A Gateway Architecture for Mobile Multimedia Streaming

Jens Brandt
brandt@ibr.cs.tu-bs.de

Lars Wolf
wolf@ibr.cs.tu-bs.de

Institute of Operating Systems and Computer Networks
Technische Universität Braunschweig
38106 Braunschweig, Germany

ABSTRACT

Transmission of digital audio and video streams is one of the major applications in future wireless networks. Increasing bandwidths and processing power make it possible to receive video streams almost everywhere. But the classes of mobile devices vary from multimedia enabled mobile phones to high-end notebooks. The capabilities of these devices such as display resolution, processing power, memory size or network connection vary also so that we have to deal with a great heterogeneity of client devices. Thus, content providers have to take into account plenty of different requirements when offering video streams to users with mobile devices. In this paper we present a comprehensive gateway architecture for multimedia streaming using IETF standard protocols such as RTSP and RTP. This gateway system provides a video adaptation service for users with mobile devices. A mobile client discovers an appropriate gateway by using a gateway location mechanism. Based on the capabilities of the mobile device as well as on the user's preferences, a suitable media format and adaptation method are chosen at the gateway by which the video is tailored to the requirements of the client.

1. INTRODUCTION

Digital video plays an increasingly important role on the Internet. Advances in audio and video coding make multimedia streaming across a wide range of different networks possible. But the transmission of digital audiovisual data still needs high bandwidths which results in high resource requirements at the consuming client. Additionally, due to increasingly complex video coding standards, the decoding process also becomes more and more complex. MPEG-4 video, for instance, has quite extensive prediction schemes and supports object based coding to increase the compression rate. In contrast, mobile devices usually have limited resources and capabilities and therefore often cannot comply with these requirements of digital video. Thus, today there is a gap between high quality video streams and the capabilities of mobile devices.

EuMob'06, European Symposium on Mobile Media Delivery,
September 20, 2006, Alghero, Italy

On the other hand content providers intend to support as many different devices and platforms as possible. If they want to support all those different existing and future devices, ranging from mobile phones to notebooks, they need to offer media streams in several different quality versions. Obviously this solution can neither meet the requirements of all existing devices nor individual preferences of the users because it is too static. Therefore, we propose to use on-demand video adaptation at the time the stream is transmitted to the client.

Due to the compressed nature digital video cannot be manipulated easily, but needs to be decompressed, adapted and compressed again. Obviously, this procedure is quite costly regarding processing time. However, by the use of transcoding techniques, which operate in the compressed domain, the processing time needed for decompressing and compressing can be saved at the expense of lower flexibility and higher complexity compared to tailoring the uncompressed video. Although transcoding can save processing time, the processing power needed to adapt a stream is still quite high. Therefore, the adaptation should not be performed on the presenting device. Instead, we need some assistance from the network for this purpose, which also reduces the amount of data transmitted over the wireless link to the client. One possibility is to implement an adaptation engine into each media server. This would lead to lower network usage and possibly higher video quality, because potentially additional data, such as meta information could be used for the adaptation process. Another solution is to implement the adaptation engine into gateways which can be placed close to the clients in the access networks. This approach is more flexible than the former one, because the adaptation service is available for all accessible media servers. Especially if the video stream is not exclusively delivered to one client, i. e., by the use of broadcast or multicast delivery, an individual adaptation of the stream is still possible.

From these observations we have derived the following requirements which have to be considered in a future video adaption system:

(1) media streams can be tailored to the requirements of the receiving client defined by the capabilities of the mobile device as well as the user's preferences;

(2) different adaptation techniques are available at the adaptation system to satisfy the demands of different users and devices;

(3) from the user's perspective the adaptation system should be transparent;

(4) the mobility of the users needs to be supported;

(5) the system should be highly interoperable with existing streaming server and client solutions.

In this paper we propose a system which meets all of the aforementioned requirements (1) to (5). Requirements (1) and (2) are achieved by using different transcoding techniques on intermediate gateways. To meet the requirement (3) we use several gateways which can be located automatically. Additionally, the client's requirements can be exchanged during session setup. By cooperation between the gateways we can also achieve criterion (4) because this facilitates a session handoff. To support requirement (5) we use widely-used standard streaming mechanisms and protocols as far as possible.

The remainder of this paper is organized as follows: After discussing related work in section 2 we present information about different video adaptation techniques in section 3. Our system architecture is described in section 4, including a presentation of our gateway location mechanism as well as an initial idea about capability exchange. In section 5 we present some details about our implementation before concluding this paper in section 6.

