Multimedia Content Adaptation for QoS Management over Heterogeneous Networks[1]

Narendra Shaha, Ashish Desai & Manish Parashar
Department of Electrical and Computer Engineering
Center for Advanced Information Processing
Rutgers University
Piscataway, NJ, U.S.A

Abstract: *The rapid growth of Internet has resulted in an increased heterogeneity in the network devices and connections used for information access. It has also led to a proliferation of Internet applications with diverse traffic characteristics and needs. Internet Protocol has proved to be a robust and scalable solution to handle the heterogeneity of network devices and connections. However this is achieved by keeping the network core simple with only a best effort service guarantee to the Internet traffic. Several bandwidth and delay sensitive applications can suffer severe performance degradation in the absence of traffic guarantees. In this paper, we present a mechanism for providing Quality of Service (QoS) to multimedia applications in distributed heterogeneous networks. Our approach is to adapt multimedia information content based on current client, system and network state, to meet end-to-end QoS requirements.*

Keywords: *Quality of Service (QoS), Heterogeneous Networks, Multimedia, Content Adaptation, Modality Transformation.*

1 Introduction

Internet Protocol (IP) [1] provides 'best effort' service to applications by routing packets independently (using unique addressing), and seamless delivery over heterogeneous networks (using fragmentation and reassembly). The scalability and robustness of IP is achieved by keeping the network core simple and depending on higher layers to satisfy reliability and other data transfer constraints. Although traditional Internet applications are not affected by this limitation, the performance of applications involving audio and video streaming, with high bandwidth and low latency requirements, can be considerably degraded. QoS over the best effort IP based Internet assumes greater significance with the rise in web based distributed multimedia applications that require some level of quantitative or qualitative determinism in the service provided by the network. Increasing number of web sites are providing rich multimedia content in the form of an integration of text, graphics, audio and video. Many Internet appliances such as handheld computers, personal digital assistants (PDAs) and smart phones are emerging to leverage the potential of the Internet and provide users more ubiquitous access to information than ever before. The lack of Internet infrastructure to accommodate this growing heterogeneity raises challenging research issues for enabling effective information access over the Internet. The issue of QoS can be addressed at different levels of the network protocol stack:

- User level by specifying user perceivable service parameters (qualitatively/quantitatively).
- Application level by adapting the application based on network and system resource availability.
- Transport level by defining traffic models, classification of service disciplines, and resource reservation on a per-flow or flow aggregate basis.
- Network level by processing application data in transit using intelligent routers/switches using application specific information.

Distributed multimedia applications typically operate in heterogeneous environments

[1] The research presented in this paper is based upon work supported by the National Science Foundation under Grant Numbers IIS 9872995 (KDI) ACI 9984357 (CAREERS) awarded to Manish Parashar.

and have to contend with unpredictable resource availability as the client's requirements, end-host capabilities and network state vary significantly. Additionally, the inherent dynamic nature of the resource requirements of these applications makes it very difficult to optimally define the QoS for such applications. As a result, achieving the contracted QoS may not always be feasible. However such applications exhibit a common characteristic of operating satisfactorily in less than ideal situations by allowing tradeoffs between certain service requirements. For example, audio streaming applications are highly sensitive to jitter and hence can compromise response time by buffering data at the receiver before starting playback to smooth out delay variations during playback. This application characteristic can be used to enable operation of such applications with acceptable performance despite network, end host and client dynamics. In this paper we present an adaptive QoS mechanism based on Content Adaptation techniques. The mechanism adapts multimedia information content based on current client, system and network state, to meet end-to-end QoS requirements. It leverages from Real Time Transport Protocol (RTP) [2], to implement a policy driven control scheme for application level adaptations. Various content adaptation techniques for enhancing the performance as perceived by the end-user are presented.

The rest of the paper is organized as follows- section 2 enumerates contemporary content adaptation techniques. Section 3 gives a brief description of the framework for adaptive QoS. Section 4 provides the implementation and operational overview with a discussion on performance issues. Section 5 details the experimental verification. Section 6 presents related work and Section 7 concludes the paper with suggestions for future work.

