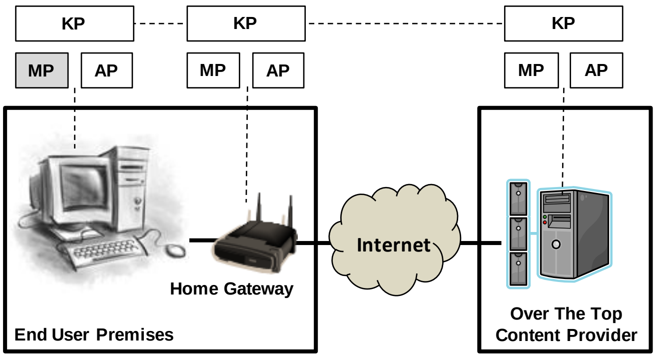

Figure 1 QoE Optimization Framework

The focus in this paper is to describe the Monitor Plane component located in the end user client, and how this can be implemented for emerging video streaming services.

3. Video Streaming Solutions and Technologies

A video streaming service gives a client the opportunity to access video and audio content over the Internet. The streaming content is made available by service providers with the use of streaming servers. The distributed content is either real-time (scheduled) or on-demand. The challenge of providing these services in a best effort network environment is to handle the frequent fluctuations in network characteristics such as available bandwidth and experienced delay/jitter. This section provides an overview of technical approaches and concepts used for video streaming, both historically and today.

3.1 Traditional Streaming Protocols

The Real-Time Transport Protocol [15] provides end-to-end network transport functions for real-time multimedia content (audio, video, etc.). RTP is only a transport protocol and has no means of providing interactivity to the client. In order to achieve such interactivity – giving the users an opportunity to play, pause and fast forward, RTSP [16] has to be used. Hence, RTSP can be considered an extension of the RTP protocol.

Both RTP and RTSP can use RTCP [15] for probing the quality of active sessions. RTCP is based on periodic transmission of control packets. The feedbacks which can be gathered from using RTCP are number of lost packets, jitter and RTT. These parameters can then be used by RTP to perform adaptations to the quality or transmission rate to be used for the data distribution [10].

The reason for RTP/RTSP not being as popular as expected is that it runs on special ports which could be blocked by firewalls and NATs. Another disadvantage is that it requires special servers configured for this technology. In order to overcome these issues, the concept of HTTP based streaming has emerged.

3.2 TCP in Media Streaming

Earlier it was believed that TCP did not have the properties needed in order to provide media streaming because of the contribution to extra delay when retransmission is activated. Another reason for the concern about the streaming capabilities of TCP was the congestion avoidance algorithms used. It was believed that congestion avoidance would contribute to large fluctuations in the video bit rate and viewers QoS/QoE.

However, it has been shown that TCP can be used for video streaming – and it is actually quite commonly used for this purpose. The two main methods of dealing with the retransmission issue are buffering and bit rate adjustment. Offering the video in a set of different bit rates will allow the client to choose a bit rate best suited to its traffic condition. This way the client can handle large bit rate fluctuations, and in combination with a buffer also deal with retransmission delay. Using custom players we can obtain smooth transitions between bit rates. The idea then is that having a smooth transition to a lower bit rate gives a higher QoE than if the video was to stop playing in order to fill the buffer again. These properties in combination with TCP functions such as fairness are why TCP is being used as the transport protocol of choice for video streaming [9].

3.3 Progressive Download

There are several solutions to provide media streaming. One is to transfer the video content from the streaming server to the client using a TCP session at the highest possible bit rate. A buffer is then used at the client-side which starts playing the video content after some byte threshold in the buffer is reached [3]. This will reduce the clients waiting time, compared to downloading the full length video file, but it will not perform well under fluctuating network conditions. This can be prevented using large buffers, but the client waiting time until it starts playing the video content will approach the video download time, which again reduce interactivity. This technique is referred to as progressive download and is used by services such as YouTube.