2. RELATED WORK

One possibility to support adaptive video streaming is the use of layered video encoding. Layered video streams consist of a base layer and one or more enhancement layers. A client can receive as many enhancement layers as needed to meet its quality requirements. In [1] and [2] two different proxies are proposed which are capable of discarding enhancement layers according to the available bandwidth. Due to its nature, layered video is well-suited for multicast transmissions because each layer can be transmitted over a separate multicast channel. When using receiver driven multicast each client tries to join as many channels as its bandwidth permits[3]. Another approach is network driven multicast where all channels are transmitted to all clients but routers can stop forwarding a channel if there is not enough bandwidth available [4, 5].

The main disadvantages of layered video encoding are its limited granularity of adaption and the higher complexity needed for producing and displaying layered video streams. Moreover, a reliable transmission of the base layer has to be assured because in most cases the enhancement layers cannot be decoded without the base layer. Therefore we concentrate on single layer video streams in our work. However, the support of layered video can be added quite easily to our approach, as we use a modular design so that layered video can be supported as an additional media format.

In the literature several systems for adaptation of single layer video can be found and may be classified by the number of nodes involved. Single-node systems consist of a single autonomous node, whereas multi-node systems consist of several cooperative nodes. Multi-node systems can be further subdivided into systems where the adaptation itself is achieved on a single node and systems where this process is accomplished on multiple nodes.

2.1 Single-node Systems

A first video gateway was described by Amir et al. in [6]. They presented an application level gateway which was able to transcode Motion-JPEG to H.261 videos. An overview of available transcoding techniques for media conversions to support mobile users was given in [7]. The authors also indicated that a video gateway will be needed in future networks to support mobile devices. Chi et al. used an existing web caching proxy implementation to build a transcoding system for web objects, including MPEG4 videos [8]. Due to the use of a web caching proxy their system only supports HTTP for data transport. An approach that focuses more directly on multimedia was described in [9], where an active router transcodes multimedia streams during their transmission to the client. In [10] a quite static implementation of a transcoding gateway was suggested. However, the use of non-standard protocols limits the usability of this approach. A more standard-based and flexible proxy which combines adaptation techniques as well as caching capabilities in a single proxy was proposed in [11]. However, this architecture supports only very limited adaptation techniques and cooperation between different proxies is not mentioned.

2.2 Multi-node Systems

A system consisting of multiple nodes was proposed by HP Research in [12]. It consists of transcoding nodes and management nodes. The latter ones are called portal nodes and they are responsible for selecting an appropriate transcoding node by contacting a central service location manager. In contrast to all of the following systems, the transcoding process itself takes place on a single transcoding node. In [13] Hemy et al. proposed to implement filter capabilities on routers, which should be able to adapt video streams by frame dropping if congestion occurs. Mao et al. presented in [14] a system consisting of several adaptation nodes by which a video stream can be adapted. But before any transcoding takes place the complete adaptation path from the source to the client needs to be determined and configured. A more adaptive approach was presented in [15] where a middleware based on mechanisms known from the Java Media Framework [16] was proposed. An approach based on active networking technology was proposed in [17] where active nodes build a multicast tree and some of them can transcode video streams.

In sum, to the best of the authors' knowledge there is no comprehensive adaptation solution for streaming of single layer video supporting all the requirements (1) to (5) mentioned before. In another paper [18] we have already proposed a two level proxy architecture for video streaming in UMTS networks. The focus of that work was a channel adaptation proxy which uses error protection mechanisms to cope with varying wireless channel characteristics. Those channel adaptation mechanisms can also be integrated in the architecture presented in this work, but here we concentrate on the mechanisms needed to tailor video streams to the requirements of the clients.

3. VIDEO ADAPTATION

A promising way to adapt compressed digital video streams is the use of transcoding techniques. This means that the video is manipulated in the compressed domain. Thus the process of decompressing and compressing, which is the most computational intensive part, can be saved. A good overview covering several transcoding techniques for single layer video can be found in [19].

