

RTSP에 기반한 MPEG-4 스트리밍 시스템 구현

이상현* 박병준* 차호정**

*광운대학교 컴퓨터학과

**연세대학교 컴퓨터학과

{shlee,bjpark}@cs.kwangwoon.ac.kr hjcha@yonsei.ac.kr

Implementation of a RTSP-based MPEG-4 Streaming System

Sanghyun Lee* Byungjoon Park* Hojung Cha**

*Dept. of Computer Science, Kwangwoon University

**Dept. of Computer Science, Yonsei University

요 약

본 논문은 RTSP에 기반한 MPEG-4 스트리밍 시스템의 구현에 대해 기술한다. 구현된 스트리밍 서버는 하나의 클라이언트 세션에서 다수의 미디어 스트림을 전송할 수 있다. 스트리밍 클라이언트는 MPEG-4 전송계층인 DMIF를 RTSP로 대체하여 구현하였고 기존의 파일 재생시 뿐만 아니라 스트리밍시에도 MPEG-4의 특징인 사용자 상호작용 기능을 가능하게 한다.

1. 서론

최근 컴퓨터 하드웨어와 네트워크 기술의 발전으로 인터넷을 이용한 다양한 형태의 비디오 서비스들이 등장하게 되었다. 이러한 요구중 가장 특징적인 것은 단방향의 방송이 아닌 양방향성 방송, 즉 대화형 방송에 대한 요구이다. 동영상에 관련된 국제 표준중 이러한 사용자의 요구를 잘 수용할 수 있는 표준의 하나가 MPEG-4이다[1].

MPEG-4는 최근에 제정된 멀티미디어 국제 표준으로써 다양한 미디어의 지원과 사용자 상호작용 기능으로 MPEG-1/2를 대체할 것으로 보인다. MPEG-4는 시스템, 비디오, 오디오, Conformance, 소프트웨어, DMIF(Delivery Multimedia Integration Framework)의 6개 하위 그룹으로 구성되어 있다. 본 논문에서 구현한 스트리밍 시스템은 첫 번째 하위 그룹인 시스템 그룹과 관련이 있다. MPEG-4 시스템은 하나의 장면을 구성하는 미디어 요소들 즉, 비디오, 오디오, 이미지, 텍스트들을 객체화하고 부호화하며 전송한다. 또한 개별적인 객체들간의 시공간적인 관계를 기술하기 위해 BIFS(Binary Format for Scene)라는 장면기술 언어를 사용하고 있다[2].

MPEG-4 시스템 그룹에서는 MPEG-4 파일을 생성하기 위한 멀티플렉싱 도구를 제공하고 있다. 멀티플렉서에 의해 생성된 파일은 사용자 상호작용 문제등 여러점을 고려하면 스트리밍시에는 부적합하다. 본 논문에서는 멀티플렉싱된 하나의 파일을 스트리밍에 사용하는 대신 MPEG-4 파일을 구성하는 각각의 미디어 데이터를 MPEG-4 파일 타입으로 변환하여 스트리밍하는 방식을 채택하여 구현하였다.

스트리밍 서버는 각각의 미디어 데이터를 사용자의 요청에 따라 전송한다. 서버는 하나의 클라이언트 세션에서 요청하는 다수의 미디어 데이터를 동시에 전송할수 있도록 구현되었으며 표준 스트리밍 제어 프로토콜인 RTSP(Real Time Streaming Protocol)를 사용한다[3]. RTSP는 인터넷상에서 스트리밍 데이터를 제어하기 위한 프로토콜이며 플레이어의 여러 가지 명령 방식을 쉽게 원격지의 미디어 데이터에 대해 적용할 수 있다.

클라이언트는 스트리밍 재생시에도 MPEG-4에서 제공하는 모든 사용자 상호작용을 지원하고 있다. 사용자는 화면을 구성하는 특정객체에 이벤트를 발생시켜 새로운 장면을 구성하거나 원하는 위치로 이동시킬 수 있다.