2 Content Adaptation

In recent years we have seen many Internet appliances such as handheld computers, personal digital assistants (PDAs), set-top boxes, and smart phones emerging as pervasive computing devices. Internet appliances will soon revolutionize the way that information is used and accessed, enabling low-cost and ubiquitous access to Internet content and services. However, before we can take full advantage of these devices, there are many technical problems that need to be resolved. Until recently, most Web content has been designed with desktop computers in mind, and they often contain rich media. As a result, Internet access is still constrained on these devices, and users frequently experience frustration when their devices are unable to handle certain media types or the data takes a long time to download. These shortcomings have prompted the development of new approaches for information delivery.

- *Universal access* [3] addresses technical issues for enabling information access in a heterogeneous network environment, by accommodating the special needs of users and the constraints of client devices and network characteristics.
- *Adaptive content delivery* is a system technology that transforms Web content and delivery schemes according to viewers' heterogeneous and changing conditions to enable universal access.

HTML	Image	Audio	Video
- Text summary - Text-to-audio - Format conversion - Table-to-list - Font size reduction	- Image removal - Size reduction - Color-to-grayscale - Format conversion - Image reduction	- Audio removal - Stereo-to-mono - Audio-to-text encoding - Format conversion	- Video removal - Video-to-image - Video-to-text - Video-to-audio - Format convert - Frame-rate reduction - Resolution reduction - Thumbnail extraction

Table 1: Media Types and Content Adaptations

Many media processing technologies can be used for intelligent information delivery in order to increase content accessibility and improve the user's experience within a heterogeneous network environment. Several existing content adaptation systems apply image-processing techniques to adapt the online images of a Web page according to the characteristics of the client display, such as screen size or color depth. Table 1 lists the possible content adaptations for different media types.

Classification is useful for developing a general decision-making strategy to optimize adaptive content delivery over the Internet. The purpose of classifying content adaptation techniques is to make the exercise modular and extensible. Content adaptation technologies can be classified them into the following categories, based on their applications:

- *Information abstraction* - The goal of information abstraction is to reduce the bandwidth requirement for delivering the content by compressing the data, while preserving the information that has highest value to the user. Examples of information extraction include text summarization, image thumbnail generation, and video highlighting and key-frame extraction.
- *Data prioritization* - Data prioritization aims to distinguish the more important part of the data from the less important part so that different quality of service levels can be provided when delivering the data through the network. Data prioritization can be achieved within a single media type by using special encoding schemes such as layered coding [5][6] and multi-resolution compression.
- *Modality transformation* - Modality transformation is the process of transforming content from one mode to another so that the content can become useful for a particular client device. For instance, most handheld computers cannot handle streaming video data due to hardware and software constraints. In order to make the information contained in the video accessible on these devices, video can be transformed into sets of images, extracted audio or closed caption text.
- *Data Transcoding* – Data transcoding [4] is the process of tailoring the content of web pages to enable universal access using pervasive computing devices. Transcoding can be used to adapt video, images, audio and text content, statically or on-the-fly to match client device specifications and/or user preferences. Examples of data transcoding include video format conversion (such as MPEG-to-QuickTime), audio format conversion (such as WAV-to-MP3) etc.
- *Purpose classification* - A typical Web page contains a lot of information and media objects that are redundant or may not be of interest to a user. For example, an e-commerce web site may have multiple images for linking to the same product site on the top, bottom and side of the page. Purpose classification deals with classifying the purpose of each media object in a Web page, to improve the efficiency of information delivery by either removing redundant objects (assuming the related copyright issues have been properly addressed) or prioritizing them according to importance.

3 Adaptive QoS Framework

The adaptation framework, depicted in Figure 1, uses Real-time Transport Protocol, implemented over UDP [7] and IP-multicast to transport application data. RTP is a light-weight protocol designed to facilitate implementation of application level algorithms for real time optimization of application network traffic. It provides useful information such as payload type identification, payload independent sequencing, payload dependent time stamping, source identification etc.

