Adaptation and Synchronization in Bandwidth-Constrained Internet Video and Audio

See-Mong Tan Roy H. Campbell Zhigang Chen Willy S. Liao
David K. Raila Fabio Kon Miguel Valdez Dong Xie

Department of Computer Science
University of Illinois at Urbana-Champaign
1304 W. Springfield
Urbana, IL 61801

Vosaic LLC
1242 N. Lakeshore Drive
Chicago, IL 60641

{*stan,roy,zchen,liao,raila,f-kon,m-valdez*}@cs.uiuc.edu, dongx@ifi.uio.no

Abstract

Systems for the real-time transmission of video and audio on low-bandwidth lines must account for the limited bandwidth available, the bandwidth variance, packet loss rates, as well as CPU load. Vosaic, short for Video Mosaic, integrates real-time continuous media within standard Web pages, and handles the issues of transmitting continuous media with the Video Datagram Protocol (VDP). VDP dynamically adapts the media stream to the available network and CPU bandwidth, thus improving video performance. Built as an object-oriented framework, Vosaic permits the easy addition and customization of the basic VDP protocol for different media types. This family of protocols currently support two different standard video formats, as well as four audio formats.

1 Introduction

Early systems for the delivery of real-time video and audio on the Internet, such as Berkeley's CM Player[12], required high-end workstations and high-speed Ethernet or T1 data rates. These systems concentrated on transmitting high quality, and consequently high bandwidth, video and audio over the Internet, or over specialized high bandwidth networks. Even later multimedia tools, such as the "low-bandwidth" NV and VAT for the MBONE[3, 7], required at least ISDN rates for proper functioning.

Providers of real-time video and audio have only recently appeared in commercial form on the Internet. The rapid growth of the World Wide Web has accelerated the deployment of real-time multimedia solutions for the low-bandwidth environments under which the average home user of the Web operates. A major goal of the handful of new efforts has been to efficiently transmit real-time video and audio within the constraints imposed by a 28.8 Kbit/s Plain Old Telephone Service (POTS) connection. In addition, the available bandwidth is also *variable* and not constant. Users of major Internet service providers may also experience high packet loss rates during peak service periods, due to overwhelming demand. Real-time video and audio systems must compensate and account for these factors in order to function adequately in such an environment.

This paper summarizes our approach to providing a video-on-demand service integrated into standard Web pages. The system is called Vosaic, and includes client, server, server management and authoring software. Besides straightforward video and audio, the system also supports novel and useful innovations, such as hyperlinks which accompany the real-time video, allowing one to click and follow a hyperlink on an object within a video. The system also permits random access and easy reuse of all or parts of any movie and sound segment.

A popular approach to video over POTS lines is the use of proprietary video compression algorithms, eg., wavelets[11]. These algorithms are competitive with Fourier transforms and may provide very high compression ratios. In contrast, we developed our system to transmit the most widely available compression formats. For example, we implement support for the *standard* compression formats MPEG[6] and H.263[13]. An adaptable continuous media transmission protocol underlies our approach. Our approach allows us to leverage off the large body of established software tools and hardware encoders and decoders already on the market, while providing performance comparable to other systems that use proprietary encoding techniques.

2 A Short History

Vosaic started as a research project in the Department of Computer Science at the University of Illinois. The first implementation added video to the source base for Mosaic, thus the name Vosaic is a concatenation of *V*ideo M*osaic*. The browser we built incorporates real time video and audio into standard hypertext pages and which are displayed in place. Video and audio transfers occur in real time; there is no file retrieval latency. The video and audio result in compelling Web pages.

However, we recognized that TCP[10] is unsuitable for transmitting real-time audio and video. The reasons are:

- **TCP imposes its own flow control and windowing schemes on the data stream.** These mechanisms effectively destroy the temporal relations shared between video frames and audio packets. Lost packets, especially on noisy or highly loaded networks, trigger backoffs and cause inappropriate pauses (stop-and-start behavior) in multimedia playback.
- **Reliable message delivery is not required for video and audio.** Video and audio streams tolerate frame losses. Losses are seldom fatal although detrimental to picture and sound quality. TCP retransmission causes further jitter and skew internally between frames and externally between associated video and audio streams.

