

캠퍼스 망에서 비디오 강의 시스템 구축

(An Implementation of a Video Lecture System in a Campus Network)

배 시 규
(Shi-Kyu Bae)

요 약 통신망을 통해서 동영상을 전송하는 기술은 다양한 응용이 가능하다. 하지만, 현재의 인터넷 환경에서는 저속과 대역폭의 큰 변화로 인하여 많은 제약이 따른다. 학교에서 동영상과 음성으로 이루어지는 강의 시스템을 구축함으로써 효율적인 강의로 학습 효과를 증대시키고, 전체 실습실의 활용도를 높일 수 있다. 본 논문은 실시간 전송을 위한 RTP(Real Time Protocol)를 사용하여 캠퍼스내에 강의 시스템을 구축하기 위한 것으로, 멀티캐스트 라우터로 구성된 인터넷에서 Java와 JMF(Java Multimedia Framework)를 사용하여 개발이 이루어졌다.

1. 서 론

인터넷에서 문자 정보뿐만 아니라 오디오나 비디오와 같은 멀티미디어 정보를 활용하는 경우가 급격히 늘어나고 있으며, 실제 인터넷상에서 차지하는 멀티미디어 트래픽의 비중이 기존의 문자 트래픽과 대등하거나 이보다 앞서가고 있는 추세이다.

웹서버를 통하여 멀티미디어 데이터를 제공하는 응용이 가장 일반적이지만, 멀티미디어의 특성을 살려 성공적인 재생을 하는데는 많은 문제점이 나타난다. 이는 인터넷의 불규칙한 지연 시간과 지터가 주된 원인이지만, 웹 서버가 멀티미디어의 특성을 제대로 처리하지 못하는 것도 큰 문제점이다.

웹서버 동작의 한계를 극복하기 위해서는 멀티미디어 특성을 적절하게 처리할 수 있는 미디어 전용 서버를 별도로 사용하고 클라이언트측의 재생 프로그램과 함께 적절한 처리를 해 주는 미디어 스트리밍 방법이 필요하다.

인터넷을 통하여 오디오와 비디오를 실시간으로 전송하는 기술은 다양한 응용 분야가 가능하기 때문에 많은 연구와 함께 실용화가 속속 이루어지고

있다.

현재 인터넷의 주요 전송 프로토콜인 TCP와 UDP는 시간 연속(time-continuous) 미디어를 전송하기에는 부적합한 점이 많다. 이를 개선하기 위한 노력으로 RTP가 제안되었으며 업계의 표준으로 자리잡아가고 있다[1]. RTP는 기존의 프로토콜과는 달리, 모든 인터넷 환경에서 미리 설치되어 있지 않고 응용 프로토콜에서 직접 구현을 해주어야 함으로 응용 프로그램 개발자 입장에서는 구현하는데 적잖이 부담이 되고 있다.

최근에 Java를 개발한 SUN사와 IBM을 비롯한 몇 개 업체에서 공동으로 Java 환경에서 멀티미디어 처리를 가능하게 하는 패키지를 개발하여 제공하고 있으며, 이를 JMF라고 명명하였다. 특히, 최근에 구현된 JMF에는 인터넷에서의 멀티미디어 전송이 가능하도록 RTP를 사용하는 클래스도 추가하였다[2-5].

본 논문은 여러 건물을 백본망을 통하여 연결한 캠퍼스 인터넷 환경에서 Java 개발 도구와 JMF를 사용하여 동영상 강의 시스템을 구축하기 위한 방법에 관한 것이다.

2장에서는 RTP에 관하여 설명하고, 3장에서는 JMF의 구성과 동작, 특히 RTP를 사용하는 기능에

* 동양대학교 컴퓨터공학부 조교수

대하여 설명한다. 4장에서는 이를 사용하여 강의 시스템에서 비디오와 오디오를 전송하는 서버와 클라이언트 프로그램으로 이루어지는 동영상 강의 응용 프로그램을 개발하는 과정을 설명한다. 마지막으로 결론을 다루었다.

2. RTP와 RTCP

RTP는 인터넷에서 멀티미디어 스트림을 전송하기 위한 전송 프로토콜로서, 멀티캐스트 IP와 UDP에서 동작한다. RTP는 다른 프로토콜 위에서 동작하며, 하부 프로토콜의 제약을 최소화하기 위하여 오류 제어나 흐름 제어 등을 하지 않는 경량화 프로토콜이다[1].

