Design and implementation of IPv6 multicast based High-quality Videoconference Tool (HVCT) *

Taewan You, Hosik Cho, Yanghee Choi
School of Computer Science & Engineering
Seoul National University
Seoul, Korea
{twyou, hscho, yhchoi}@mmlab.snu.ac.kr

Minkyo In, Seungyun Lee, Hyoungjun Kim
Electronics and Telecommunications
Research Institute
Daejeon, Korea
{mkin, sylee, hjkim}@etri.re.kr

ABSTRACT
Recently, an Internet protocol version 6 (IPv6) that is next generation IP has been researched and standardized. But IPv6 is not prevalent yet, merely being adopted in research networks. With such a background, we developed high-quality multimedia conferencing application based on IPv6 multicast for activation of IPv6. In this paper, we describe the software architecture, multicast protocol that can be many to many conferencing, RTP (Real-Time Transport Protocol) and RTCP (RTP control Packet) library that was implemented in C++ to support real-time data transmission, and DirectShow Technology that is provided by Microsoft DirectX that is an advanced suite of multimedia application programming interfaces (API) built into Microsoft Windows operating system. And we present the experimental result that measures delay of a respective module and bandwidth of these applications.

Keywords
Videoconference tool, Moving Picture Experts Group standards 4 (MPEG 4), MPEG-1 layer 3 (MP3), high-quality multimedia transmission

1. INTRODUCTION
Shortage of IPv4 address pool motivates the birth of IPv6 whose addresses are 128 bits long. Recently a much research and deployment about IPv6 increase investments and efforts in the global countries. But IPv6 is not prevalent yet, merely being adopted in research networks like 6BONE. Also applications, which can use IPv6 networks, are not enough to activate [1].

On native IPv6 networks (networks composed of IPv6-supporting hosts only), most of works have done with unicast feature only. Recently, however, a number of research institutes involve in porting legacy IPv4 MBONE applications into IPv6 networks, and it is expected that researches based on IPv6 networks will be popular such as VIC (videoconferencing tool) and RAT (robust audio tool) are representative developed by UCL [2].

With such a background, we proposed that a new application, which works on native IPv6 networks and supports high-quality multimedia transmission, should be available. Specifically, we designed and implemented a high-quality videoconference tool using MPEG-4 (Moving Picture Experts Group standards 4) video codec and MP3 (MPEG-1 layer 3) audio codec. This application was implemented a library, which was made by C++ that is programming language to use RTP and RTCP protocol to support real-time data transmission and multicast protocol [3]. And we compressed video and audio using MPEG4 codec, which supports low bitrate video coding which can be optimized for the bandwidth constraints of wireless, Internet, and PSTN applications and MP3 codec, which provides CD (compact disk) quality audio each to offer high-quality multimedia conferencing environment and bandwidth fewer than 1 Mbps per user. Also this application grab the multimedia data from hardware device such as soundcard and graphiccard in personal computer.

Therefore, in this paper, we examine the applicability of MPEG4 and MP3 codec to real-time video communications, presenting our high-quality videoconference tool implementation and its experimental results. In Section 2 introduces software design. Then in Section 3 we present the system architectures. Message communication scenario by RTP and RTCP module will be given in Section 4. Section 5 shows experimental results, and finally we conclude this paper in Section 6.

2. Software Design
We implemented high-quality multimedia conferencing system and this system based on GUI (Graphic User Interface) is the application program for Windows and is made to operate at IPv6 network environment [4]. Development environments were Microsoft Visual C++ and DirectX SDK for Windows 2000 and this system need conference camera which supports USB interface, a sound card and a mike that supports DirectShow [5]. The overview of the application that we implement is as presented in Figure 1.

This work was supported in part by Electronics and Telecommunications Research Institute (ETRI), in part by the Brain Korea 21 project of Ministry of Education, and in part by the National Research Laboratory project of Ministry of Science and Technology, 2003, Korea.