We thus invented the Video Datagram Protocol (VDP), a protocol specialized for carrying video on the highly variable Internet. Our experiments indicate that the protocol successfully handles real time Internet video by minimizing jitter and incorporating dynamic adaptation to the client CPU load and network congestion. Further, the protocol imposes minimal storage requirements on the client and these requirements are independent of the length of the video transmission. In a test from Illinois, USA to Oslo, Norway, a forty-four-fold increase in received video frame rate was achieved (0.2 frames per second (fps) to 9 fps) with the use of VDP in lieu of TCP, with a commensurate improvement in observed video quality.[5]

In the first and second quarter of 1996, Vosaic became a commercial entity with the University of Illinois participating in a joint venture with Digital Video Communications. We reimplemented the core modules in Vosaic in an object-oriented manner using lessons learnt in the building of the *Choices* and μ*Choices*[1, 2] object-oriented operating systems. The object-oriented framework permits the easy addition and the incremental customization of protocols and decoders. We expanded the number of video and audio formats we support. Vosaic now supports:

- MPEG1 and MPEG2 video
- H.263 video
- MPEG audio
- GSM audio (13 Kbit/s)
- Half-rate GSM audio (8.3 Kbit/s)
- Half-rate G.723 audio (3.3 Kbit/s)

We continued to refine and improve VDP to support the transmission of real-time continuous media over low-bandwidth POTS line environments. VDP is, in effect, a family of associated protocols customized for each different media type within the Vosaic framework.

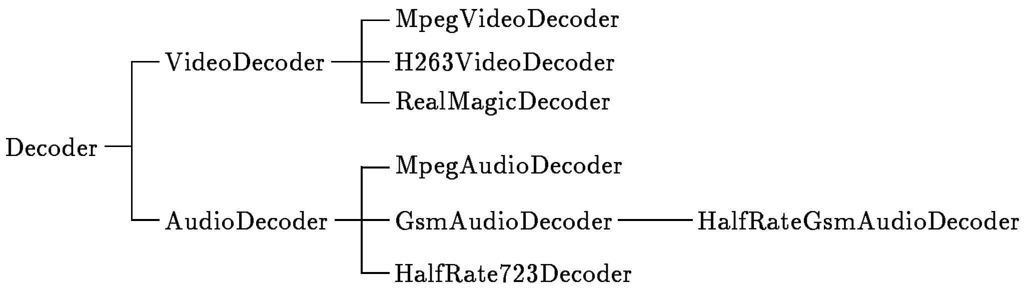
At the present writing, the Vosaic client operates as a Netscape plugin for Windows 95 and Windows NT, MacOS 7.5.3, as well as Sun Solaris. We expect the SGI plugin to be available shortly. An ActiveX Control version of the client also operates under Microsoft's Internet Explorer, and can be embedded as an OLE control into other applications on any 32-bit Windows platform. The Vosaic server operates on Windows 95 and Windows NT, MacOS, as well as most Unixes.

3 Vosaic Software Architecture

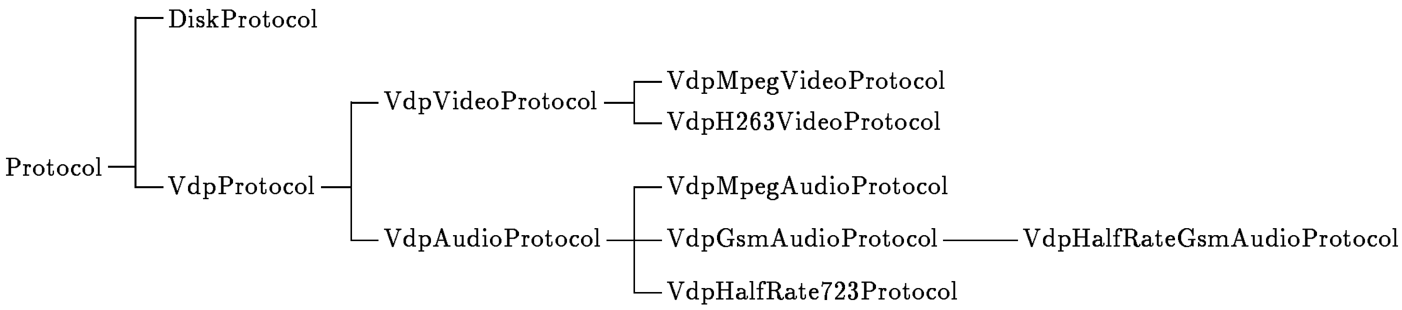
Both the server and client in Vosaic are implemented in C++[9] as object-oriented frameworks[8] of classes.