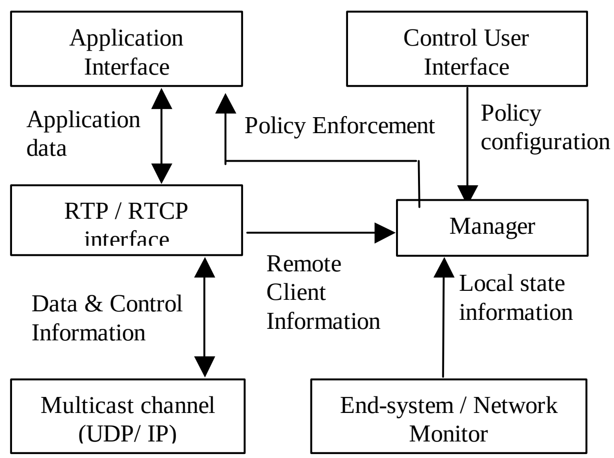

Figure 1: Policy based adaptive QoS framework

Remote client information and statistical transmission/reception information is monitored and exchanged using Real Time Control Protocol (RTCP) [2]. The QoS framework utilizes these features to gain vital information about the traffic characteristics of the application. Local system and network state information is monitored using Simple Network Management Protocol (SNMP) [8]. Based on the local and remote information, the policy driven manager is able to make intelligent decisions for adapting the application's traffic profile. To support deployment in a heterogeneous environment, the policy is configurable independently at each client, using the Control User Interface. The policy is enforced locally by the manager entity using the local state and remote client information. The Application

Interface is a control handle for the Manager to enforce appropriate adaptations on the fly to enable the application to operate optimally under dynamic resource availability conditions.

4 Implementation and Operational Overview

A streaming application is considered with a media server streaming multimedia data on multicast channels. The content is streamed on three distinct multicast channels – media channel, audio channel and text channel. Both the audio and video content is streamed on the media channel. Only the audio content is transmitted over the audio channel. The transcript of the audio channel is sent over the text channel.

Each of these media types has different traffic characteristics and resource requirements. Parameters such as streaming rate, packet loss, delay and jitter define the network resource requirements associated with the media type. The end system requirements for processing the media type are defined by processor and memory utilization. The permissible values for these parameters define the allowable extent of adaptation of the streaming application while maintaining the semantic content of application data.

Clients interested in receiving the content subscribe to one of the three multicast channels depending on their interests and capabilities. The client selects the desired media type and this determines the target service level that the adaptation module at the client end would try to achieve. The network and processing load induced at the client due to this streaming application depends on the media type that the client is subscribed to and the rate at which the media is streamed to the client. Threshold values are defined for network and processor utilization to indicate network congestion and processor overload. The operating states for the client are categorized as Normal, Congested and Overload. A client is considered to be operating in Normal state if the client is able to accept and process the media data while remaining within the threshold limits of processor and network utilization. A client operates in the Congested state if the traffic, due to the media stream that the client is subscribed to, causes network congestion at the client end i.e. the network utilization threshold is exceeded. A client is categorized as operating in the Overload state if the processor utilization or both the processor utilization and the network utilization exceed the threshold.

The media server performs rate-based adaptations independently for the three media types. It maintains an aggregate status of the clients subscribed to each of the multicast channels. Media server adjusts the rate of data transfer, within the acceptable limits, for respective media types to obtain an optimal aggregate performance for the clients subscribed to that media channel.

Each media type has a corresponding network overhead due to its traffic characteristics and a processor overhead to account for processing the media content. Hence traffic characteristics and processing factors are associated with each of the media types to reflect the resource requirements for receiving and processing the media type satisfactorily. Distributed clients are fed with different network and processor load profiles to generate heterogeneous clients. The load profiles are designed to simulate some of the possible resource constraints experienced by users accessing information over the Internet using different network connections and devices. A client is considered to be able to support a particular media type if it can satisfy the network and processing requirements without violating the threshold limits. The client side adaptations are performed to decide which media type, closest to the desired service, can be supported, given the current operating state of the client.