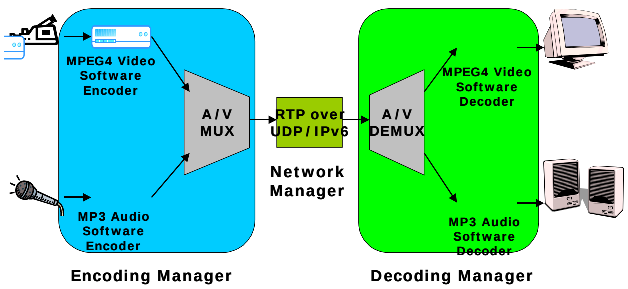

Figure 1. Software Design

2.1 The Encoding Manager

This part is to process the multimedia data acquired from a video camera and a microphone. The video stream captured by a video camera is compressed by MPEG4 codec and the audio stream from a microphone is compressed by MP3 codec and then is transmitted to A/V MUX (Audio and Video Multiplexer). The Codec used in this system are MPEG layer-3 codec and all of them are a software codec that is basically provided from Microsoft MPEG4 Video codec v1 and Windows. A/V MUX makes video stream and audio stream into one stream and sends it to network manager.

2.2 The Network Manager

This part takes charge of role that sends to or receives from destination multimedia data that receive through A/V MUX using RTP and RTCP protocol. Also this part takes charge of role that transmit video data through IPv6 network in addition to RTP header that receive from A/V MUX and that only remove data of payload in RTP packet that receive through network and pass to decoding manager. And also, this exchanges user information, bandwidth and other feedback information through RTCP [4]. We refer to detailed explanation of RTP and RTCP module in chapter 4.

2.3 The Decoding Manager

This part is to process multimedia data received through network. A/V DEMUX (Audio and Video De-multiplexer) separates stream received from network manager into video stream and audio stream and sends each of them to MPEG4 decoder and MP3 decoder. Decoded Video is looked on screen through Video Renderer and audio is played by speaker through soundcard. MPEG4 and MP3 decoders used to this part are software decoders that are offered basically in Windows. Also the decoding manager achieves role that accept user information, data amount that it send and receive, and the error rate etc. that receive from RTCP beside necessary output of multimedia data in video conferencing and represents to GUI (Graphic User Interface).

3. System Architectures

This videoconference tool has architecture as presented in Figure 2. Socket used to send/receive of RTP data is operated to different thread and use shared buffer between play modules and capture implemented filter of DirectShow. UI takes charge of the overall control function of this application.

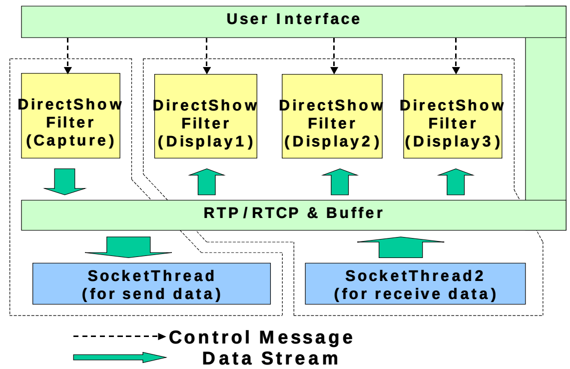

Figure 2. Architecture of HVCT

3.1 User Interface

The user interface acts as container of videoconference tool. This shows video of person participated in conferencing now and provides individual play/stop of video and audio and volume control. Also this application plays the role that shows user information of a name, address of conferencing participant and information related network of bandwidth which have been used currently and error rate and so on from RTCP database.

3.2 Sender

The sender does capture video and audio in high-quality videoconference tool and then transmits it by network. The following Figure 3 describes an architecture of sender. Video has resolution of 320 X 240 pixel and captured by 30 fps (frame/sec). And audio is encoded by 56Kbps.