Mainly five different classes of adaptation mechanisms which are useful for mobile devices can be distinguished:

- temporal adaptation: reducing the frame rate
- spatial adaptation: reducing the spatial resolution
- quality adaptation: reducing the quality of each frame
- format adaptation: changing the encoding format
- structural adaptation: changing the contents of the stream

Spatial adaptation is needed to tailor a video stream to the display size of the receiving device, which naturally also reduces the bandwidth of the video. If further or alternative bandwidth reduction is needed, temporal and/or quality adaptation could be used. We conducted interviews with potential users concerning the acceptance of different adaptation methods. The results[1] of this survey show that video streams at frame rates of 12 frames per second (fps) were preferred to video streams with the original frame rate of 24 fps. This shows that reducing the frame rate to half of the commonly used 25 or 30 fps does not lower the acceptance by the users. The positive effect of temporal adaptation compared to quality adaptation arises from the fact that the frames of the adapted stream with a reduced frame rate can be coded with a higher amount of data. For instance, if a specific target data rate needs to be accomplished, mainly two possibilities exist: i) the frame rate is reduced or ii) the quality of each frame is reduced. If, as in i), the frame rate of the original video stream is reduced to 15 or 12 fps, the amount of data available to encode each frame is higher than in ii) where the frame rate is unchanged. Thus, the quality of each frame in case i) is higher than in case ii) and therefore the subjective quality of the whole video stream in case i) is better than if the frame rate remains unchanged. Format adaptation is needed if the requesting client is not able to decode the format of the original video stream. In this case the transcoding process has to change the syntax of the video bit stream, e. g. from MPEG-2 to MPEG-4, or to partially decode those parts of the stream which are not supported by the client, e. g. the client only supports MPEG-4 Simple Profile. Structural adaptation is useful whenever the content of video streams should be adapted, e. g. a summary of a football match.

4. SYSTEM ARCHITECTURE

Our target scenario is depicted in figure 1. We have an access network (dark gray) containing several access points by which mobile clients can connect to the Internet. We assume that the mobile devices have network layer connectivity via the access points, e. g. by the use of mechanisms like mobile IP [20, 21], and seamless handoffs between access points are supported. Media servers are located on the Internet and can be reached by the clients. Additionally, we assume that the network supports multicast communication between clients and gateways as well as between gateways.

[1]The complete results of this survey will be published in the near future.

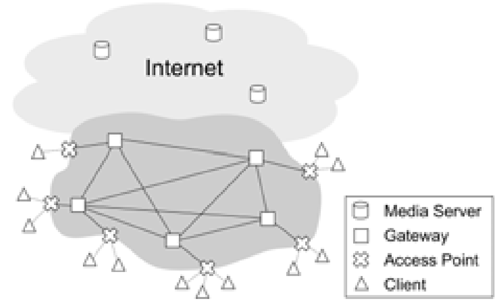

Figure 1: Scenario

We assume that the multimedia gateways are placed in the access network. The delay between gateways and clients has to be quite low in order to react to client movement or variation in communication conditions on the wireless link as quickly as possible. Additionally, this placement is reasonable for the carrier of the access network due to the enhancement of its service provided for its customers. However, a placement in the access network is not absolutely necessary, as long as a certain latency threshold is preserved, which can be assumed in an access network.

For application layer communication we are using the IETF standard protocols RTSP and RTP. RTSP is used for signaling purposes to create and control streaming sessions, whereas RTP is used for data transport. Each client which would like to use the adaptation service needs to use one of the gateways as an RTSP/RTP proxy. Now the complete signaling as well as RTP data communication is sent via a gateway, which adapts the stream if possible.

Several access networks of different providers are also supported by our architecture as long as a handover between those networks as well as communication between gateways located in different networks is possible. In this case we can treat access networks as one single access network.

So far, we have disregarded the questions of gateway location and capability exchange. At the time a client initially connects to the access network it is possible to set an initial proxy address. But this kind of static proxy configuration would not be sufficient to support mobility of the clients. If a client moves to a place where another gateway is available it could be reasonable to migrate the current session, as explained below. Moreover, a static configuration neither takes into account the work load of a proxy nor supports proxies with different adaptation capabilities (e. g. the desired target format is not supported by the nearest gateway). Therefore we propose to use a gateway location mechanism which enables clients to discover their surrounding gateways.

4.1 Gateway Location

For usability of our system we need some easy to use mechanisms for the mobile clients to discover an appropriate gateway. The Discovery of a gateway is similar to the problem of service location in local area networks. This problem is addressed by service discovery mechanisms like those in UPnP, Salutation, Jini or SLP [22]. All of those mechanisms are based on a reactive service discovery paradigm, i. e., a client requests a service whenever it is needed. For static service environments this may be an appropriate solution but for mobile devices a more proactive approach similar to DRIVE (Discovery of Internet Gateways from Vehicles) [23] is needed. Therefore, we propose a gateway location mechanism which uses reactive service requests as well as proactive service announcements. This leads to a hybrid approach where clients can send service requests whenever they need to discover new gateways, and gateways periodically send service announcements.