3.1 Framework

The basic classes in Vosaic's client framework include the abstract `Protocol` and `Decoder` classes. Instances of the `Decoder` class supports the decoding of the different media types. The `RealMagicDecoder` class supports hardware decoding of MPEG streams on Intel/Windows platforms through the Sigma Designs Real Magic MPEG decoder hardware.



The `Protocol` class encapsulates the way data is carried from the source to the destination decoder. The `Disk` class supports reading continuous media data off a disk file using calls to the standard I/O



library. This is useful for testing the framework with a local file. The subclasses of `VdpProtocol` implement behavior particular to the media type (concrete class of `Decoder`) that it supports. For example, the `VdpHalfRateGsmAudioProtocol` expects audio samples at 32 millisecond intervals instead of 20 millisecond intervals for `VdpGsmAudioProtocol`.

A complementary framework exists on the server for the transmission of video and audio to the client.

3.2 Server Parse Files

The server relies on *parse files* generated by parse tools in order to allow random access into the middle of video and audio streams without requiring a sequential search. Parse files provide indices into the media file. Clients access video segments by specifying time indices into the stream. These are sent to the server, which then uses the parse files to locate the correct location in the media stream without requiring a sequentual seek from the beginning of the file.

4 Protocol Issues

The client and server in Vosaic communicate through a pair of Internet Protocol (IP) connections:

- A reliable channel (through TCP) for client commands, such as *Play*, *Fast Forward*, *Rewind* and *Stop*.
- An unreliable channel (through UDP) for data and feedback.

Vosaic employs a classic closed-loop feedback system to monitor network and client CPU congestion. Feedback returned to the server allows the server to mediate the bandwidth of the continuous media stream at the source.

4.1 Separate Transmission and Adaptation

Vosaic separates audio and video into different files before transmission. The files are broken up into chunks and transmitted as independent streams. The separation of audio from video is imperative for proper adaptive behavior. Storing the continuous media stream as an interleaved audio and video file simplifies synchronization during playback, but renders adaptation very problematic.

Vosaic adapts by modifying the frame rate of the video. When the available bandwidth falls below the level required to support the current frame rate, Vosaic *thins* the video stream by dropping video frames at the server. Audio is *never* dropped. Dropping audio would cause undesirable glitches in the playback of the media stream. In contrast, human visual perception does not find a decrease in playback frame rate excessively annoying.

4.2 Smoothing and Loss

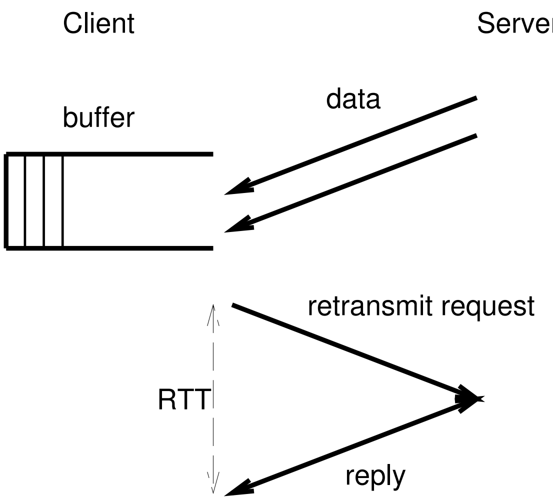


Figure 1: Retransmission and size of buffer queue.

As described in [5], VDP maintains a buffer of size $\rho \times \mathrm{RTT}$, where RTT is the round trip time between the server and the client, and $\rho \geq 1$. The buffer smoothes network jitter and allows the retransmission of key frames which are lost.

Packets are time stamped and have sequence numbers. Lost frames are detected at the tail of the queue. A retransmission request is sent to the server if the client VDP decides that a frame has been lost (a packet with a sequence number more than what was expected arrives). We require ρ to be greater than or equal to 1 in order that the lost frame have enough time to arrive before its slot arrives at the head of the queue. The exact value of ρ is an engineering decision.