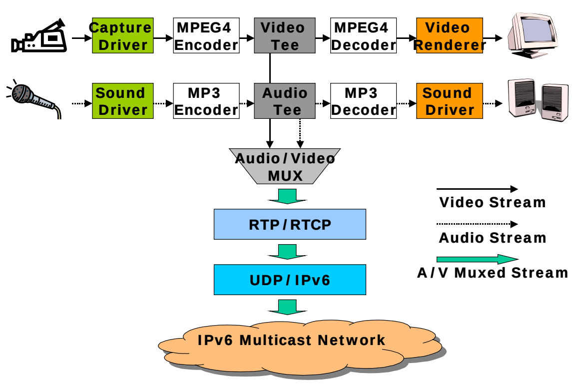

Figure 3. Architecture of A/V sender

This module provides video camera or sound card driver and codec as filter by DirectShow technology and provides interface for buffer negotiation or stream control between filters. Data stream that is separated from video and audio Tree adds RTP header and transmit it to network by IPv6 socket.

3.3 Receiver

RTP packets that are transmitted through network are classified according to each source in application layer and each of source created filter graph such as Figure 4 and reproduce video and audio. Receiver is possible to control of screen and audio of each of user so that used to media control interface provided by filter graph.

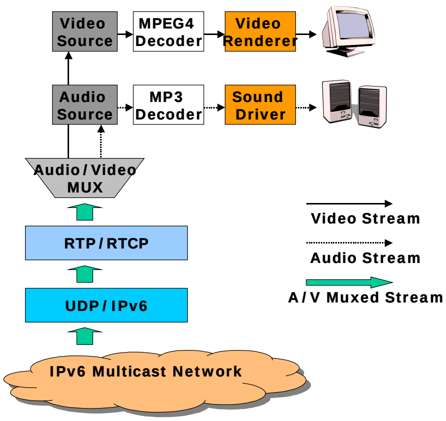

Figure 4. Architecture of A/V receiver

4. RTP, RTCP Module

This chapter describes operation of RTP, RTCP protocol that is used in this application and message flow that can establish video conferencing between multi-users.

4.1 Message Communication

The following Figure shows operation of functions is defined for operation of RTP, RTCP module.

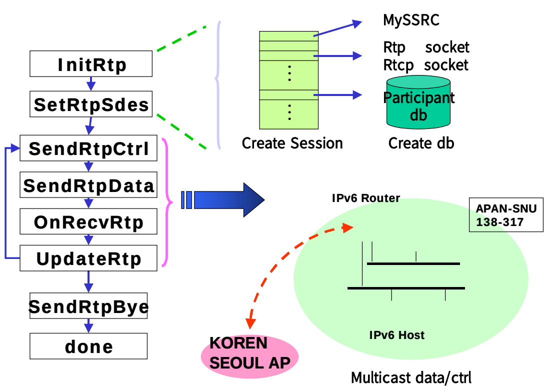

Figure 5. Flow of RTP/RTCP module

First, this module initiates a session through InitRtp(). The Session may be thought as a multicast group which does video conferencing and consist of one multicast address and port. SetRtpSdes() archive to database generated own information for its SDES(Source Description) when initialize a session. This database archives all information of session participants later.

If initialization routine succeeds, it iterates a roof and is executed four functions such as upside Figure 5. That is, this module call a SendRtpCtrl() to send RTCP data, a SendRtpData() to send multimedia data, a OnRecvRtp() to receive RTP and RTCP data and a UpdateRtp() that update session database to keep using received information. There are notifies participants using SendRtpBye() to terminate session and then is executed a done() completely to close a session after kill several processes is executing for video conferencing.

4.2 Multicast message communication

The following Figure 6 presents the communication of RTP, RTCP packets that generate when participant 1 participated in video conferencing that uses the multicast address is ff0e::1:2:3 and the port number is 5002.