3.4 Single Rate Adaption

An enhancement made in traditional streaming was to determine the user bandwidth during startup of the streaming session. This allows for choosing a custom video quality for a specific user. Using the bandwidth measurements, a static bit rate is chosen for the client. Some fluctuations in the network conditions can be handled by using this technology in combination with a client side buffer. This approach performs well under rather stable network conditions, but it does not cope well with large fluctuations in network conditions. This technique can be referred to as single rate adaption [8]. A further enhancement of this is the concept of self-adaptive streaming which will be presented in the following section.

3.5 Self Adaptive Streaming

Self-adaptive streaming is a concept developed for handling fluctuations in a network environment. In general the streaming server advertises a set of available streams with distinct bit rates and other relevant parameters to the client, using a manifest file written in a markup language like XML. The streaming server achieves different bit rates by dividing a video file into small segments (some seconds long) with different resolution, hence file size. The client will then monitor network parameters such as display resolution, CPU usage, available bandwidth, etc. Based on these parameters the client will use HTTP GET requests to retrieve segments with the most appropriate bit rate in accordance to the current network conditions [21].

The network conditions are monitored periodically to detect fluctuations and the switching between different bit rates is done seamlessly (cf. Figure 2). It is believed that this is the technique of choice as it does not require any changes to the current Internet architecture. Another advantage is that by using adaptive streaming there is a possibility to constrain the buffer size and therefore reduce the wasted bandwidth if a client decides to stop watching the video half way in.

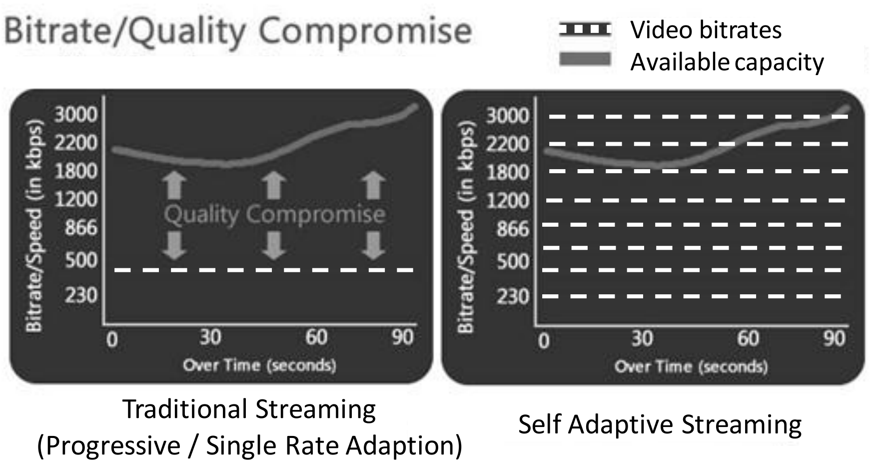

Figure 2 Traditional vs. Adaptive Streaming

This technology can adapt to different users with different connections to the Internet. All the users can view the media without the need for the media itself being scaled for the user with the lowest bandwidth. Hence, a client with high bandwidth can view a video with high quality, while clients with low bandwidth can view videos with lower quality

A negative property of adaptive streaming is that there are no standards for this technology, but there is ongoing standardization work. This has resulted in an IETF draft [13]. Some examples of proprietary solutions using this HTTP streaming are Adobe Dynamic HTTP Streaming, Apple HTTP live streaming and Microsoft Smooth Streaming (SilverLight framework).

3.6 HTML5 Video Streaming

HTML5 is not a web standard yet, but still it is of great interest to the future of video on the web [18]. Most browsers support the video tag introduced in HTML5 which allows for easy addition of video in a web site, without the need for browser-specific third-party plug-ins. The problem with the plugins is that the video players and content were a black box, as seen from the browser. This gives some challenges in terms of applying style sheets or transitions to the video being showed on the web page [5]. In less technical terms, the integration of the video component and the rest of the web page is not optimal.

