

論文99-36T-12-17

## 웹 기반의 화상회의 시스템의 설계 및 구현

(A design and implementation of the video conferencing system on the WWW)

金星眞\*, 朴容震\*

(Sung-Jin Kim and Yong-jin Park)

### 요 약

화상회의 시스템은 지리적으로 분산된 컴퓨터 사용자들이 회의 시스템을 공유함으로써 오디오 및 비디오 정보를 사용하여 회의에 참가할 수 있는 환경을 제공한다. 그러나, 기존의 화상회의 시스템은 전용시스템 도입을 필요로 하며, 시스템이 설치된 장소에서만 회의가 가능하므로 특정 플랫폼과 네트워크 환경에 사용자가 종속되는 문제점을 가지고 있다. 또한, 사용자들이 회의 참가자들의 네트워크 접속상황을 사전에 인지해야 하며, 반드시 동일한 종류의 회의 시스템 응용프로그램을 사용해야 하는 약점이 있다. 본 논문에서는 위와 같은 문제점을 해결하기 위해 웹 기반에서 운영되는 화상회의 시스템의 설계와 구현 방법에 관하여 논한다. 제안된 화상회의 회의 시스템은 회의시스템 응용이 서버에서 전송되어 웹 브라우저에서 수행되므로, 회의 참가자들은 웹 브라우저만으로 회의를 수행 할 수 있다. 회의 세션관리 및 회의 응용 도구, 멀티미디어 제어 및 전송 도구는 웹 서버에서 전담하므로, 웹 서비스가 가능한 곳에서는 네트워크 접속 상황에 관계없이 화상 회의가 가능하다. 웹 기반의 회의 시스템을 구현하는데 가장 중요한 과제인 회의 참가자간의 상호 작용과 지속적인 회의 세션 연결을 구현하기 위해 자바 기술을 사용하였으며, 오디오 및 비디오 제어 및 처리는 하드웨어 제어가 용이한 ActiveX 기술을 사용하였다.

### Abstract

A video conferencing system provides sharing the conference environment for geographically dispersed computer users who use the audio and video information. But the conventional video conferencing systems have some problems which are dependent on specific software and/or hardware and bound the certain platform and network environment. Furthermore the participants must know the information about other participants before joining the conference session and they have to use the same video conferencing system.

This paper describes design and implementation of the video conferencing system on the WWW to solve the mentioned problems. The conference applications are transmitted from a WWW server and executed in the participants Web browsers. The participant can carry out conference services by using only the web browser. The WWW server takes charge of conferencing management including the information related to the participants and provides supported conference tools such as whiteboard, chatting and multimedia controls. Therefore the participants can easily join the conference sessions and perform conference working regardless of network connection situations.

We used the Java to implement the seamless session connections and interaction between the conference participants which are the most important when implementing the video conferencing system on the WWW and used the ActiveX technology about the audio and video controls to make it easy the hardware control.

\* 正會員, 漢陽大學校 電子工學科

接受日字:1999年10月29日, 수정완료일:1999年11月29日

(Dept. of Electronic Engineering, Hanyang Univ.)

## I. 서론

화상회의 시스템은 지리적으로 분산된 회의 참가자들이 가상적인 회의 시스템을 공유 함으로써 비디오 및 오디오 정보를 사용하여 회의 세션에 참여할 수 있는 환경을 제공한다.

이러한 시스템은 자신의 화상을 얻게 해 주는 비디오 입력장치, 다른 참가자들의 화상과 음성 정보를 실시간으로 주고받을 수 있게 해주는 전송 매체, 수신된 정보를 보여주는 출력장치 등의 회의장비가 필요하다. 요즘은 멀티미디어 기능을 갖춘 일반 컴퓨터에 인터넷을 기반으로 하는 화상회의 시스템들이 대두되고 있다.<sup>[1][2]</sup> 하지만, 기존의 방식에서는 해당 화상회의 전용 프로그램이 설치되어 있어야 하며, 특정 플랫폼/네트워크 환경에 종속되는 점과 사용자들이 참가자들의 네트워크 접속 상황을 사전에 인지해야 하는 등의 단점이 존재한다.<sup>[1]</sup> 본 논문에서는 인터넷 서비스의 기본적인 사용자 인터페이스의 위치로 변해가고 있는 웹 브라우저를 회의 시스템의 인터페이스로 이용함으로써 이러한 단점들을 개선하였다.

화상회의 시스템이 웹 기반의 사용자 인터페이스를 가지게 되는 경우에는 기존의 화상회의 전용 프로그램을 사용하는 시스템보다 몇 가지 장점을 가지게 된다. 첫째, 웹 서비스에서 사용되는 프로그램들은 사용자의 특별한 지시 없이 필요한 프로그램들을 자동으로 다운로드하여 설치하기 때문에 설치과정이 간단하다. 둘째, 프로그램이 수정, 향상 되었을 경우 자동으로 갱신된 프로그램을 다운로드 받아 교체해주므로 회의 웹 페이지에서 최신의 서비스를 받을 수 있다. 셋째, 회의 상대방과 연결하기 위해서는 상대방의 네트워크 주소를 알아야 하는데 웹을 이용 할 경우 회의 웹 페이지의 URL(Uniform Resource Locator) 주소를 알면 원하는 회의에 참여 할 수 있다.

위와 같은 웹 기반의 화상회의 시스템을 설계하는 데에는 몇 가지 고려되어야 할 사항이 있다. 첫째, 화상회의 시스템에서는 동일한 회의 웹 페이지를 참조하고 있는 회의 참가자들 간의 상호동작이 필요한데, 웹 서비스에서는 동일한 웹 페이지를 접근하는 사용자들간의 상호동작이 불가능하다. 둘째, 화상회의 시스템에서는 회의에 참여하고 있는 동안에 서버와 브라우저간의

지속적인 연결이 요구되는데, 웹 서비스에서는 지속적인 연결이 고려되지 않은 HTTP(Hyper-Text Transfer Protocol)프로토콜이 사용되므로 다른 통신 방안이 제공되어야 한다.