본문의 구성은 다음과 같다. 2절에서는 스트리밍 시스템의 개요, 스트리밍을 위한 미디어 데이터의 멀티플렉싱 과정, 스트리밍 서버와 클라이언트의 구조, 실행 결과에 대하여 기술하고 3절에서 결론을 맺는다.

2. MPEG-4 스트리밍 시스템

다음은 전체적인 스트리밍 시스템의 구조와 사용자의 요청에 따라 MPEG-4 미디어 데이터를 전송할 수 있는 서버의 구조를 설명한다. 그리고 개별적으로 전송되는 미디어 데이터를 수신하여 재생할 수 있는 클라이언트의 구조를 기술한다.

2.1 스트리밍 시스템 개요

MPEG-4 시스템 그룹에서는 MPEG-4 파일을 생성할 수 있는 멀티플렉서를 제공한다. 기존의 멀티플렉서를 사용하여 파일을 생성할 경우 BIFS 스크립트 파일에 지정된 재생 시간대로 파일이 생성되지 않는다. 파일은 단순히 절대적인 타임스탬프 값과 멀티플렉서 스크립트 파일에 기술된 순서대로 저장된다. 이러한 파일 형태는 사용자가 하나의 MPEG-4 파일

* 본 연구는 정보통신부에서 지원하는 대학기초연구지원사업으로 수행하였음 (과제번호 : 2001-076-3)

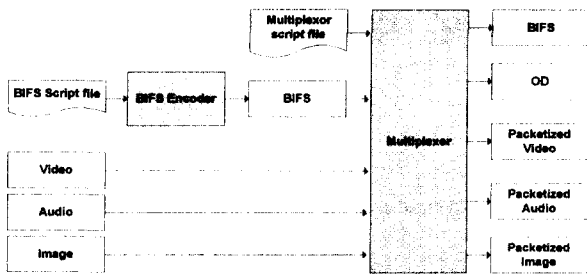


그림 1 멀티플렉싱 과정

내에서 장면 전환등으로 새로운 스트림을 요청할 경우 파일내에서의 점프등 어려운 점이 있다. 단순히 스크립트 파일에 기술된 순서대로 저장되기 때문에 사용자가 재생하고자 하는 것과는 무관하게 저장될 수 있다. 본 논문에서는 이러한 문제를 해결하기 위해 MPEG-4 파일을 구성하는 각각의 미디어 데이터들을 개별적인 MPEG-4 파일로 생성하여 이를 스트리밍시에 사용한다. 또한 장면 구성과 미디어 데이터에 대한 정보를 담고 있는 BIFS와 OD 데이터도 개별적인 파일로 생성한다. 그림 1은 이러한 멀티플렉싱 과정을 보여준다.

스트리밍 서버는 개별적인 미디어 데이터를 사용자의 요청에 따라 전송한다. 클라이언트는 각각의 미디어 스트림을 서로 다른 포트로 전송받아 BIFS에 의해 생성된 장면 구성 트리에 맞게 화면을 구성한다. 사용자 상호작용에 의해 새로운 스트림이 요청될 경우 서버로 이를 요청하여 새로운 장면을 구성한다. 그림 2는 이와 같은 스트리밍 시스템의 전체적인 구조를 보여준다.