The question then becomes if HTML5 will be able to provide the wanted functionality that the proprietary plug-ins already do. The usability across platforms is a plus, but to replace the proprietary technologies it must also provide technology that makes it a viable substitute.

4. Methods for Information Collection

In this section we will describe how a Monitor Plane component for streaming services can be implemented in a home network domain. There are several approaches to this, including both active and passive methods. The location of these can also vary, whereas both end systems and intermediate nodes are possible locations.

4.1 Monitoring Traffic

The traffic monitoring approach contains both a traffic capture and an interpretation part. The goal is to make use of the information that is transmitted between the client and server, by an intermediate node. In adaptive streaming services there is usually an initial message sequence where the server advertises its services and bandwidth options. In addition, the client often provides information for authentication. As mentioned in section Y, the former can be done using a manifest file. The latter is often done in a similar matter with XML encoded strings presenting the user information to the server. As long as this information is transmitted as clear text with a well known format, an intermediate node such as the home gateway could capture this information and use it for the purpose of QoS/QoE optimization.

4.2 Altering Streaming Client Code

A tempting alternative to the traffic monitoring approach is to collect all required information directly from the client. In this scenario one would be less vulnerable for changes in message format, and also the clear text requirement would then disappear. The reason for this is that the video streaming clients are downloaded from the service provider every time the user watches a video. This means that users would get the software and successive upgrades without noticing it

4.3 Information from the web browser

One thing the clients have in common is that they are all viewed in a web browser, and most web browsers allow for custom third-party plug-ins or add-ons. By creating an add-on for information collection and reporting purposes we could retrieve information about the resource consumption and serviced running. This method has several imperfections which in our view could disqualify it.

First of all, the number of browsers is quite large. A custom add-on would have to be implemented for each browser and constantly be updated as the browsers often have version upgrades. In addition, video player plugins based on e.g Silverlight or Flash are black boxes to the browser – and therefore information collection from these objects are not possible.

4.4 Reporting Application

Another solution for the active approach is to have a standalone application running on the end system. This application could be written using any programming framework and language. This application would run in the OS directly and therefore give easier access to a lot of OS parameters such as CPU load and available memory. In addition we could get information on what type of interface the end system uses to connect to the home gateway, which is interesting since e.g. wired and wireless Internet connections have different levels of QoS. The network link load could also be collected.

The standalone approach has some advantages in retrieving network related and OS specific information. However, the retrieval of application information would be a significant challenge. Application specific information, like the current quality level used by a Smooth Streaming application, will be difficult to retrieve. The application is run in a black box, on top of a browser. It is not made for access by other applications

5. Monitor Plane Component for Adaptive Streaming

The self adaptive streaming solution chosen for closer study was based on the Microsoft SilverLight framework. The reason for this choice was the use of it by a research project partner (TV2) in their commercial product. A Silverlight based web page allows code to be added for more than stream delivery. For our purpose we added code which observed parameters of the video player and some platform aspects (cf. Table 1). This approach is in line with the concept described in section 4.2. Whenever changes in these parameters were observed – or at regular intervals, a report would be sent.

Parameter	**Description**
event	The type of event triggered in Silverlight client
eventMessage	A message in relation to the event
playState	Current playstate of the Silverlight player
fullScreen	Boolean value indicating if player is full screen
clientId	A Silverlight client globally unique identifier
avgProcessorLoad	Processor load of the client computer
avgProcessLoad	Processor load caused by Silverlight
videoDownloadBitrate	Current rate of streaming
avaliableVideoBitrates	Video bit rates the media is offered in
timeIncrement	Time since application was loaded, in seconds
sourceIpAndPort	IP and port of streaming server, as DNS name

Table 1 Information components in SilverLight video player

In addition to the information available directly from the SilverLight video player, a set of additional parameters can be collected by some additional processing (cf. Table 2). This would typically be a software process on the server side which inspects the incoming HTTP POST messages, extracts the source IP address and perform a lookup toward a whois service in order to obtain the AS number.