RTP는 RTCP(Real-Time Control Protocol)라는 제어 프로토콜과 쌍으로 구성된다. RTP가 실제 데이터를 전달하는 역할을 하는 반면에, RTCP는 스트림의 성능이나 품질을 감시하기 위하여 모든 참가자에서 보고를 하는 역할을 한다. RTP와 RTCP는 RFC-1888과 RFC-1889에 각각 정의되어 있다.

RTP는 UDP와 함께 전송 계층의 프로토콜에 속하는 것으로 볼 수 있지만, 실제 사용할 때는 응용 계층에 더 가깝다. 그러므로, RTP는 실제 별도의 프로토콜로서 구현되는 것이 아니라, 응용 프로그램의 일부로서 구현이 된다. 즉, 응용 프로그램을 작성할 때, UDP를 사용하는 RTP 코드를 직접 포함시키거나, 라이브러리 형태를 호출하여 활용하고 있다.

RTP가 실시간 데이터 전송을 위한 프로토콜임에도 불구하고 현재의 인터넷에서 동작하므로, 실제 실시간으로 전송을 책임지거나 서비스 품질을 보장해주는 방안을 제공하지는 않는다. 다만, 송신자가 RTP 형식의 패킷을 만들어 UDP를 통해서 수신자까지 전달하며, RTP 패킷에 포함된 정보를 사용하여 수신자가 실시간성을 최대한 제공하도록 처리하는 것이다. 즉, RTP는 하부 망에서 자원 예약이나 서비스 품질을 제공하지 않는 인터넷과 같은 망에서 멀티미디어의 시간 관계를 고려하기 위한 비연결형 최선노력형 프로토콜이다. RTP는 일대일 전송뿐만 아니라 멀티캐스트 전송을 사용하도록 되어 있다.

그림 2은 RTP 패킷 형식을 나타내고 있다.

현재의 버전은 2이며, 페이로드 형태(PTYPE)는 데이터의 종류를 나타낸다. 특히, 시간 표시(Time

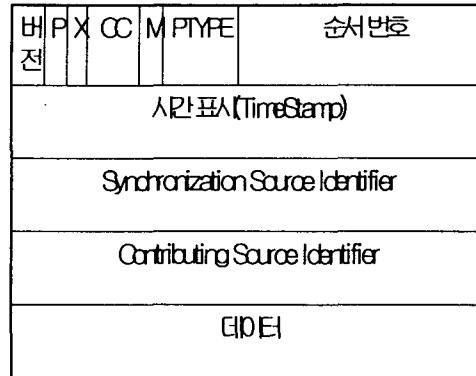


그림 1 RTP 프로토콜 형식

Stamp)는 패킷의 시간 동기화를 위해 필요하다. 패킷의 생성 시간을 기록하며, 단위는 미디어의 종류에 따라 달리 정한다. 예를 들어, 오디오의 경우에는 추출률(sampling rate)을 단위로 그 값을 기록한다. 수신측에서는 이 시간 스탬프를 사용하여 앞 뒤 패킷 스트림간의 간격이나, 여러 멀티미디어 스트림 사이의 시간 관계(예, 오디오와 비디오)를 조정할 수 있다. SSRC(Synchronization Source ID)는 세션을 구별하기 위한 유일한 ID이며, 임의의 숫자를 사용하여 송신측에서 설정한다. CSRC(Contribution Source ID)는 패킷의 내용에 기여하는 SSRC를 기록한다.

RTP에서는 송신자와 수신자 사이의 중간 노드를 두 가지 형태로 고려하는데, 혼합기(mixer)와 변환기(translator)이다. 혼합기는 여러 경로를 통해서 오는 멀티미디어 스트림이 합쳐져서 새 패킷으로 전달되는 곳을 말하며, 합쳐진 패킷의 SSRC를 새로이 부여한다. 이 때, 결합되기 전의 원래 근원지의 SSRC ID를 CSRC에 더한다. 만일, 여러 근원지로부터 도착한 패킷이 동기가 맞지 않는다면 혼합기에서 이들의 시간 관계를 변경한다. 혼합기의 예로는, 멀티미디어 회의 시스템에서 여러 사람이 말하는 오디오 정보를 합하여 전달하는 것을 들 수 있다. 변환기는 패킷의 내용을 변화시키는 역할을 하는 것으로, 코딩 변환기 등이 그 예이다.

RTCP는 전송 패킷 수, 손실 패킷 수, 지터 등을 내용에 포함시킬 수 있으며, 이 정보를 어떻게 활용할 지는 표준에서 명시하지 않고 응용 프로그램 개발자가 임의로 결정할 수 있다. 예를 들어, 손실 패킷 수를 사용하여 송신자가 전송률을 조절하거나, 지터를 사용하여 버퍼링을 하는 등의 활용이 가능

하다.