Both, service announcements and service requests, are sent to well-defined multicast addresses. Due to the fact that the usability of a gateway is topologically bound, we limit the range of service announcements by using a low time to live (TTL) for these messages. The appropriate value of the TTL depends on the number of available gateways in the access network. Now every time a client reaches the domain of a gateway, which is defined by the TTL of the announcement messages, it will notice its existence and can check its suitability. If, for instance, the conditions of the communication channel between client and gateway such as delay, jitter or bandwidth decline, it may be worthwhile to transfer the session to another gateway. However, the suitability of a gateway is not only limited by the conditions of the communication channel but also by its current load and available transcoding mechanisms.

4.2 Capability Exchange

Existing systems mainly focus on the adaptation of video streams to the current state of the communication channel. However, this strategy is only sufficient if the communication channel is the bottleneck of the transmission. Problems such as size mismatch between display and stream resolution or unsupported media formats are not addressed by this strategy. Therefore, we propose to adapt the video streams not only to the current state of the communication channel, but mainly to the capabilities of the client device as well as to the user's preferences. In order to increase the transparency of our system for the users themselves, we do not want to bother the users with decisions about different coding formats or encoding parameters. Thus, we moved this decision from the user to the gateway. For this purpose the gateways need information about the client device as well as about the user's preferences. RTSP, as defined in RFC 2326, does not support any kind of capability exchange regarding the playback capabilities of the client. Therefore, we decided to use the possibility of RTSP to exchange general parameters for this purpose. RTSP defines two methods called `GET/SET_PARAMETER` which can be used to exchange any user-defined data between both parties of an RTSP session. Currently we are working on a sufficient scheme for the description of client capabilities and user's preferences.

5. IMPLEMENTATION

At the moment we use our own RTSP proxy, implemented in C++, which follows our paradigm of separated control and data paths as presented in [24]. This proxy is able to load different transcoding libraries at runtime according to the requirements of the client. The client is currently identified by its network address. For data transport within our proxy implementation we use small data processing units called StreamHandlers (SH). Those SH are connected by a control unit and build a data path through which the video stream is pipelined. Each SH can process and manipulate the video stream. By the use of a special SH the proxy is able to load a so called subgraph into the data path. Thus, different transcoding modules can be loaded as a subgraph and then tailor the stream to the requirements of the client. If several users are watching the same video stream, our RTSP proxy can use a single control as well as a single data path between the server and the gateway and deliver the stream to each client. Instead of establishing a separate session between the server and the gateway for each client, the data packets are duplicated at the proxy and sent to the clients. Although all clients receive the same contents, they do not necessarily get the same data stream, because the proxy can tailor the stream individually for each client.

6. CONCLUSION & FUTURE WORK

In this paper we presented the architecture of a comprehensive multimedia gateway system which offers adaptation services to users with mobile devices. Due to the increasing heterogeneity of wireless devices on the one hand and increasing quality of multimedia streams on the other hand, video adaptation will be very important in future wireless networks. In contrast to current existing static solutions, i. e., providing video streams in various media formats and quality levels, on-demand video adaptation can meet the capabilities of devices and users' preferences more exactly. In the beginning of this paper we presented five requirements which have to be considered for future video adaptation systems. As shown in section 2 no existing system can currently be found in the literature which meets all aforementioned requirements.

Currently we are working on a suitable scheme for capability exchange between clients and gateways, which should also include some default profiles. To support mobility of the devices we need to develop a mechanism for session transfer from one gateway to another. If, for instance, the connection between a client and its current gateway becomes unstable, a migration of the running session to another gateway can be considered. Therefore, some collaboration between neighboring gateways is additionally needed. For the reduction of network usage we are also working on distributed caching schemes, which are able to deal with different quality versions of video streams.

7. REFERENCES

[1] R. Rejaie and J. Kangasharju, "Mocha: A quality adaptive multimedia proxy cache for internet streaming," in *International Workshop on Network and Operating Systems Support for Digital Audio and Video*, 2001.

[2] Q. Zhang, Z. Xiang, W. Zhu, and L. Gao, "Cost-based cache replacement and server selection for multimedia proxy across wireless internet," in *IEEE Transactions on Multimedia*, vol. 6, 2004, pp. 587 – 598.

[3] S. R. McCanne, V. Jacobson, and M. Vetterli, "Receiver-driven layered multicast," in *Applications, technologies, architectures, and protocols for computer communications*. ACM Press, Aug. 1996, pp. 117–130.