Parameter	**Description**
myIP	IP address of the streaming client
ispAS	The AS number for the client IP, identifying ISP

Table 2 Additional client information components

Further on, the Silverlight code uses a set of event handlers to notify if changes have occurred to the player or video stream. When such an event occurs, methods are invoked to retrieve a set of system parameters. These parameters will be retrieved and sent when the video player triggers a PlayStateChange, FullScreenChange, BitRateChange, Initial- or PeriodicEvent. This allows us to monitor what it is currently running and also be notified when it changes.

The combined information from the Silverlight event and system parameters can then be passed as HTTP POST message to an external entity. The external entities of special interest in the context of QoE optimization is the home gateway, and for analysis - a data collection / statistics server.

6. Experimental Implementation

In order to test the proposed concept of having a Monitor Plane component integrated with an adaptive streaming service a prototype implementation has been developed [7]. This prototype has been implemented in a lab environment (cf. Figure 3). The lab setup is supposed to be similar to a typical home environment, with a number of PC's – a home gateway (experimental router) and a broadband access with a limited capacity set by the access bandwidth control node.

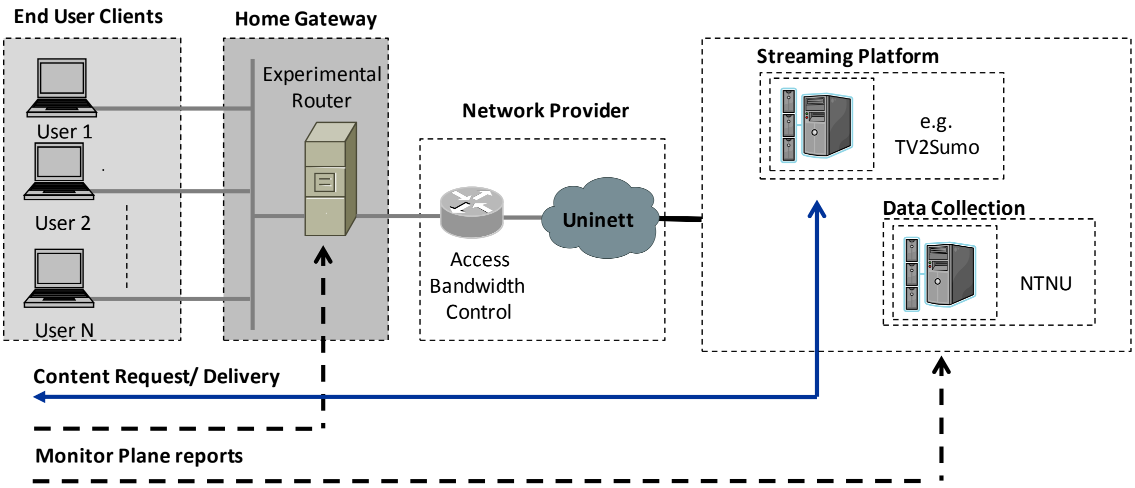

Figure 3 Monitor Plane component testbed

The prototype Monitor Plane Component in the adaptive streaming application used in the testbed is integrated with live TV channels from TV2 Norway, using their content delivery solution for webTV called TV2Sumo. In the testbed the Monitor Plane messages are sent both to an experimental home gateway and to a central statistics server. This has given a good basis for evaluating the feasibility of having such a Monitor Plane component present.

```
Event: InitialInformation with message DONE
from client a2db9c1d-ef45-43ac-ae2c-0ecd61fd3aef.
Average total cpu load: 51%. Silverlight cpu load: 36,6%.
Your playstate is currently: Playing. Fullscreen is False.
You are currently streaming at 2436000, and the bitrates to choose from are:
300000 - 427000 - 608000 - 866000 - 1233000 - 1636000 - 2436000.
Your IP is 129.241.197.94. which belongs to AS224 and you are streaming from:
server video3.smoothhd.com.edgesuite.net:80.
This event occured at time: 1,64 seconds after app was loaded.
```

Figure 4 Monitor Plane report content

A typical Monitor Plane report is illustrated in Figure 4. The use of these reports in respectively the home gateway and statistics server will be described in the following section.