그림 2는(은) RTCP의 패킷 형식을 나타낸다.

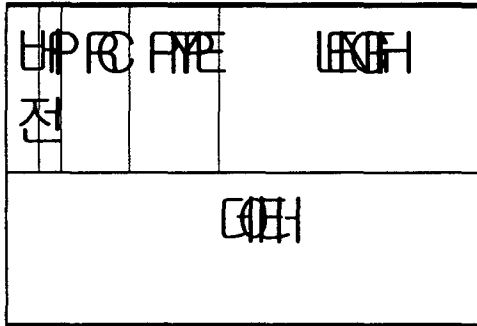


그림 2 RTCP 패킷 형식

현재의 버전은 2이다. 보고서 수(Report Count)는 포함될 보고서의 개수를 나타내며, 길이(LENGTH)는 헤더를 포함한 전체 패킷의 길이이다. 패킷 종류(Packet Type)에는 다음에 나타날 보고서의 형식과 내용을 결정한다.

보고서 종류에는 송신자 보고서, 수신자 보고서, 근원지 설명, Goodbye, 응용의 5가지가 있다.

3. JMF

JMF는 Java에서 멀티미디어 환경을 제공하기 위한 API로서, 기본 Java 환경에 포함되어 있지 않고, 별도로 설치한다. JMF는 1.x 버전을 거쳐 현재는 2.1.1 버전이 나와있다[2].

JMF의 기능은 javax.media.* 패키지에 정의되어 있으며, 프로그램에서 import하여 사용한다[5].

JMF에는 여러 클래스가 포함되어 있으며, 이 중에 멀티미디어 파일을 재생하는 클래스로 Player가 중요한 부분이다. 그리고 JMF 2.0부터는 실시간 전송 프로토콜로서 자리잡아가고 있는 RTP를 구현하여 포함을 시킴으로서, 본 논문과 같이 실시간 멀티미디어 전송이 요구되는 응용에서 아주 유용하게 활용될 수 있다.

JMF에서 RTP에 관련된 API는 javax.media.rtp, javax.media.rtp.event, javax.media.rtp.rtp의 패키지에 정의되어 있다.

JMF RTP를 구성하는 요소로는 세션 관리자(Session Manager), 데이터 소스(Data Source), 프로세서(Processor), 데이터 싱크(Data Sink)가 있다.

세션 관리자는 세션을 통제하고 세션에 참가한

참가자(Participant)에 대한 추적과 관리, 전송되는 스트림의 관리를 수행하는 등의 가장 핵심적인 기능을 담당한다(javax.media.Manager).

데이터 소스는 저장된 멀티미디어 파일이나 캡처되어 들어오는 데이터를 세션 관리자를 통하여 RTP로 보내기 위한 부분이다. (javax.media.protocol.DataSource).

데이터 싱크는 입력된 데이터를 저장 장치에 저장하기 위한 부분이다(javax.media.DataSink).

프로세서는 데이터의 내용을 변화시키거나 필요한 처리를 하는 부분이다(javax.media.Processor).

RTP에 관련된 사건(event) 처리에 관한 부분은 javax.media.rtp.event에 정의되어 있으며, 다음과 같은 것이 있다. SessionListener는 세션의 상태 변화를 감시한다. SendStreamListener는 전송되는 스트림의 상태 변화를 감시한다. ReceiveStreamListener는 수신되는 스트림의 상태 변화를 감시한다. RemoteListener는 상대방 참가자로부터 수신되는 사건을 처리한다.

4. 비디오 강의 시스템 구현

본 논문에서는 생중계나 녹화 상영과 같이 동영상으로 이루어진 강의 내용을 세미나실이나 컴퓨터 실습실을 통하여 전달하기 위한 것이다.

그림 3에 나타난 것처럼 학교내의 강의실 환경은 일반적인 인터넷에서 동영상을 방송하는 환경과는 다소 다른 특수한 특징을 가지고 있다.

첫째, 클라이언트 호스트가 일정한 위치에 집중되어 있고, 동시에 동일한 동영상이 전송된다. 즉, 실습실별로 수십 대의 컴퓨터가 LAN에 집중적으로 연결되어 있으므로, 멀티캐스트 전송이나 서브네트워킹으로 방송을 하면 트래픽을 크게 줄일 수 있다. VOD 서비스처럼 분산되어 있는 여러 사용자 호스트에 동영상을 각기 원하는 시간에 제공하려면 시간대 별로 다른 사용자는 별도의 사용자와 마찬가지로 대역폭을 차지하므로 금방 네트워크 전체의 대역폭이 부족한 상태가 올 것이다.