4.3 Head Synchronization

As VDP performs thinning of the video stream at the source, video frames do not arrive at the same rate as audio packets. In Vosaic, the synchronization of the video and audio streams is performed by the *video synchronizing to the audio*. The framework ensures that a video image is presented at the time when the appropriate audio segment is playing.

4.4 Tail Adaptation

In order to adapt the video stream, VDP must detect when the media streams' bandwidth exceeds the available bandwidth of the network. Systems such as Cen, Pu, Staehli, Cowan and Walpole's Distributed MPEG Player[4] adapt when the display frame rate falls below some level. In contrast, we perform adaptation at the tail of the buffer queue instead of measuring display rate. This allows us to respond to changing conditions *before* these effects propagate to the head of the queue. The VDP algorithm constantly measures the inter-frame arrival rate, as well as packet loss rates, for both audio and video. Increasing inter-arrival times indicate queue build-up within the network and require corrective feedback.

5 Conclusion

This paper summarizes the Vosaic philosophy (the use of standard compression formats), architecture (object-oriented frameworks) and protocol issues (head synchronization and tail adaptation) in our system for the support of low bandwidth Internet video.

Video for low bandwidth, POTS line Internet environments face different problems than that for high bandwidth environments. In the latter, bandwidth constraints are relatively unimportant and the problems devolve to proper synchronization between streams and realizing the workstation I/O bandwidth required to support high quality video. In low bandwidth environments, innovative protocol design has allowed us to transmit video encoded in MPEG on POTS lines. We believe that our work has the potential for incorporation as a standard for adaptive multimedia on the World Wide Web.

References

[1] Roy H. Campbell and Nayeem Islam. " Choices: A Parallel Object-Oriented Operating System". In Gul Agha, Peter Wegner, and Akinori Yonezawa, editors, *Research Directions in Concurrent Object-Oriented Programming*. MIT Press, 1993.

[2] Roy H. Campbell and See-Mong Tan. μChoices: An Object-Oriented Multimedia Operating System. In *Fifth Workshop on Hot Topics in Operating Systems*, Orcas Island, Washington, May 1995. IEEE Computer Society.

[3] S. Casner and S. Deering. First IETF Internet Audiocast. *ACM Communications Review*, 22, July 1992.

[4] Shanwei Cen, Calton Pu, Richard Staehli, Crispin Cowan, and Jonathan Walpole. Demonstrating the Effect of Software Feedback on a Distributed Real-Time MPEG Video. In *ACM 1995 Multimedia Conference*, San Francisco, California, November 1995.

[5] Zhigang Chen, See-Mong Tan, Roy H. Campbell, and Yongcheng Li. Real Time Video and Audio in the World Wide Web. *World Wide Web Journal*, 1, December 1995.

[6] D. Le Gall. MPEG: A Video Compression Standard for Multimedia Applications. *Communications of the ACM*, 34(4):46–58, April 1991.

[7] S. Deering and D. Cheriton. Multicast routing in datagram internetworks and extended LANS. *ACM Transactions on Computer Systems*, pages 85–110, May 1990.

[8] L. Peter Deutsch. Design Reuse and Frameworks in the Smalltalk-80 Programming System. In Ted J. Biggerstaff and Alan J. Perlis, editors, *Software Reusability*, volume II, pages 55–71. ACM Press, 1989.

[9] Margaret A. Ellis and Bjarne Stroustrup. *The Annotated C++ Reference Manual*. Addison-Wesley, 1990.

[10] J. Postel. Transmission control protocol. Technical Report RFC 793, DARPA, September 1980.

[11] Olivier Rioul and Martin Vetterli. Wavelets and signal processing. *IEEE Signal Processing Magazine*, pages 14–38, October 1991. VIdeo Compression.

[12] L. A. Rowe and B. C. Smith. A continuous media player. In *Proceedings of the 3rd International Workshop on Network and Operating System Support for Digital Audio And Video*, San Diego, CA, November 1992.

[13] International Telecommunications Union. Video coding for low bit rate communication. ITU Recommendation ITU-T H.263, April 1996.