7. Using the Monitor Plane reports

There are several areas where the Monitor Plane reports could be used, but in the context of this research the focus is on how to optimize QoE. In order to perform such optimizations it is necessary to have methods for both analysis and control. The analysis part of objective QoE in our scenario is done in the data collection / statistics server, while the control part is foreseen done in the home gateway by means of a bandwidth broker [12] function.

7.1 Input to objective QoE metric

There is a wide range of metrics which influence how satisfied an end user is with a service such as e.g. video streaming. One such QoE metric which is closely related to network aspects is end user perceived fairness. Research from the social science and psychology domain [11] states that perceived fairness is closely related to what is called 'Social Justice'. In this context a queuing system or any other resource allocation mechanism would be considered as a 'Social System'. It has further been found that users react negatively to any system behavior which gives better service to other user, unless justification is provided.

The end user notion of system discrimination has been suggested by [14] as an important measure of perceived service quality, and more specifically the perceived fairness is stated to be closely related to the discrimination frequency. Applying the concept of discrimination to adaptive streaming, it would be related to situations where end user expectations are not met over time, but also to any negative change in the service quality (e.g. rate level reductions).

Using the Monitor Plane component described in this paper it would be possible for the data collection / statistics server to process the incoming reports per stream, and present a perceived fairness metric based on e.g. the number of rate level reductions per minute. This metric could be considered as an indicator of objective QoE for the service.

7.2 Input to bandwidth broker

Traditional QoS implementations in network elements are based on the classification and differentiation of each IP packet based on relationship to certain priority classes. The concept of a bandwidth broker [12] located in the home gateway could provide significant enhancements in this area. A bandwidth broker could utilize more information about each flow of IP packets and perform bandwidth allocations according to this. The additional information about the available quality levels for adaptive streams would be very valuable for this type of function. In the case where congestion occurs, the home gateway should enforce bandwidth limitations to some of the traffic in order to avoid overall service degradation. Knowing what the levels are for the adaptive services has the potential of making the function more efficient.

The Monitor Plane reports described in this paper could be used as input to this type of function located in the home gateway. The potential gain of doing so has been reported in [20] where simulations have been used to study the effect.

8. Conclusions

The reported work should be considered as one of many potential components in a framework for QoS/QoE optimization. The value of Monitor Plane components as decribed is determined of how well they are applied. The suggested use as input to an objective QoE metric and a bandwidth broker located in the home gateway are those examples which relates most to the specific research area.

The conceptual mode of operation – where end user applications participate in the process of optimization by knowledge sharing is considered as very promising and represents a new approach. In general, it is considered to have great potential of simplifying flow identification and classification, something which in the traditional approach has been very challenging in a dynamic environment. The specific use together with the home gateway represents a novel enhancement to existing Internet based service delivery architectures.

The proposal of using perceived fairness as an indicator of objective QoE is an important step in the direction of being able to measure and influence the end user experience. The understanding and definitions of objective QoE is challenging, but is expected to gain increased attention.

9. Future Work

As future work it is planned to both enhance the capabilities of the Monitor Plane component, and also the way this information is used by different entities. As the QoE optimization framework consists of many components, it is also required to find a way to combine and reason the information obtained from different sources.

The ability to post-process Monitor Plane reports from a high number of live users are considered as very interesting in order to both understand user behavior and also to study the real performance of this service type.

Investigating the correlations between achieved quality levels and changes in quality levels (perceived fairness) up against completed service delivery could give more important knowledge about objective QoE.

10. Acknowledgements

The reported work is done as part of the PhD studies for the first author which is an integrated part of the Road to media-aware user-Dependant self-aDaptive NETWORKS - R2D2 project. This project is funded by The Research Council of Norway.

References

[1] Saamer Akhshabi, Ali C. Begen, and Constantine Dovrolis. An experimental evaluation of rate-adaptation algorithms in adaptive streaming over http. In *ACM Multimedia Systems (MMSys)*, 2011.