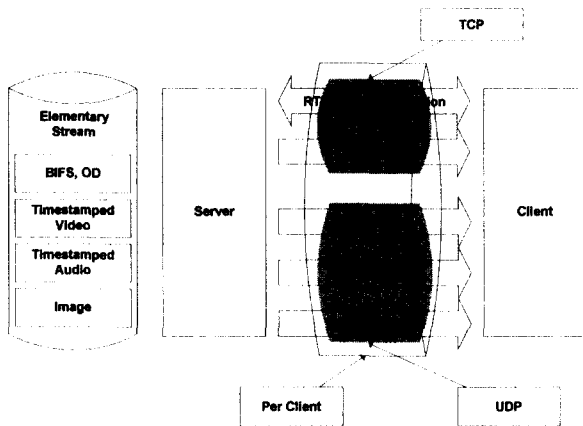


그림 2 MPEG-4 스트리밍 시스템 구조

2.2 스트리밍 서버

스트리밍 서버는 사용자의 요청에 따라 각각의 미디어 데이터를 전송한다. 서버는 RTSP 프로토콜을 사용하여 데이터와 제어 메시지의 송수신을 분리한다. 일반적인 VOD 서버와는 달리 하나의 클라이언트 세션당 전송해야 할 미디어 데이터의 수가 여러개가 될 수 있으므로 이를 수용할 수 있어야 한다. 논문에서 구현한 서버의 구조는 그림 3과 같다. Admit Manager

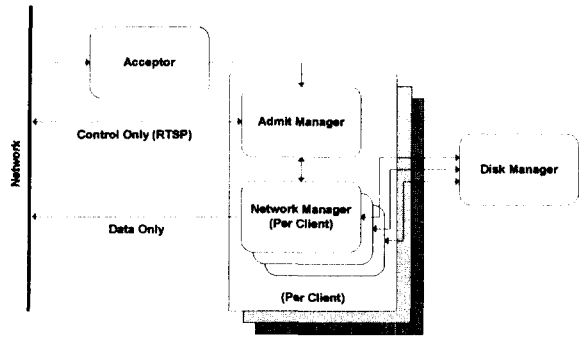


그림 3 스트리밍 서버의 구조

는 제어 메시지만을 처리하고 Network Manager는 미디어 데이터의 전송만을 담당하고 있다. 하나의 클라이언트마다 하나의 Admit Manager와 여러개의 Network Manager를 할당함으로써 제어 메시지와 데이터의 송수신을 분리하고 한 세션에 여러개 스트림의 전송을 가능하게 하였다. Acceptor는 클라이언트로부터의 접속을 받아들여 그 요구를 Admit Manager에게 넘겨준다. Admit Manager는 RTSP 프로토콜을 사용한 제어 메시지의 처리 외에도 새로 접속한 클라이언트의 수용 여부를 결정한다. Network Manager는 Admit Manager에 의해 생성되며 실제 미디어 데이터의 송신을 담당한다. 하나의 세션에서 여러개의 미디어 데이터를 전송하기 위해 Network Manager는 클라이언트가 새로운 스트림을 요청 할 때마다 새로 생성된다. Disk Manager는 Network Manager의 요청에 따라 전송할 데이터를 파일에서 일정 크기만큼 읽어들이고, 이때 효율적인 파일 입출력을 위하여 비동기 입출력 방식을 사용하고 있다.

2.3 스트리밍 클라이언트

스트리밍 클라이언트는 MPEG-4 하위 '소프트웨어' 그룹에서 제공하고 있는 IM-1 버전 3.6 플레이어를 사용하였다[4]. 기존의 플레이어를 수정하여 네트워크를 통해 스트리밍 재생이 가능하며 사용자와의 상호작용도 가능하도록 하였다.

MPEG-4에서는 데이터의 전송, 제어와 관련하여 DMIF 라는 별도의 계층을 두어 전달하도록 하고 있으며 이에 대한 연구도 활발히 진행중이다[5]. 이러한 맥락에서 DMIF를 기존의 프로토콜로 대체 구현하려는 연구가 진행중인데 대표적인 예가 DMIF를 RTSP와 RTP를 사용하여 구현하는 것이다. 본 논문에서는 이러한 제안을 IM-1 플레이어를 바탕으로 실제 구현하였다.