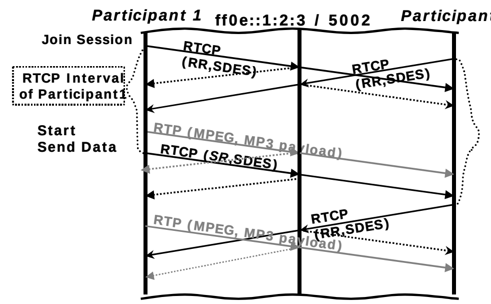

Figure 6. Communication sequences

Participant 1 informs own existence using RTCP packet including RR (Receiver Report) and SDES (Source Description) attending to session.

This RR provides statistical information related to receive data from each of sender so that includes receiving situation information about sender of all RTP data, which is receiving. Participant 2 that receives this packet creates entry of participant 1 in its participant information database.

When participant 1 begins to transmit multimedia data to session participants (Start Talking), then transmits data through RTP packet.

Participant 2 that receive this RTP packet registers participant 1 to transmitter (Active Source) in active state and sends data that receive to decoding manager.

And then RTCP packet that participant 1 in active state sends informs to everybody that participant 1 is the transmitter that includes SR (Source Report) in active state.

And this SR included a data size transmitted up to now cause receiver to analyze the network situation and to help multimedia data play in sender.

5. Experiment

In this chapter, first we introduce about a native IPv6 network, which connected Seoul National University to ETRI using PC-based IPv6 routers and ATM switches. Second, we show our experimental result in this IPv6 network.

5.1 Experimental Environment

A native IPv6 network in Seoul National University, as depicted in Figure 7, connected to Seoul AP. and Seoul AP constructs native IPv6 and is connecting with ETRI. For this experiment, we are doing IPv6 BGP peering with ETRI. So we can experiment of HVCT with ETRI consequently.

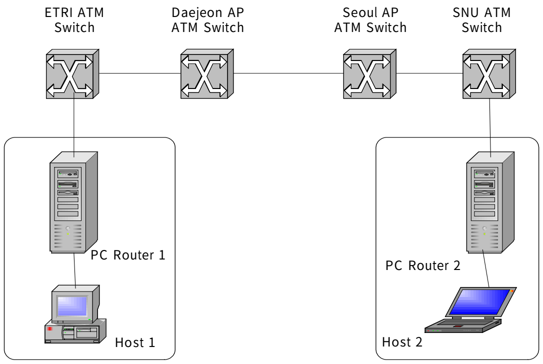

Figure 7. Network configuration

5.2 Multimedia Quality

The HVCT supports MPEG (Moving Picture Experts Group) 4 codec for high quality video data. This MPEG 4, a codec combines conversation style function (interactivity) of computer and transmission function of communication to support interactivity multimedia broadcasting, mobile multimedia communication, and next generation multimedia service of various forms such as Internet can encode object that include all of multimedia data including audio and video that need to real time communication, broadcasting, web, movie, and game and so on as is independent and flexible. Also the MPEG 4 use low bandwidth by compression encoding technology, 5Kbps to 150Mbps' bit rate though the quality does not fall off compare with MPEG 1. The HVCT using MPEG 4 codec shows 350 to 400 Kbps' bit rate.

And the HVCT uses MP3 codec for high quality audio. Traditional voice encoders like G.711 uses PCM [6]. More specifically, existing digital audio employs PCM with 16 bits per sample and sampling rates of 32 kHz, 44.1 kHz, 48 KHz [7], and this leads to bandwidth consumption of a few Mbps. With MP3 encoding, however, 128 to 256 kbps achieves almost equivalent audio quality. As mentioned above, the HVCT uses MPEC 4 codec and MP3 codec to support high-quality video and audio.