본 시스템에서는 참가자간의 상호작용과 지속적인 세션 연결의 해결을 위해 별도의 회의 서버를 제작하여 웹 서버와 함께 운영하는 방식을 취하였으며, 자바 언어로 만든 회의 응용 애플릿이 회의 서버와 지속적인 연결을 보장한다. 또한 화상회의 시스템에서는 멀티미디어 정보를 주고 받기 위해 수신하여 출력할 수 있는 하드웨어에 대한 접근이 필요한데 자바 언어는 보안을 고려하여 컴퓨터 자원에 대한 접근을 금지하고 있다. 본 논문에서는 멀티미디어 정보를 다루게 될 멀티미디어 컨트롤은 하드웨어에 대한 접근이 가능한 ActiveX 기술을 이용하여 구현하였다.

본 논문 2장에서는 기존의 화상회의 시스템의 문제점과 웹 기반의 화상회의 시스템 장점 및 구현에 따른 기술적 과제의 해결 방안에 관하여 논한다. 3장에서는 시스템의 전체적 설계와, 4장에서는 시스템의 구현 및 실행 방식을 설명하고, 5장에서는 결론 및 향후 과제를 제시한다.

## II. 화상회의 시스템

### 1. 기존 화상회의 시스템의 문제점

#### 1) 화상회의 전용 응용프로그램

기존의 화상회의 시스템은 해당 회의 전용 응용 프로그램을 이용하기 전에 그에 필요한 하드웨어나 소프트웨어가 컴퓨터에 미리 설치되어 있어야 한다. 준비가 되어 있지 않다면 복잡한 설치과정을 거쳐야 하며 사용자가 설치 작업에 익숙하지 않다면 더욱 불편하게 된다.

#### 2) 화상회의 환경에 대한 정보의 사전 인식

기존의 화상회의 시스템에서 회의 참가자는 회의 세션에 참가하기 이전에 미리 상대방에 대한 정보를 사전에 알고 있어야 한다. 예를 들어, 인터넷에서 두 컴퓨터가 통신을 하기 위해서는 연결할 상대방의 IP주소를 미리 알고 있어야 하는데, 만일 사용자들이 PPP(Point-to-Point Protocol) 접속이나 DHCP(Dynamic Host Configuration Protocol)와 같은 방법으로 IP주소를 얻는 경우에는, 사용자가 인터넷에 연결을 설정할 때마다

호스트의 주소가 변경되기 때문에, 상대방이 현재 사용하고 있는 주소를 정확히 알 수가 없게 된다.<sup>[3][4]</sup>

## 2. 웹 기반 화상회의 시스템의 장점

### 1) 웹 브라우저의 사용

웹 브라우저는 웹 서비스의 보편성 때문에 PC, 워크스테이션, 인터넷 TV 등과 같은 다양한 호스트와 운영체제에서 실행 가능하도록 이식되어 있다. 웹 브라우저를 통해서 화상회의 시스템이 구현된다면 호스트 환경에 종속되기 보다는 여러 플랫폼에서 실행되는 범용의 시스템 구현이 가능하다. 또한 화상회의에 필요한 ActiveX 컨트롤이나 자바 애플릿은 웹 브라우저가 서버로부터 해당 페이지를 읽을 때 가져오므로 사용자는 화상회의 페이지에 접속하기만 하면 자동 설치와 함께 프로그램이 수정, 향상 되었을 경우 동적 갱신(live-update)에 관련된 모든 절차가 자동으로 수행되어 편리하다.

### 2) 네트워크 연결 상황에 대한 인지 불필요

웹 서비스에서는 URL(Uniform Resource Locator)을 사용하여 인터넷 상에 있는 자원에 접근한다. 회의 참여자들은 화상회의를 위한 웹 페이지의 주소를 알고 있는 것만으로 자유롭게 회의에 참여 할 수 있게 된다

## 3. 웹 기반 화상회의 시스템의 기술적 과제

### 1) 회의 참여자들 간의 상호동작

화상회의 시스템에서는 회의 참가자들을 관리하기 위해서 여러 시스템간의 상호동작이 필요하다. 그러나 현재의 웹 서비스 환경에서는 서버에 연결한 웹 브라우저가 동일한 서버에 연결되어 있는 다른 웹 브라우저를 인지할 수 없도록 설계되어 있고, 그들 사이의 정보교환도 불가능하다. 따라서, 이들 회의 참가자들이 서로 정보를 주고받기 위해 화상회의의 서버 쪽에 상호동작을 위한 별도의 회의 관리자가 존재해야 한다.

### 2) 웹 서버와 웹 브라우저 간의 지속적인 연결

웹 서비스에 기본적으로 사용되는 프로토콜인 HTTP(HyperText Transfer Protocol)는 비연결 지향성 프로토콜이다. 웹 서버는 웹 브라우저의 서비스 요청에 대한 응답을 보내고 난 후, 웹 브라우저와의 연결을 종료한다. 하지만, 화상회의 시스템에서는 회의 참가자가 회의를 끝내기 전까지 계속해서 연결이 보장되어야 하므로 HTTP 자체만으로는 이 문제를 해결할 수가 없다.<sup>[5][6]</sup> 따라서 본 논문에서는 자바 기술을 이용하여 서버와 브라우저간의 지속적인 연결을 가능하게 하였다.