5 Experimental Verification

We verified the performance enhancement achieved by our adaptation mechanism by modeling a simulated heterogeneous device/network test-bed and a streaming application.

5.1 Experimental Setup

The experimental setup consists of the Streaming server, Multimedia client, Streaming application and Load profiles. The adaptation mechanism is integrated into the Streaming server and the

Multimedia client to integrate server and client side adaptations.

- *Streaming Server* - The streaming server establishes separate RTP sessions to handle transmission of each media data. It uses RTCP Receiver Reports for monitoring the state of the clients subscribed to each media type. The server maintains a running average of the packet loss suffered by each client group where the group membership is determined by clients participating in the corresponding RTP session. The server aims to optimize the overall performance of the streaming application for individual client groups by controlling the frame rate to minimize the average packet loss.
- *Streaming Application* - A low-resolution video streaming application is modeled based on the characteristic frame rates and resource requirement for transmitting and processing the audio/video media types. The application typically provides acceptable performance with medium to low resolution video and stereo-to-mono audio quality. The typical bandwidth requirements are tabulated below.

Media	Typical Bandwidth Requirements		
Video	1.5Mbps	512Kbps	128Kbps
Audio	64Kbps	32Kbps	16Kbps

- *Multimedia Client* - Multimedia client is characterized by the device type and Internet connection type available to the user. The device types are characterized by typical processing speed and memory size. Tabulated below are values for devices and Internet connections.

Device type	CPU speed	Memory size
Desktop	800MHz	256MB
Laptop	500MHz	128MB
Portable	200MHz	64MB

Connection type	Bandwidth
Cable modem	4Mbps
DSL	1.5Mbps
ISDN	128Kbps

The multimedia client accepts user preferences to determine the desired Quality of Service. This is the target QoS for the client-side adaptation mechanism. The three media types used are characterized by their requirements in terms of the processing resources, network bandwidth and memory buffer size. The processing and network bandwidth requirements are expressed as percentage of the device processor speed and link bandwidth respectively. Memory buffer is allocated as fraction of device memory capacity and hence the device type determines the number of packets that can be buffered at the client.

- *Load Profiles* - To simulate client side processor and network load, processor and network load profiles are generated at the client. The load profiles are specified as percentage of the maximum processor and network resources, which depend on the device and connection type of the client.

The packets transmitted by the streaming server can be lost due to network congestion and processor overload. Packets lost due to processor overload are packets that are delivered to the client but are discarded by the client as they could not be processed in time. This packet drop also result in a waste of network resources since these packets use the network bandwidth while finally being discarded by client. To capture this loss an application efficiency figure is defined as the fraction of the total packets delivered to the client that are actually processed by the client. It should be noted that this application efficiency figure is not valid when network losses dominate the packet loss figure.

5.2 Results and Discussion

Three heterogeneous clients subscribe to the streaming video application. The first client uses a desktop terminal and cable modem connection. The second client has a laptop device and is connected with a DSL connection. The third client is a portable computer device and is connected via an ISDN link. All three users indicate that they would prefer to subscribe to the video media channel. Similar load profile is generated at the three clients and their performance is measured in both the optimized (i.e. with the adaptation mechanism) and unoptimized case (i.e. without the adaptation mechanism). The packet drop profile and the application efficiency observed at the clients are plotted in Figure 2. The packet drop profile is plotted to logarithmic scale while the application efficiency is plotted to linear scale.