[2] E. Areizaga, L. Perez, C. Verikoukis, N. Zorba, E. Jacob, and P. Odling. A road to media-aware user-dependent self-adaptive networks. In *Proc. IEEE Int. Symp. BMSB '09*, pages 1–6, 2009.

[3] A. Begen, T. Akgul, and M. Baugher. Watching video over the web: Part 1: Streaming protocols. *Internet Computing, IEEE*, 15(2):54 –63, march-april 2011.

[4] Xu Cheng, , Jiangchuan Liu, and Cameron Dale. Understanding the characteristics of internet short video sharing: Youtube as a case study. *IEEE TRANSACTIONS ON MULTIMEDIA*, 2009.

[5] François Daoust, Philipp Hoschka, C. Patrikakis, Rui Cruz, Mário Serafim Nunes, and David Osborne. Towards video on the web with html5. October 2010.

[6] Hans Dequeker, Tom Van Caenegem, Kris Struyve, Edith Gilon, Bart De Vleeschauwer, Pieter Simoens, and Wim Van de Meerssche. Advanced features for mm enabled access platform (muse project deliverable). Technical report, Alcatel Bell and IBBT - IBCN, December 2006.

[7] Kristian Haugene and Alexander Jacobsen. Network based qoe optimization for “over the top” services. Master’s thesis, NTNU, 2011.

[8] Charles Krasic. *A framework for quality-adaptive media streaming: encode once - stream anywhere*. PhD thesis, 2004. AAI3119036.

[9] Charles Krasic, Kang Li, and Jonathan Walpole. The case for streaming multimedia with tcp. In *In 8th International Workshop on Interactive Distributed Multimedia Systems (iDMS 2001*, pages 213–218, 2001.

[10] S Latr´e, P Simoens, B De Vleeschauwer, and Van de Meerssche. An autonomic architecture for optimizing qoe in multimedia access networks. *Computer Networks 53*, 10:1587–1602, 2009.

[11] Hanoch Levy, Benjamin Avi-Itzhak, and Davidz Raz. Network performance engineering. chapter Principles of fairness quantification in queueing systems, pages 284–300. Springer-Verlag, Berlin, Heidelberg, 2011.

[12] K. Nichols, V. Jacobson, and L. Zhang. A two-bit differentiated services architecture for the internet. RFC 2638 (Informational), July 1999.

[13] May Pantos. “http live streaming”. ietf draft november 2010. visited: 16.02.2011 <http://tools.ietf.org/html/draft-pantos-http-live-streaming-05>.

[14] Werner Sandmann. Quantitative fairness for assessing perceived service quality in queues. *Journal Operational Research*, pages 1–34, April 2011.

[15] H. Schulzrinne, S. Casner, R. Frederick, and V. Jacobson. RTP: A Transport Protocol for Real-Time Applications. RFC 3550 (Standard), July 2003. Updated by RFCs 5506, 5761, 6051, 6222.

[16] H. Schulzrinne, A. Rao, and R. Lanphier. Real Time Streaming Protocol (RTSP). RFC 2326 (Proposed Standard), April 1998.

[17] Cisco Systems. Cisco visual networking index: Forecast and methodology, 2010-2015. Technical report, June 2011.

[18] Steven J. Vaughan-Nichols. Will html 5 restandardize the web? *IEEE Computer*, 43:13–15, 2010.

[19] Bjørn J. Villa and Poul E. Heegaard. Monitoring and control of qoe in media streams using the click software router. In *Norsk Informatikk Konferanse (NIK2010), Norway. ISBN 978-82-519-2702-4.*, volume 1, pages 24–33, November 2010.

[20] Bjørn J. Villa and Poul E. Heegaard. Towards knowledge-driven qoe optimization in home gateways. *ICNS 2011*, May 2011.

[21] Alex Zambelli. Iis smooth streaming technical overview. http://www.microsoft.com/silverlight/whitepapers/, March 2009.