### 3) 화상 및 음성 정보를 얻어내는 방법

화상을 전송하기 위해서는 카메라로부터 화상 데이터를 얻어내는 하드웨어 제어 프로그램이 필요하고 음성을 얻기 위해서는 마이크로부터 음성 데이터를 얻어내는 하드웨어 프로그램이 있어야 한다. 웹 기반에서는 플랫폼에 독립적인 특성을 갖는 프로그래밍 언어로 자바 언어를 사용한다. 그러나 자바 언어는 작성된 프로그램이 네트워크 환경에서 사용된다는 특징 때문에 보안성을 상당히 고려하고 있어 다운로드 되어 실행되는 자바 애플릿은 파일의 삭제나 수정, 데이터의 삭제 작업과 같은 컴퓨터 하드웨어 자원에 접근하는 일을 방지하도록 설계되어 있다. 그러므로 비디오 장치로부터 화상 데이터를 얻어내는 작업과 오디오 장치로 음성 데이터를 얻어내는 작업에는 부적합하다.<sup>[7]</sup> 따라서 본 논문에서는 화상 및 음성 데이터를 얻기 위해 ActiveX 기술을 이용하였다.

## III. 웹 기반 화상회의 시스템 설계

본 논문에서 제안하는 시스템은 웹 서버에서 실행되는 부분과 웹 브라우저로 전송되어 실행되는 부분으로 나뉘어진다. 웹 서버에는 회의를 총괄하는 회의 관리자와 개별적인 회의 세션을 관리하는 세션 관리자가 구동되며, 회의에 참여하고 있는 사용자들의 각 브라우저에는 응용관리자와 멀티미디어 컨트롤이 서버로부터 전송되어 실행된다. 자바로 구현된 화이트보드 및 채팅과 같은 회의 응용 프로그램이 브라우저로 전송된 후 TCP/IP 프로토콜을 이용하여 서버와 지속적인 세션을 유지 할 수 있으며, 회의와 관련된 여러 정보를 주고 받을 수 있다[8]. 회의 응용 프로그램들은 세션 관리자와 통신을 하며 서로 데이터를 교환하고 양방향 오디오와 비디오 응용들은 세션 참가 후 직접 상대방의 응용과 데이터를 교환한다.

### 1. 서버측 요소

서버측에는 회의 웹 페이지를 전송해주는 일반적인 웹 서버가 있다. 클라이언트 요청이 있을 때마다 회의 웹 페이지, 응용관리자, 채팅과 화이트보드 프로그램, 비디오 컨트롤, 오디오 컨트롤 등 요구 되어진 것들을 전송해 준다. 그림 1의 회의 관리자(Conference Manager)는 회의 시스템 내 한 개만 존재하며 전체 회의 시스템을 총괄한다. 회의에 참가하려는 사용자에게

현재 진행중인 세션에 대한 정보를 제공하며, 회의에 참가하려는 사용자와 세션 관리자를 연결해 주는 역할을 수행한다. 세션 관리자(Session Manager)는 회의 세션 수 만큼 존재하며 회의에 참가하고 있는 사용자에게 회의를 진행할 수 있는 환경을 제공해 주며 각각의 개별적인 세션 정보를 관리하고 각 참가자들의 회의 응용이 보내온 정보를 세션내의 모든 회의 응용에게 멀티캐스팅 해 준다. 즉, 웹 브라우저의 채팅과 화이트보드 회의 응용 프로그램으로부터 입력 받은 기본적인 회의 데이터를 세션 내의 다른 참여자들에게 전송해주는 역할을 수행한다.

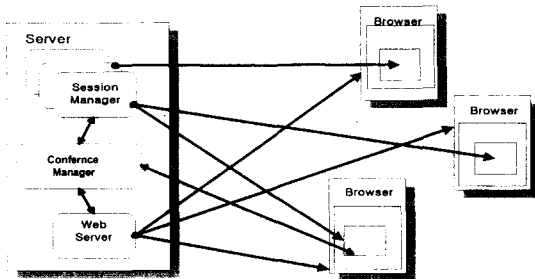


그림 1. 서버측의 내부구조  
Fig. 1. Server side internal structure.

## 2. 클라이언트측 요소

웹 브라우저 내에서 실행되는 클라이언트측 요소에는 서버측의 세션관리자와 지속적인 연결을 통해서 정보를 주고 받는 그림 2의 응용관리자(Applet Manager)와 채팅, 화이트보드와 같은 텍스트 기반 회의 응용 프로그램과 멀티미디어 데이터를 주고 받는 멀티미디어 컨트롤 등이 있다. 응용관리자는 화이트보드와 채팅을 위한 도구 쓰레드를 제어하고 오디오/비디오 응용을 제어하기 위한 이벤트를 브라우저에게 발생 시키는 역할을 한다. 응용관리자를 통해서 동일한 세션에 참가해 있는 사용자들의 데이터들을 공유할 수 있게 된다. 이때 채팅, 화이트보드의 정보 공유는 세션 관리자를 통해서 교환되는 정보량이 적으므로 별도의 멀티캐스트를 위한 네트워크 시스템을 갖추지 않고 서버의 세션 관리자를 거쳐서 정보를 공유 할 수 있다. 멀티미디어 컨트롤은 비디오 컨트롤과 오디오 컨트롤로 구성되며, 각각 비디오 입력장치와 오디오 입력장치로부터 데이터를 입력 받아 상대방에게 전송하게 된다. 멀티미디어 컨트롤은 상호간에 주고 받는 데이터량이 많기 때문에

서버의 세션 관리자를 통해서 데이터를 주고 받을 경우 네트워크 상에서 멀티캐스트를 지원하지 않으면 서버측의 네트워크에 과부하가 걸리기 때문에 멀티미디어 컨트롤 상호간에 직접 데이터를 주고 받도록 설계하였다.