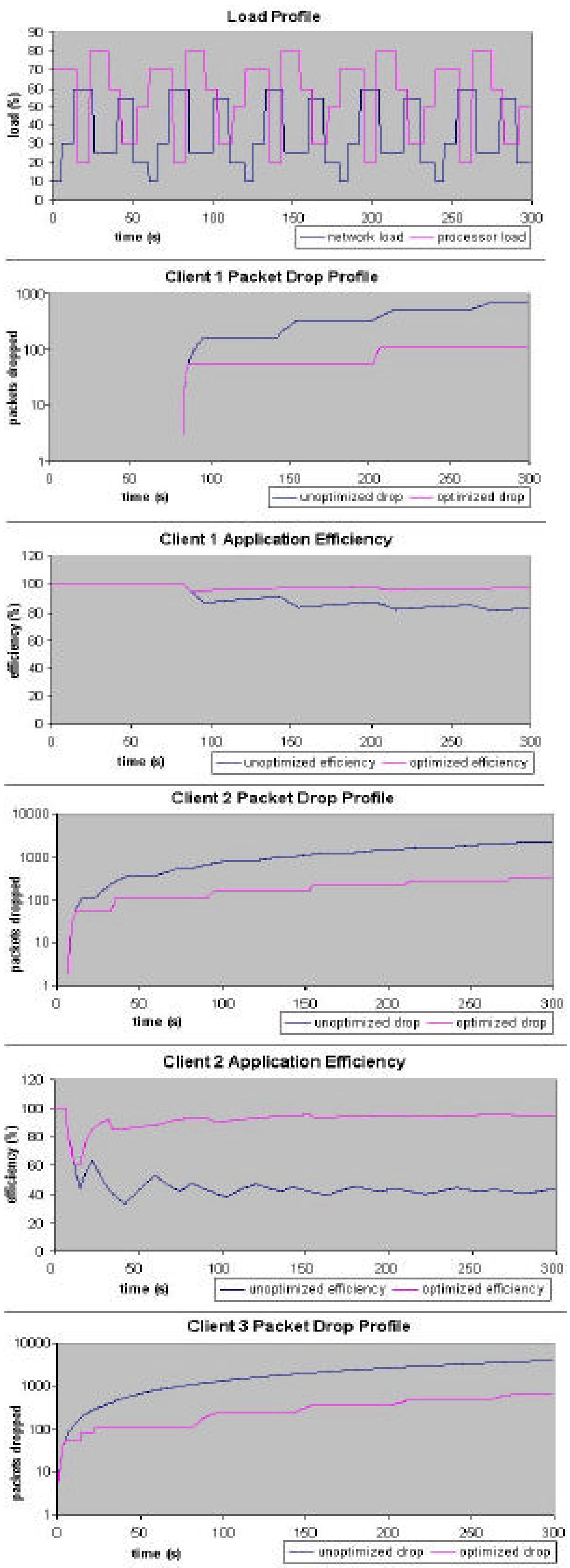

Figure 2: Experimental Results for three heterogeneous clients.

It can be observed that all three clients gain significantly by switching to appropriate media type, thereby resulting in a much lower packet drop rate. Also the efficiency of network usage by the application using the adaptation mechanism is significantly enhanced. It can be observed that the third client being connected via a low bandwidth connection is not able to support video media and suffers excessive network loss thus making the application useless for the client. The adaptation mechanism corrects this by switching the media type so that the client can effectively use the available resources and be able to get relevant information in appropriate media. As noted earlier, application efficiency figure is not valid for client 3 since its packet drop figure is dominated by network losses.

6 Related Work

Numerous companies, academic communities, and standards organizations have recognized the issues for delivering content under heterogeneous clients and network conditions. Examples of commercial products and research prototypes in this area include OnLineAnywhere [9], Spyglass [10], Intel QuickWeb [11], ProxiNet [12], IBM Transcoding proxy [13]. They usually design their systems only for narrow needs. The types of content adaptation they looked into are mostly image-centric transformation. In contrast, our framework is developed to provide a broad range of Web content adaptations for all different types of devices under heterogeneous and changing network conditions.

The W3C and the IETF have existing standards and on-going discussions on facilitating server/proxy decision-making on the mechanisms of content adaptation and content delivery. Most of these protocols are new Web techniques that have yet to gain the recognition of their potential in facilitating Web content delivery. One notable success is the Synchronized Multimedia Integration Language (SMIL) [14][15] which Real Networks [16] has adopted as a key distinguishing feature in their system. SMIL is a markup language that enables the synchronized delivery of multiple video streams, audio streams, and images. It provides conditional constructs to switch tasks (e.g. request different content) based on bandwidth conditions. The Extensible Markup Language (XML) [17] describes the logical representation of data and can facilitate the serving of content to different types of clients under heterogeneous network conditions. The logical representation of data can be converted into an appropriate representation for display using the Extensible Style Sheet Language (XSL) [18].