5.3 Bandwidth Consumption

The video data that receive from PC camera by 15 fps (frame per second) comes in to I420 format to MPEG 4 encoder. The codec process the data and it output encoded data to achieve average 350 to 400 Kbps. Also the MP3 codec receive audio data from microphone and it output data with two-channel stereo and 44.1 kHz sampling rate (bandwidth requirement will be 16bits/sample * 44,100 * 2 1.35 Mbps) to achieve 56 to 64 Kbps. Hence by deploying MPEG 4 codec and MP3 codec we can acquire a mount of effectiveness in terms of bandwidth consumption to support high quality multimedia data.

We compare our high-quality videoconference tool (HVCT) with videoconference tool (VIC), which is developed by UCL and supports h.261 encoding. Retaining the similar quality, we choose h.261 codec using RGB 24 format and 10fps for the VIC instance, and bits rate of both of VIC and HVCT is measured audio and video data together.

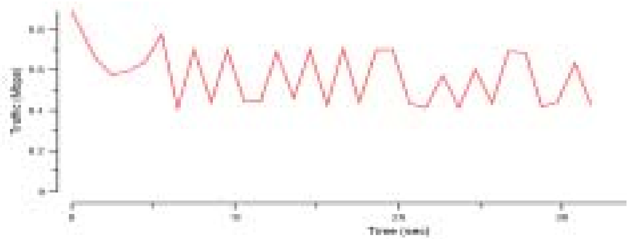

Figure 8. Bandwidth consumption of HVCT

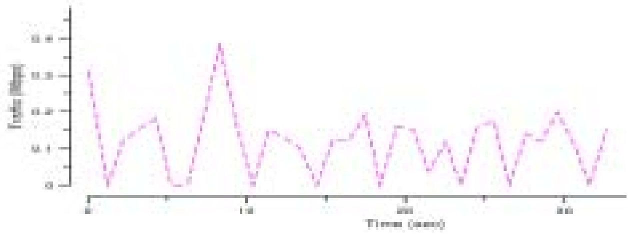

Figure 9. Bandwidth consumption of VIC

Figure 8 and 9 depicts the bandwidth consumption of HVCT and VIC, measured by tele traffic tapper (TTT) [8]; In Figure 8, MPEG 4 codec send data by average 400Kbps' bits rate and MP3 codec sends data to achieve 64Kbps. HVCT got a average bandwidth requirement about 500 Kbps. The figure shows saw-toothed shape. Because MPEG 4 codec sends I frame every 5 seconds periodically. And Figure 9 presents a VIC's bandwidth consumption. H.261, which was designed for data rates, which are multiples of 64Kbit/s. These data rates suit ISDN lines so we can observe that the bandwidth requirement of VIC is about two times lower than that of HVCT with MPEG 4 encoding. But HVCT support significantly high quality of audio and video better than VIC support that.

5.4 Delay

As presented above, MPEG 4 and MP3 codec extremely enhances high-quality multimedia data but it is a common concern that the complexity of encoding procedure would cause substantial amount of end-to-end delay that is not appropriate for interactive voice communication where the maximum end-to-end delay of 200 to 300 ms should be guaranteed [6]. Without loss of generality, assume that the latency caused by data transmission over the network is dependent of video and audio coding mechanism in use, the main concern would be the latency caused by MPGE 4 and MP3 encoding and decoding.

Table 1. Video Delays

Step	Latency measurement timing	Latency
1	MPEG4 Codec	40
2	Video Tee	5
3	Video Sending Buffer	20
4	RTP/RTCP	5
5	Video Receiving Buffer	700
6	MPEG4 Decoder	20
	Total Latency	790

Table 2. Audio Delays

Step	Latency measurement timing	Latency
1	Audio Capture device	45
2	MP3 Codec	45
3	Audio Tee	5
4	Audio Sending Buffer	10
5	RTP/RTCP	5
6	Audio Receiving Buffer	700
7	MP3 Decoder	40
	Total Latency	850

Above Tables present measured latency without concerning network delay. In Table 1 is case of video. In step 1, a process that is encoded video data from PC camera and has little amount of delay time about 40 ms. But this measurement result is represented in the environment, which CPU is lower loaded. However, when many people exist in videoconference session, delay of the encoded process increases. In step 2 to step 3, encoded data that is by MPEG4 waits for transmit to network being copied from Video Tee to Video Sending Buffer by RTP. Delay of this step is about 25 ms. In step 4, it is process that data is delivered through the network using RTP and RTCP module. And we do not consider network delay. Encoded data that receive from sender entered Video Receiving Buffer to decode data.