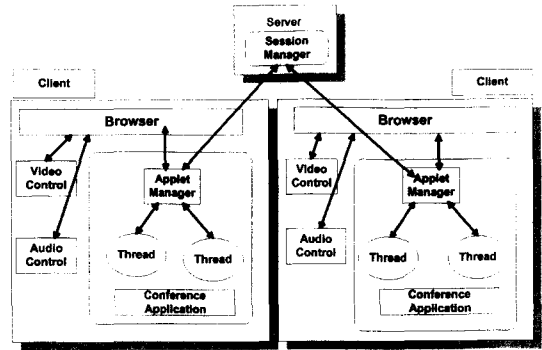


그림 2. 클라이언트측의 내부구조  
Fig. 2. Client side internal structure.

## IV. 화상회의 시스템 구현 및 실험

### 1. 시스템의 구현 방법

#### 1) 자바를 이용한 구현

서버측 요소의 웹 서버는 일반적으로 사용되는 웹 서버이면 어느 것이든 가능하다. 회의 관리자와 세션 관리자는 웹 서버와 같이 실행 되어야 하며, 플랫폼에 독립적이어야 하므로 자바언어를 이용해야 한다. 자바 언어는 이동코드 기술로 서버로부터 실행 가능한 프로그램인 애플릿이 전송되어 브라우저 안에서 독립적인 프로그램 수행이 가능하게 해주는 특징이 있다. 웹 브라우저로 전송되어 실행되는 응용 관리자와 채팅, 화이트보드 회의 응용 프로그램 역시 자바 애플릿을 이용하였다.

#### 2) ActiveX를 이용한 구현

화상회의는 기본적으로 비디오/오디오 데이터의 양방향 전송 및 제어를 요구하는데, 웹상에서 이러한 멀티미디어 데이터를 처리하기 위해서는 다음의 두 가지 사항을 고려하여야 한다. 첫째는 멀티미디어 데이터 처리는 웹 브라우저에 과도한 부하를 가져 올 수 있다. 처리하여야 할 데이터의 크기가 크고 실시간으로 처리하여야 하므로 자바 언어와 같이 인터프링팅 방식으로 수행되는 언어에는 필요한 처리속도를 기대하기 어렵

게 된다. 둘째는 멀티미디어 데이터를 입력 받기 위해서는 비디오 카메라나 오디오 하드웨어에 대한 접근이 필요하나 자바나 CGI에 의한 접근은 이와 같은 하드웨어 입력에 많은 제약을 받게 된다. 따라서 본 논문에서는 비디오/오디오 컨트롤을 구현할 때 ActiveX 기술을 이용한다. ActiveX는 자바와 마찬가지로 서버로부터 웹 브라우저에 전달되어 실행되는 구조를 가지고 있지만 클라이언트측에 설치되어 실행되는 형태를 가지기 때문에 인터프리티형식의 실행구조를 가지는 자바보다 실행속도가 빠르고 하드웨어에 대한 접근이 가능하다.<sup>[9]</sup> 또한 웹 브라우저 내에서 자바 애플릿 프로그램과 ActiveX 컨트롤이 공존하는 경우에 상호간의 동작을 위해서는 ActiveX 기술의 일부인 ActiveX 스크립트가 필요하다.<sup>[10]</sup> 이는 웹 페이지 내에 있는 객체들의 상호 동작을 지정해 주는 것으로써 세션 관리자로부터 멀티미디어 컨트롤에게 임의의 명령을 내리기 위해서는 웹 브라우저에 있는 응용관리자에게 명령을 내려서 응용 관리자가 같은 웹 페이지 내에 속해있는 멀티미디어 컨트롤 객체에게 이벤트를 발생시켜 그 동작을 제어하는 방식을 취하게 된다. 본 논문에서 사용한 ActiveX 스크립트는 VBScript를 이용하였다.<sup>[11][12]</sup>

2. 멀티미디어 컨트롤

1) 비디오 컨트롤

웹 브라우저 내에서 회의 참가자들에게 비디오 전송을 해 줄 수 있도록 해주는 컨트롤이다. 화상 입력 장치로부터 데이터를 입력 받아야 하는 특성상 ActiveX 기술을 이용한 컨트롤을 만들어야 한다. 그림 3처럼 비디오 컨트롤은 레코더 컨트롤(Recorder Control)과 플레이어 컨트롤(Player Control)로 나눌 수 있다.

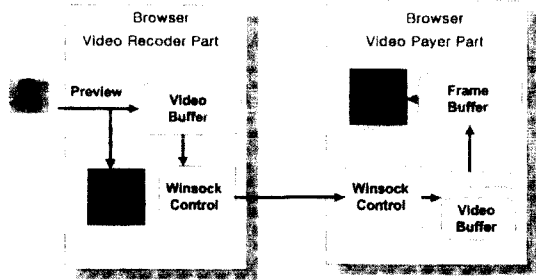


그림 3. 비디오 컨트롤의 구조  
Fig. 3. Video Control structure.

(1) 비디오 레코더 컨트롤

비디오 레코더 컨트롤은 일반적으로 윈도우 기반의 시스템에서 비디오 입력의 표준으로 사용되는 Video for Windows(VfW)라는 API를 이용하여 화상입력을 받는다.<sup>[13]</sup> 고가의 전용 화상회의 시스템에서는 입력 받은 비디오 데이터를 전송하기에 알맞도록 작은 크기로 압축하여 전송하나 실시간으로 이러한 작업을 해주기 위해서는 실시간 압축을 위한 별도의 하드웨어가 필요하기 때문에 일반적으로 쓰이지 않는다. 또한 상대방에서도 실시간으로 압축을 풀고 재생해야 하는 오버헤드가 발생하기 때문에 상대방 역시 별도의 하드웨어 사용이 불가피하게 된다. 이러한 장단점을 고려해보면 초당 입력 받는 비디오의 프레임 수를 줄여서 압축하지 않은 상태로 보내는 것이 효율적임을 알 수 있다. VfW API를 이용하여 얻어낸 비디오 데이터를 컨트롤 내부에서 상대방과 연결되어 있는 내장 인속 컨트롤에게 넘겨 주어 상대방에게 전송해 준다. 비디오 레코더 컨트롤에서 수행하는 동작은 다음과 같다.