둘째, 실습실 혹은 건물별로 서브네트워크 구성이 효율적으로 되어 있으면, 실습실에서 비교적 많은 대역폭을 차지하는 동영상이 강의에 사용되더라도 다른 서브네트워크에는 트래픽 부담을 주지 않으므로 전체 캠퍼스내의 통신망의 효율적인 사용이 가능하다.

셋째, 학교 내에서는 여러 개의 컴퓨터를 갖춘 실

습실이 필요하다. 이 때, 실습실의 규모를 적절하게 결정하는 것이 학교 전체 실습실 사용 효율에 영향을 미친다.

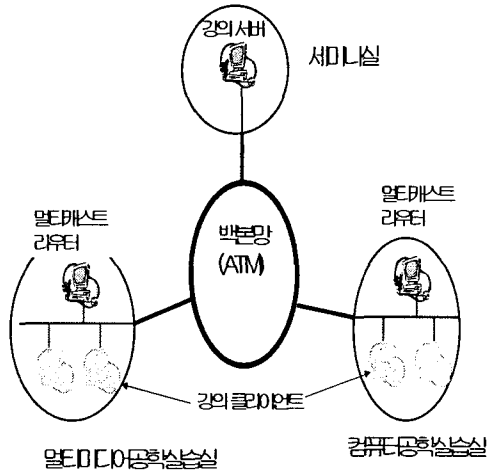


그림 3 강의 시스템 동작 환경

예를 들어보자. 40명을 기준으로 한 소형 실습실 2개와 80명을 기준으로 대형 실습실을 1개를 두 개의 학과에서 공유로 사용한다고 해보자. 학과에 따라 수업 인원이 다양하므로, 60여명의 인원이 수업을 받는다면, 소형 실습실 2개를 사용하거나 대형 실습실 1개를 사용할 수 있다. 어쩔 수 없이 대형 실습실을 소수의 인원이 사용할 때는 실습실의 활용도가 낮아진다.

소형 실습실을 여러 개 갖추수록 실습실 활용도가 높아지나 분반이 되어 강사의 수와 강의 시간이 늘어날 것이며, 반대로 대형 실습실을 확충하는 데는 많은 투자 비용에 소요되는데 비하여 실습 기자재의 활용도가 낮을 것이다.

그림 3에서는, 세미나실에서 동영상 강의 시스템의 서버를 사용하여 동영상을 교내에 멀티캐스트로 전송하는 예를 나타내었으며, 강의 서버는 각 실습실에도 설치가 되어 있으므로, 각 실습실에서의 강의가 마찬가지로 다른 강의실이나 실습실로 중계가 가능하다. 이 때, 수강 인원 에 따라 동영상 강의 시스템을 통하여 강의실이나 세미나실의 활용을 적절하게 확장시킬 수 있다.

사용한 장비는 MS-Windows 계열의 운영체제의 PC를 다수 사용하고 있으나, SUN Solaris를 비롯한 워크스테이션에서도 동작이 가능하며, 소프트웨어

개발은 JDK(Java Development Toolkit) 1.3.1, JMF API 2.0 환경에서 이루어졌다.

전체 시스템의 동작은 RTP를 사용하여 동영상을 전송하는 하나의 미디어 서버와 다수의 클라이언트로 이루어진다. 서버와 클라이언트 프로그램의 동작 흐름은 그림 4와 같이 이루어진다.

생중계인 경우에는 동영상 카메라와 마이크를 통해서 입력되는 신호를 추출하여 데이터 소스로 보내며, 녹화 중계인 경우에는 저장된 동영상 파일로부터 재생한 데이터를 데이터 소스로 보내어 사용한다. 이 때 동영상의 형식은 미리 정의되어 있는 JPEG/RTP 형식으로 하였다. 데이터 소스, 프로세서, 세션 관리자, 재생기 등은 JMF에서 정의되어 있는 클래스를 사용하여 생성한 객체이다.