The HTTP/1.1 content negotiation capability [19] and the Client Capability / Preference Profiles (CC/PP) [20] are mechanisms for the client to convey along with its request its preferred version of content and its user agent information. In HTTP/1.1 content negotiation, a user agent can specify in the HTTP header that, for example, English documents are preferred over French, or that JPEG images are preferred over GIF images. CC/PP specifies client capabilities and user preferences as a collection of URIs and RDF text [20], which is sent by the client along with a HTTP request. The URIs point to an RDF document, which have the details of the clients' capabilities. The RDF text can be used to provide additional details that the referenced RDF documents do not provide. RDF (Resource Description Framework) provides a way to express "metadata" for a Web document. The CC/PP scheme allows proxies and servers to collect information about the client, from the client directly, and to make decisions based on this information for content adaptation and delivery. If CC/PP becomes widely deployed it holds great promise for adaptive content systems.

7 Conclusion

In this paper we have presented the design of an Adaptive QoS mechanism based on Multimedia Content Adaptation. The mechanism adapts multimedia information content based on current client, system and network state, to meet end-to-end QoS requirements. It leverages Real Time Transport Protocol (RTP) [2], to implement a policy driven control scheme for application level adaptations. Preliminary experimentation results show that an adaptation mechanism can be very useful in optimizing the performance of an application with respect to its resource usage and the effective Quality of Service as perceived by the end user. We plan to integrate content adaptation algorithms to test the adaptation mechanism with real streaming applications.

References

[1] J. Postel. Internet Protocol (IP), RFC 791, September 1981.

[2] H. Schulzrinne, S. Casner, R. Frederick, V. Jacobson. RTP: a transport protocol for real-time applications. Report RFC 1889, Internet Engineering Task Force, Jan. 1996.

[3] T. Abdelzaher and N. Bhatti, "Web content adaptation to improve server overload behavior," Proceedings of the 8th Intl. WWW Conference, pp.485-499, May 1999.

[4] http://www.research.ibm.com/networked_data_systems/transcoding/

[5] S. McCanne, M. Vetterli, and V. Jacobson, "Low-complexity video coding for receiver-driven layered multicast," IEEE JSAC, vol. 16, no. 6. Pp. 983-1001, Aug. 1997.

[6] D. Taubman, A. Zakhor, "Multi-rate 3D sub-band coding of video," IEEE Trans.on Image Processing, vol.3,no.5,pp.572-540,Sep. 1994.

[7] J. Postel. User Datagram Protocol (UDP), RFC 768, August 1980.

[8] J. Case, M. Fedor, M. Scho stall, and C. Davin. RFC 1157: Simple network management protocol (SNMP). May 1990.

[9] http://www.onlineanywhere.com

[10] http://www.spyglass.com

[11] http://www.intel.com/quickweb

[12] http://www.proxinet.com

[13] J. Smith, R. Mohan, and C. Li, "Scalable multimedia delivery for pervasive computing," ACM Multimedia, 1999.

[14] L. Bouthillier, "Synchronized multimedia on the Web," Web Techniques Magazine, 3(9), Sep. 1998.

[15] W3C Synchronized Multimedia Integration Language http://www.w3.org/TR/REC-smil

[16] RealNetworks. http://www.real.com

[17] W3C Extensible Markup Language (XML) http://www.w3.org/TR/REC-xml

[18] W3C Extensible Stylesheet Language (XSL) http://www.w3.org/TR/WD-xsl

[19] IETF Working Group onContent Negotiation. http://www.ietf.org/html.charters/conneg-charter.html

[20] W3C Composite Capability/Preference Profiles http://www.w3.org/TR/NOTE-CCPP