- ① 하드웨어로부터 윈도우즈에 기본으로 제공되는 Video for Windows를 이용해서 비트맵 형식의 화상 정보를 얻은 후 콜백(Callback) 함수를 호출한다.
- ② 콜백 함수 내에서 비트맵의 헤더와 데이터를 버퍼에 저장한다.
- ③ 비디오 플레이어 컨트롤에게 먼저 저장된 비트맵 헤더를 전송하고 콜백 함수 내에서 데이터를 지정된 패킷 크기에 맞게 나누어 전송한다.

(2) 비디오 플레이어 컨트롤

비디오 플레이어 컨트롤은 크게 두 가지 기능을 하는 독립적인 쓰레드로 구성되어 있다. 하나는 송신측에서 전송해주는 비디오 데이터를 받아 순서에 맞게 버퍼에 저장하는 역할을 수행하며 다른 하나는 패킷으로 나누어 전송되어온 데이터를 조합해서 원래의 한 프레임의 데이터로 복원하는 역할을 수행한다. 수신을 담당하는 쓰레드는 화면 출력을 처리할 수 있도록 출력 쓰레드를 활성화 시킨다. 출력 쓰레드는 활성화된 후 해당 프레임을 정해진 화면에 보여주는 역할을 수행한다. 비디오 플레이어 컨트롤에서 수행하는 동작은 다음 순서와 같다.

- ① 레코더 컨트롤로부터 비트맵 헤더를 전송 받는다.
- ② 소켓 버퍼에 전송된 비트맵 데이터 패킷이 프레임 크기만큼 되면 한 프레임을 소켓 버퍼에서 받아 버퍼에 저장한다.

③ 버퍼에 저장되어 있는 프레임을 순서에 맞게 화면에 보여준다

2) 오디오 컨트롤

웹 브라우저내에서 오디오를 이용하기 위해서는 오디오 컨트롤이 필요하다. 오디오의 샘플링을 위해서 Wave API를 이용한다. 표준으로 제시된 MCI(Multimedia Control Interface)는 오디오를 샘플링 하거나 재생하는 것은 가능하다, 샘플링 한 데이터를 메모리 상에 적재할 수 없으며, 반드시 파일에 기록해야만 한다. 또한, 재생도 파일의 데이터만 재생할 수 있을 뿐, 메모리 상에서 직접 재생하는 것은 불가능하다. MCI는 파일 시스템을 이용한 일반적인 오디오 응용에는 적합하나, 본 논문에서 사용되는 것처럼 실시간 처리를 요구하는 응용에는 적합하지 않다. 실시간 처리용으로 샘플링 된 오디오 데이터는 메모리 상의 버퍼에서 네트워크에 연결된 TCP/IP 소켓으로 직접 전송되어야 할 필요가 있기 때문이다. 만일 파일 시스템을 경유한다면 심한 오버헤드 때문에 실시간 적인 대화는 불가능하므로 API 레벨에서 접근 할 수 있는 Wave API를 이용해야 한다. Wave API는 하드웨어에 보다 근접하여 정밀한 제어를 할 수 있는 API로써 메모리와 오디오 디바이스를 직접 제어 할 수 있다. 그림 4에서와 같이 네트워크에서 전송된 데이터는 수신측 임시 버퍼에 저장되었다가 재생하게 된다. 패킷의 크기는 약 800바이트로 구성되어 있으며, 버퍼에는 10개의 패킷이 들어있게 된다. 이 임시버퍼의 목적은 지터(Jitter)에 의해 생긴 지연을 상쇄하는 버퍼로 오디오와 같은 실시간 데이터는 수신측에 지터 상쇄 버퍼를 두어, 지연 발생을 상쇄하는 것이 일반적인 접근 방법이다. 양방향 오디오에 대해서, CCITT에서 권고한 지연시간은 약 400ms 이나 인터넷에 적용할 경우 지터 버퍼는 최소 1000ms에서 최대 2000ms의 크기를 갖도록 처리하였다.<sup>[16]</sup>

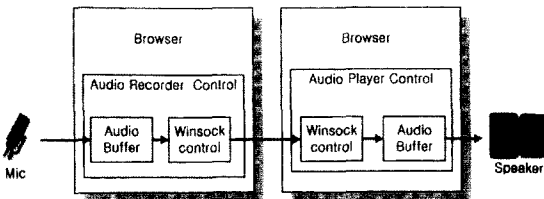


그림 4. 오디오 컨트롤의 구조  
Fig. 4. Audio control structure

(1) 오디오 레코더 컨트롤 (Audio Recorder Control)

지정된 형식의 오디오 데이터가 오디오 입력 장치로부터 버퍼에 채워지면 상대방에 전송된다. 레코더 컨트롤에 사용된 Wave API의 흐름은 다음 순서와 같다.

- ① 원하는 형식으로 오디오 디바이스를 열고 콜백 함수를 지정한다.
- ② 오디오 데이터를 저장할 디바이스 내의 버퍼를 여러 개 마련한다.
- ③ 연결이 설정되면 녹음을 시작한다.
- ④ 지정된 콜백 함수가 호출되면 소켓을 위한 버퍼로 데이터를 옮긴다.
- ⑤ 연결이 끊어지면 디바이스를 닫는다.