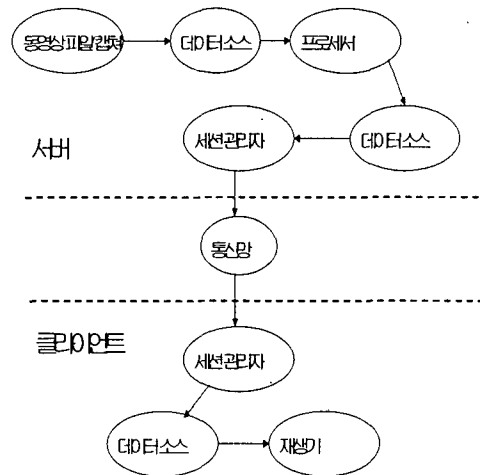


그림 4 동영상 강의 시스템의 소프트웨어 구성

그림 5는 강의 시스템에서 서버 프로그램의 동작

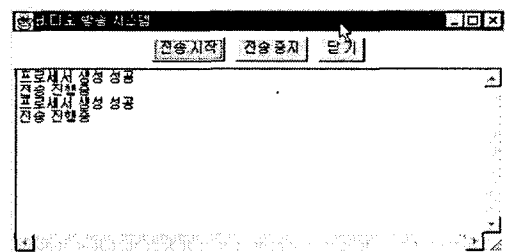


그림 5 비디오 서버 프로그램의 동작 화면

화면을 보여주고 있다. 이 예에서는 MPEG 형식으로 저장되어 있는 동영상 파일을 전송하였다.

그림 6은 클라이언트 프로그램의 동작을 보여주고 있으며, 클라이언트 프로그램을 동작시켜 앞에서 동작시킨 강의 서버로 접속하여 동영상을 재생하고 있는 장면이다.

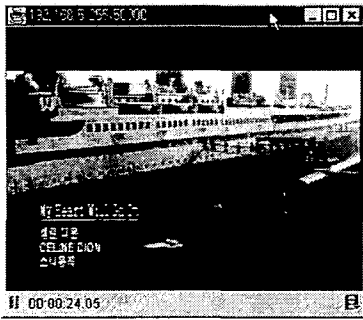


그림 6 동영상 시스템 클라이언트의 동작 화면

5. 결론

본 논문에서는 캠퍼스내의 인터넷 환경에서 Java 개발 툴과 JMF API를 사용하여 동영상 강의 시스템을 구현하였다.

JMF는 Java 환경에서 멀티미디어 처리를 위하여 개발되었으며, 최근 버전에서는 RTP를 채택하여 실시간 다중 전송 기능을 제공하고 있다.

강의 시스템의 구현이 아직 진행 중에 있으며, 실험은 일차적으로 유니캐스트 연결에서 서버와 다수의 클라이언트간에 이루어졌다.

실험 결과 서버로부터 클라이언트까지의 비디오 전송이 성공적으로 이루어졌으나, 서버에 접속하는 클라이언트의 수가 증가함에 따라 비디오의 재생속도가 급격하게 느려져 충분한 품질을 제공하지 못하였다. 이러한 현상은 MBONE 환경에서 멀티캐스트 전송을 하면 향상될 것으로 예상된다.

만일, 전체 통신망의 대역폭 사용량이 많은 시간대에는 동영상 강의의 전송은 더욱 제약을 받게 될 것이다. 따라서, 전송로의 대역폭 사정에 따라 전송할 비디오의 데이터량을 제어함으로써 다소 화질이 저하되더라도 최소한의 재생률을 제공하기 위한 처리 부분이 필요하다.

참고 문헌

1 D. Comer and D. Stevens, "Internetworking with TCP/IP Vol.3", Prentice-hall, 2001.

2 "Java Media Framework API", <http://java.sun.com/products/java-media/jmf/index.html>.

3 "JMF 2.1.1", <http://java.sun.com/products/java-media/jmf/2.1.1/download.html>.

4 "JMF 2.0 Documentation", <http://java.sun.com/products/java-media/jmf/2.1.1/specdownload.html>.

5 "Java Media Framework API Guide", <http://java.sun.com/products/java-media/jmf/2.1.1/guide/JMFTOC.html>.

6 A. Kantarci and T. Tunali, "A Video Streaming Application on the Internet", ADVIS 2000, LNCS 1909, pp.275-284, Springer-Verlag Berlin Heidelberg 2000.

배 시 규



1986년 경북대학교 전자공학과(학사)

1993년 경북대학교 대학원

컴퓨터공학과 (공학석사)

1998년 경북대학교 대학원

컴퓨터공학과 (공학박사)

1986~1991년 삼성전자(주) 종합연구소

1995~현재 동양대학교 컴퓨터공학부 조교수

관심분야 : 컴퓨터네트워크, 인터넷, 멀티미디어 통신