[4] B. Vickers, C. Albuquerque, and T. Suda, "Source-adaptive multi-layered multicast algorithms for real-time video distribution," *IEEE/ACM Transactions on Networking*, vol. 8, no. 6, pp. 720–733, Dec. 2000.

[5] T. Shanableh and M. Ghanabari, "Multilayer transcoding with format portability for multicasting of single-layered video," *IEEE Transactions on Multimedia*, vol. 7, no. 1, pp. 1 – 15, Feb. 2005.

[6] E. Amir, S. McCanne, and H. Zhang, "An application level video gateway," in *ACM Multimedia '95, San Francisco, CA*, 1995.

[7] A. Vetro and H. Sun, "Media conversion to support mobile users," in *IEEE Canadian Conference on Electronic and Computer Engineering*, 2001.

[8] C.-H. Chi and Y. Cao, "Progressive proxy-based multimedia transcoding system with maximum data reuse," in *Proceedings of the tenth ACM international conference on Multimedia*, 2002, pp. 425–426.

[9] J. Guo, F. Chen, L. Bhuyan, and R. Kumar, "A cluster-based active router architecture supporting video/audio stream transcoding service," in *International Parallel and Distributed Processing Symposium*, 2003.

[10] Z. Lei and N. D. Georganas, "Video transcoding gateway for wireless video access," in *Canadian Conference on Electrical and Computer Engineering*, vol. 3, 2003, pp. 1775 –1778.

[11] P. Schojer, L. Bszrmenyi, H. Hellwagner, B. Penz, and S. Podlipnig, "Architecture of a quality based intelligent proxy (QBIX) for MPEG-4 videos," in *Proceedings of the twelfth international conference on World Wide Web*. ACM Press, 2003, pp. 394–402.

[12] S. Roy, M. Covell, J. Ankcornand, S. Wee, and T. Yoshimura, "A system architecture for managing mobile streaming media services," in *23rd International Conference on Distributed Computing Systems Workshops*, May 2003, pp. 408 – 413.

[13] M. Hemy, U. Hengartner, P. Steenkiste, and T. Gross, "MPEG System Streams in Best-Effort Networks," in *Packet Video*, 1999.

[14] Z. M. Mao, H.-S. W. So, and B. Kang, "Network support for mobile multimedia using a self-adaptive distributed proxy," in *11th International workshop on on Network and Operating Systems support for digital audio and video*, 2001, pp. 107–116.

[15] K. Hashimoto and Y. Shibata, "Extended video stream by media transcoding functions," in *Proceedings of the 24th International Conference on Distributed Computing Systems Workshops*, 2004.

[16] Sun Microsystems, Inc., "Java media framework," http://java.sun.com/products/java-media/jmf/index.jsp.

[17] B. Duysburgh, T. Lambrecht, F. DeTurck, B. Dhoedt, and P. Demeester, "An active networking based service for media transcoding in multicast sessions," *IEEE Transactions on Systems, Man and Cybernetics*, vol. 34, no. 1, pp. 19–31, 2004.

[18] M. Dick, J. Brandt, V. Kahmann, and L. Wolf, "Adaptive transcoding proxy architecture for video streaming in mobile networks," in *Proceedings of the IEEE International Conference on Image Processing (ICIP 2005)*, vol. 3, Genova, Italy, Sept. 2005, pp. 700–703.

[19] J. Xin, C.-W. Lin, and M.-T. Sun, "Digital video transcoding," *Proceedings of the IEEE*, vol. 93, no. 1, pp. 84–97, 2005.

[20] C. Perkins, "IP Mobility Support for IPv4," RFC 3344, Internet Engineering Task Force (IETF), August 2002.

[21] D. Johnson, C. Perkins, and J. Arkko, "Mobility Support in IPv6," RFC 3775, Internet Engineering Task Force (IETF), June 2004.

[22] I. Richard, G.G., "Service advertisement and discovery: enabling universal device cooperation," *IEEE Internet Computing*, Sept.-Oct. 2000.

[23] M. Bechler, O. Storz, W. Franz, and L. Wolf, "Efficient discovery of internet gateways in future vehicular communication systems," in *IEEE Semiannual Vehicular Technology Conference, Jeju, Korea*, Apr. 2003.

[24] J. Brandt, V. Kahmann, and L. Wolf, "A flexible reflector for media streams," in *KIVS, Kurzbeiträge und Workshop*, ser. Lecture Notes in Informatics, vol. PI-61, Mar. 2005, pp. 41–48.