(2) 오디오 플레이어 컨트롤 (Audio Player Control) 상대방으로부터 오디오 데이터를 수신하여 버퍼에 저장하는 메인 쓰레드와 플레이어를 수행하는 쓰레드로 구분한다. Wave API를 중심으로 한 개략적인 순서는 다음과 같다.

- ① 연결이 되면 메인 쓰레드를 생성한다.
- ② 오디오 형식에 맞게 디바이스를 열고 콜백 함수를 지정한다.
- ③ 오디오 버퍼를 준비한다.
- ④ 수신이 되면 플레이어 쓰레드가 활성화되어 디바이스에 오디오 데이터를 출력한다.
- ⑤ 연결이 종료되면 쓰레드가 종료되면서 버퍼를 제거하고 디바이스를 닫는다.

3. 응용관리자와의 상호작용

ActiveX 컨트롤이 웹 페이지 내에서 서로 제어 정보를 주고 받기 위해서는 ActiveX 스크립트가 필요하다. 본 논문에서는 VBScript를 사용하였고 각각 발생시키는 이벤트에 따라 서로에게 정보를 제공해 주고 각 컨트롤들은 이러한 제어 이벤트를 발생시키고 수신하는 모듈을 포함하고 있다. 그림 5는 서버측의 세션관리자

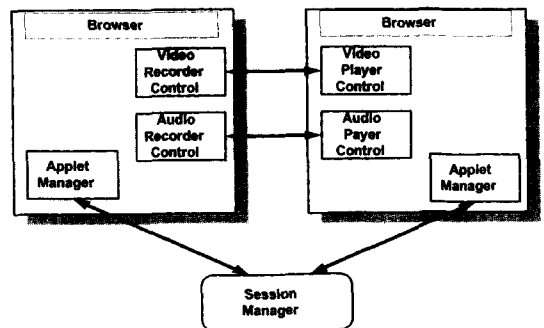


그림 5. 응용관리자와의 상호작용  
Fig. 5. Interaction of Applet Manager.

로부터 전송 받은 제어 메시지를 각각의 멀티미디어 컨트롤들에게 ActiveX 스크립트를 이용하여 전달하는 과정이다. 이러한 ActiveX 스크립트는 각각 자바로 구현된 애플릿과 비주얼 베이직을 이용하여 만든 ActiveX 컨트롤들 간의 메시지 교환을 가능하게 한다.

다음은 스크립트의 일부로서 비디오 컨트롤을 나타낸 부분과 응용관리자가 비디오 컨트롤에게 연결 설정을 지시하는 부분이다. <Object> 태그에 CLSID 변수를 통해서 비디오 레코더와 비디오 플레이어를 각각 지정하여 화면에 보여지게 된다.

```

<html>
<head>
<title>WebMeeting page</title>
<script language=VBScript>
<!
.
Sub Client_ConnectionVideo(add)
    VideoPlayControll.RemoteHost = add
    VideoPlayControll.RemotePort = 3456
    VideoPlayControll.ConnectToServer
End sub
</script>
</head>
<body>
<! -- Video Recorder -- >
<object id=VideoRecordControll width=160
height=151
classid= CLSID:8F9D205E-43D5-11D2-A49B-
0080C7C26EB8
codebase=control/Vrec.ocx#version=0,0,0,1>
    <param name=_ExtentX value=4233>
    <param name=_ExtentY value=3969>
</object>
<br>
</body>
</html>

```

#### 4. 실행 구조 및 실행 화면

##### 1) 회의 수행 절차

멀티미디어 컨트롤을 이용하여 화상회의가 이루어지는 절차는 다음과 같다. 회의에 참가하고자 하는 사용

자는 어떤 특정 세션에 참가하여야 한다. 세션이 생성되고 난 뒤에는 각 세션 관리자로부터 전송된 메시지를 다른 모든 회의 참가자들의 웹 브라우저로 멀티캐스트 한다. 각 웹 브라우저는 전송된 메시지를 응용관리자가 분석하여 해당 회의 응용 도구 쓰레드에 넘겨준다. 각각의 회의 응용 도구 쓰레드는 자신에 해당되는 메시지를 스스로 처리하므로, 응용관리자는 회의 응용 도구 쓰레드와의 데이터 통신만 관장하면 된다. 오디오와 비디오 회의 데이터는 ActiveX 컨트롤로 구현되었으므로 자바 애플릿과는 달리 서버를 경유하지 않는 1 대 1의 TCP 연결을 이용하게 된다. 다만, 처음 설정 할 때 응용관리자로부터 상대방의 IP주소와 포트 번호 정보를 넘겨받고 연결을 시도한다. 즉, 채팅과 화이트보드 프로그램은 응용관리자와 세션관리자의 연결을 통하여 텍스트를 기반으로 한 전송을 수행하고 멀티미디어 컨트롤들은 연결 설정시 응용관리자가 각각 비디오/오디오 레코더 컨트롤에게 상대방의 IP주소를 넘겨주면 각 컨트롤들은 독립적으로 상대방과 연결을 설정하고 데이터를 전송한다. 그림 6은 회의 수행 흐름도이다.