Over a mount of data that is considered delay of network and processing capacity of system exists in buffer, the data is started decoding and play for the first time. So in step 5 means buffering time to play video about 700ms. This step occupy bigger portion of end-to-end latency. The delay can reduce adjustment size of buffer properly, but if we set size of buffer so small, decoded the video can be a falling-off in quality. Delay in MPEG4 decoder is about 20ms as similar to encoding process. Whole delay of end-to-end are 790ms and actuality network delay can added one.

In Table 2, a delay that is processed audio data is handled similar to process of video data. The bigger portion of end-to-end latency would be incurred by using a software encoder instead of hardware one; most (approximately 80%) of the end-to-end latency is introduced by decoding process. One of the reasons that cause the substantial amount of latency is mp3 decoder waits for a frame of 2048 bytes, which imposes that the decoding cannot be started without being delayed for a certain amount of time.

Video data that ready to play should not synchronized with audio data, because the delay of each of data is different and delay of network is grew to unexpected one. This problem may solve that apply synchronization techniques, that adjust a time stamp of a video data based on the audio instead of send data mixed video and audio.

6. Conclusion

In this paper we present our high-quality videoconference tool, which deploys MPEG 4 (Moving Picture Experts Group 4) and MP3 (MPEG-1 layer 3) compression as the video and audio data coding.

Experimental results assert that high compression ratio can provide high-quality multimedia, and show the potentialities of MPEG 4 and MP3 as an encoding technique for interactive video and audio; It is true that the our conference tool introduces substantial end-to-end delay around 900 ms on average, which is not suitable for interactive communications that requires fast responses. And CPU of system is overloaded by using software codec. However, the problem, that most (about 80%) of the end-to-end delay is induced by the decoder could be significantly lowered by tuning the decoding process such as buffer size adjustment and that CPU of system is overloaded by using software codec should be solved that the number of video frame decease by reduce from 30 fps to 15 fps.

The HVCT that be used in Microsoft Windows 2000 and that be supported high quality videoconference tool can supply so that can use universally. In conclusion, The HVCT could be expected to contribute in IPv6 deployment offering various statistics data of traffic measurement [9].

7. REFERENCES

[1] 신명기, 김용진, “IPv6 도입을 위한 기술개발 동향,” IPv6 포럼 코리아 기술문서 2000-002, http://www.ipv6.or.kr.

[2] UCL Common multimedia library, http://www-mice.cs.ucl.ac.uk/multimedia/software/common/.

[3] RFC 1889, “RTP: A Transport Protocol for Real-Time Applications.”

[4] Microsoft IPv6 Tech-Preview for Windows 2000, http://msdn.microsoft.com/downloads/sdks/platform/tpipv6.asp

[5] Microsoft DirectX Homepage, http://www.microsoft.com/windows/directx/

[6] M. J. Karam and F. A. Tobagi. Analysis of the delay and jitter of voice traffic over the internet. In *Proc. of IEEE, INFOCOM 2001*, Anchorage, Alaska, USA, April 2001

[7] P. Noll. High quality audio for multimedia: Key technologies and MPEG standards. In *Proc. of IEEE Global Telecommunications Conference (GLOBECOM)*, Rio de Janeiro, Brazil, November 1999.

[8] K. Cho. Tele traffic tapper, http://www.cs1.sony.co.jp/person/kjc/programs.html#ttt.

[9] 6NEAT (IPv6 NEtwork and Application Testbed), http://www.6neat.net