- ① 사용자가 회의에 참여하기 위해 화상회의 웹 페이지에 접속한다. 웹 브라우저는 페이지를 분석하여 페이지 내에 있는 응용관리자, 채팅, 화이트보드, 비디오 레코더 컨트롤, 비디오 플레이어 컨트롤, 오디오 레코더 컨트롤, 오디오 플레이어 컨트롤을 웹 서버로부터 전송 받아 실행시킨다.
- ② 현재 진행중인 세션에 대한 정보를 회의 관리자로부터 얻는다. 만일 새로운 세션을 요청한다면 회의관리자가 새로운 세션관리자를 생성시킨다. 회의 관리자는 해당 세션 관리자의 TCP/IP포트와 같은 정보를 사용자 웹 브라우저에 전달한다.
- ③ 사용자 웹 브라우저는 해당 포트의 세션 관리자에게 참여를 요청하고 자신의 정보를 세션 관리자에게 넘겨 준다. 세션 관리자는 새로운 사용자의 정보를 자신의 세션 데이터베이스에 저장하고 현 세션의 정보를 새로운 사용자에게 전달한다. 회의가 수행되어 각 응용관리자로부터 전송된 메시지를 세션관리자는 다른 모든 사용자들의 웹 브라우저로 멀티캐스트 한다. 각 브라우저는 받은 메시지를 응용관리자가 분석하여 해당 회의 응용 도구 쓰레드에 넘겨준다.
- ④ 멀티미디어 데이터 전송을 위한 연결 설정은 세

션 관리자를 통하지 않고 회의 참가자들 간에 직접 연결 된다. 응용관리자가 세션 관리자로부터 회의 참가자의 정보를 받아 오디오/비디오 컨트롤을 위한 이벤트를 발생한다. 브라우저가 VBScript로 작성된 이벤트 핸들러를 이용하여 오디오/비디오 컨트롤에게 상대방 정보와 함께 연결 설정을 명령한다. 각각의 컨트롤들이 상대방의 컨트롤에 접속한 후 세션 관리자와 별도로 데이터를 주고 받는다.

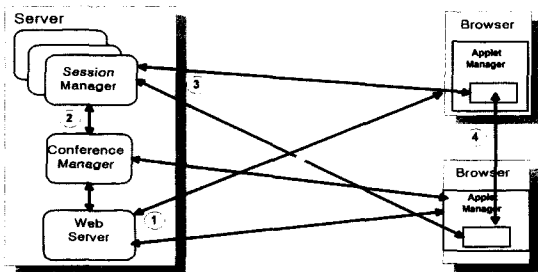


그림 6. 회의 수행 절차  
Fig. 6. Video conference procedure.

그림 7은 화이트보드, 채팅과 같은 회의 응용 도구와 멀티미디어 컨트롤들이 함께 브라우저 안에서 실행되는 화면 예이다.

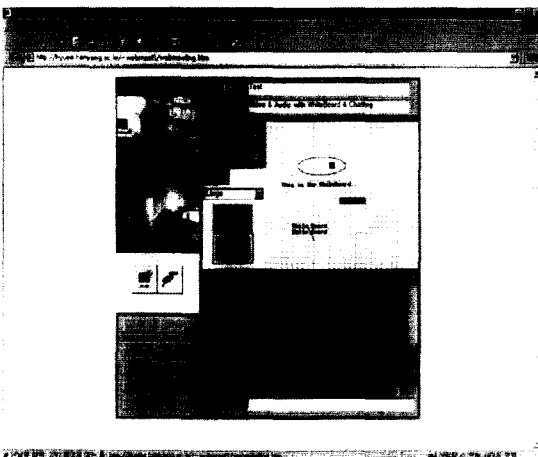


그림 7. 화상회의 실행화면  
Fig. 7. Video conference execution screen.

5. 운영환경 및 성능 평가

일반적으로 화상회의 시스템의 성능평가를 정량적으로 예측하기는 곤란하다. 화상회의 시스템의 성능은 처

리속도나 효율로 평가되기 보다는 사용의 편리성이나 운영환경에 의해서 결정되기 때문이다. 일대일 양방향 통신인 경우 최대 초당 2 프레임으로 시간 지연 없이 지속적으로 화상 데이터를 주고 받을 수 있다. 초당 2 프레임 이상의 전송을 수행할 경우에는 흐름제어를 하는 모듈이 없기 때문에 플레이어쪽의 버퍼가 넘치게 된다. 일대일 양방향 통신 시험환경으로 사용한 네트워크 환경은 10Mbps Ethernet LAN 환경이며, 웹 서버로는 Apache를 사용하였다. 비디오 디바이스는 QuickCam을, 오디오 디바이스는 Sound Blaster 16 을 이용하였다. 실험 시스템 사양은 다음과 같으며 현재 WAN 환경은 고려되어 있지 않으며 LAN 환경에서의 화상회의 시스템의 데모사이트 URL은 [hyuee.hanyang.ac.kr/~webmeet1/webmeeting/webmeeting.htm](http://hyuee.hanyang.ac.kr/~webmeet1/webmeeting/webmeeting.htm)이다.

표 1. LAN 연결시 실험 환경  
Table 1. Test Environments on LAN.

시스템 사양	서버	클라이언트 1	클라이언트 2
브라우저	Explorer	Explorer	Explorer
운영체제	SunOS 5.6 sparc	Windows'95	Windows'98

본 화상회의 실험은 사용의 편리성과 회의 수행 가능성 여부를 실험해 보았다. 현재 초당 2 프레임 정도의 처리와 익스플로러 상에서 만의 화상 회의였으나 상태는 비교적 좋았고 화이트보드와 채팅을 이용한 텍스트 기반의 회의 수행은 지연 없이 실행되었다. 멀티미디어 데이터 처리는 시스템 성능에 영향을 받으므로 시스템 사양에 따라 더 좋은 성능을 기대해 본다.

V. 결론 및 향후 과제

본 논문에서는 기존 화상회의 시스템의 약점인 전용 시스템의 도입과 회의 참가 시 회의 정보의 사전 인지도 등을 해결하기 위하여 웹 기반의 화상회의 시스템을 구현하였다. 구현된 화상회의 시스템에서는 참가자들이 웹 브라우저만을 이용하여 회의 상대방의 주소가 아닌 회의 웹 페이지 URL을 이용하여 편리하게 회의에 참여 할 수 있도록 지원하며, 사용자들의 네트워크 접속 상황이나 플랫폼에 제한을 받지 않도록 하였다. 그러나 웹 기반의 화상회의 시스템은 동일한 웹 페이지를 회의 참가자들 간의 상호 동작이 요구되는데 이를 지원



하지 못하며, 또한 회의 세션 동안은 서버와 브라우저의 지속적인 연결이 필요한데 HTTP 프로토콜 특성상을 지원하지 못한다. 이를 해결하기 위해 별도의 회의 관자와 세션관리자를 서버에 구현하였으며 회의 지원 응용 도구를 자바 애플릿으로 작성하여 회의 관리 서버와 지속적인 연결이 되도록 하였다. 멀티미디어 데이터를 처리하기 위해 멀티미디어 컨트롤 구현은 하드웨어 접근이 용이한 ActiveX로 작성하였다.

웹 서비스와 결합된 화상회의 시스템은, 앞으로 웹 서비스의 전망으로 보아 멀티미디어 통신의 기초가 될 수 있을 것이다. 또한 ActiveX기술이 자바와 달리, 현재로는 윈도우 기반의 플랫폼에서만 실행된다는 단점이 있지만, 앞으로 다양한 플랫폼을 지원하게 될 것으로 보인다. 따라서 웹 기반에서 운영되는 화상회의 시스템은 별도의 개발 없이 여러 플랫폼에서 실행될 수 있을 것으로 기대된다.

### 참 고 문 헌

- [1] David M. Fetterman, Video conferencing On-line: Enhancing Communication over the Internet, <http://cu-seeme.comell.edu/Fetterman.html>, 1998.
- [2] Microsoft, "Microsoft NetMeeting Overview, <http://www.microsoft.com/netmeeting/learn/user.htm>, 1998.
- [3] W. Richard Stevens, *TCP/IP Illustrated Volume 1*, Addison Wesley, pp. 21-27, 1994.
- [4] Watson Research Center, IBM Corp., Address Allocation for Private Internets, RFC 1597, March, 1994.
- [5] HTTP Working Group, Hypertext Transfer Protocol HTTP/1.1, <http://www.ics.uci.edu/pub/ietf/http/draft-ietf-http-v11-spec-rev-05.txt>, Internet Draft, Expires March 11, 1999.
- [6] Apache Server Project, Apache Keep-Alive Support, <http://www.apache.org/docs/keepalive.html>, Apache Documentation.
- [7] Javasoft, The Java Language Environment White Paper, <ftp://ftp.javasoft.com/docs/WhitePaper.pdf.zip>.
- [8] 정의현, 박용진, WWW상에서의 공동작업 시스템의 설계 및 구현, *정보과학회논문지(C)*, 제 3권 제 4호, pp. 384-396, 1997년 8월
- [9] David Chappell, *Understanding ActiveX and OLE*, Microsoft press, pp. 301-305, 1996.
- [10] Guy Eddon, *Active Visual Basic 5.0*, Microsoft Press, pp. 243-244, 1997.
- [11] Tom Armstrong, *ActiveX Controls-Designing and Using*, Chapter 1, 1997.
- [12] Microsoft, ActiveX Controls Papers and Presentations, <http://www.microsoft.com/com/ActiveX.asp>.
- [13] Microsoft Platform SDK-Graphics and Multimedia Services, Video Capture, [http://premium.microsoft.com/isapi/devonly/prodinfo/msdnlib.idc?theURL=/msdn/library/sdkdoc/multimed/avicap\\_3bfs.htm](http://premium.microsoft.com/isapi/devonly/prodinfo/msdnlib.idc?theURL=/msdn/library/sdkdoc/multimed/avicap_3bfs.htm).
- [14] Microsoft Platform SDK-Windows Base Services, Multiple Threads, [http://premium.microsoft.com/isapi/devonly/prodinfo/msdnprod/msdnlib.idc?theURL=/msdn/library/sdkdoc/winbase/prothred\\_5ysz.htm](http://premium.microsoft.com/isapi/devonly/prodinfo/msdnprod/msdnlib.idc?theURL=/msdn/library/sdkdoc/winbase/prothred_5ysz.htm).
- [15] Scott Jarol and Anthony Potts, *Visual Basic 4 Multimedia Adventure Set*, pp. 116-139, Coriolis Group Books, 1995.
- [16] Microsoft Platform SDK Graphics and Multimedia Services, Waveform Audio, [http://premium.microsoft.com/isapi/devonly/prodinfo/msdnprod/msdnlib.idc?theURL=/msdn/library/sdkdoc/multimed/wave\\_7jcf.htm](http://premium.microsoft.com/isapi/devonly/prodinfo/msdnprod/msdnlib.idc?theURL=/msdn/library/sdkdoc/multimed/wave_7jcf.htm).

## 저 자 소 개

## 金星眞(正會員)

1981년 동국대학교 전자계산과 졸업. 1983년 한양대학교 산업대학원 전자계산과 졸업. 1993년 한양대학교 대학원 전자공학과 박사수료. 1983년~1988년 한국 데이터통신(주) 정보통신연구소 주임연구원. 1992년~현재 한양여자대학 전산정보처리과 부교수. 주관심 분야 : 컴퓨터통신, 화상회의

## 朴容震(平生會員)

1969년 와세다대학교 전자통신공학과 졸업. 1971년 와세다 대학원 공학석사. 1978년 와세다 대학원 공학박사. 1979년~현재 한양대학교 전기전자공학부 교수. 1983년~1984년 Univ. of Illinois, Urbana 전산학과 방문 부교수. 1991년~1992년 영국 Kent 대학 방문교수. 주관심 분야 : 컴퓨터통신, 분산시스템, 초고속통신, 이동 데이터통신