DMIF는 사용자의 요청에 따라 적합한 서비스 모듈을 적재시킨다. 현재 제공되는 서비스 모듈은 로컬 컴퓨터내에서 파일을 재생 할 수 있는 파일 서비스 모듈만을 제공하고 있다. 본 논문에서는 스트리밍 재생이 가능하도록 하는 서비스 모듈을 구현하였고 그 이름을 RTSP 서비스라고 명명하였다. 그림 4는 DMIF 내에서 파일 서비스와 RTSP 서비스와의 관계를 보여주고 있다. DMIF는 사용자가 입력한 URL에 근거하여 파일 재생에 대한 요청인지 스트리밍 재생에 대한 요청인지를 구분한다. 스트리밍 재생의 경우 RTSP URL을 사용하므로 이를 구분할 수 있다. DMIF 내의 RTSP 서비스 모듈은 데이터의 전송 및

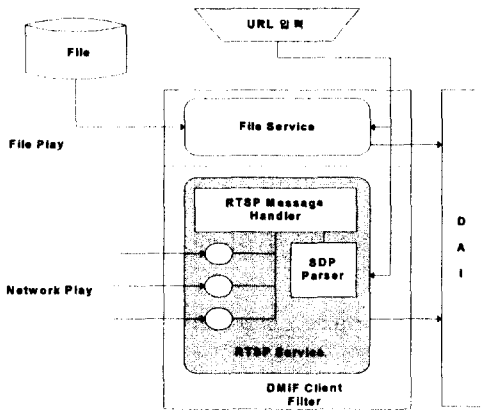


그림 4 RTSP 서비스 모듈

RTSP 프로토콜을 사용하여 미디어 데이터의 흐름제어를 담당하고 있다. RTSP 메시지 처리기는 표준에서 정의하고 있는 DESCRIBE, SETUP, PLAY, TEARDOWN, SET_PARAM 메시지들을 처리한다. SDP 파서는 세션 초기의 DESCRIBE 메시지에 대한 응답과 함께 전송된 SDP 데이터를 파싱한다[6]. RTSP 메시지 핸들러에 의해 생성된 데이터 수신 쓰레드들은 요청한 미디어 데이터를 서버로부터 받아 들인다.

2.4 스트리밍시 사용자 상호작용

MPEG-4의 가장 커다란 특징중 하나는 사용자와의 상호작용을 지원한다는 점이다. 사용자가 화면상의 특정 객체를 클릭했을 때 새로운 화면으로 전환된다든지 특정 객체의 위치를 임의로 옮긴다든지 하는 것들이 그 예 일수 있다.

MPEG-4에서 제공하는 사용자 상호작용은 BIFS의 관점에서 보면 삽입, 삭제, 전환, 장면 전환 네가지이다. 본 논문에서 수정한 플레이어에서는 스트리밍시에도 이러한 사용자 상호작용이 모두 가능하다. 특정 객체의 위치를 바꿔주는 것과 같이 스트리밍시에도 서버와의 상호작용이 필요없는 사용자 상호작용은 문제가 되지 않는다. 하지만 사용자의 이벤트 발생에 의하여 새로운 스트림을 요구할 경우에는 서버와의 통신이 필요하게 된다. 보통 파일 재생의 경우 이벤트 발생에 의하여 새로운 스트림을 요구할 경우에는 새로 요구될 스트림을 Inline 이라고 하는 BIFS 객체로 정의하고 이 스트림의 위치를 BIFS 스트림트내에 URL 값에 지정해 준다. 논문에서는 이 URL 값에 파일의 위치를 명시하지 않고 RTSP URL을 명시함으로써 새로운 스트림 요청시 DMIF에 의해 구현된 RTSP 서비스가 적재될 수 있도록 하였다. 적재된 RTSP 서비스는 클라이언트가 맨 처음 세션을 초기화 할 때의 수행과정을 반복하여 새로운 스트림을 재생 하게 된다.

2.5 실행 결과

구현한 스트리밍 서버와 클라이언트는 Microsoft Visual C++ 6.0을 사용하였고 Windows 2000 환경에서 실행하였다.

아래 그림은 구현된 플레이어의 실행 화면 모습이다. 그림 6에서와 같이 특정 이미지에 대해 MouseOver 이벤트가 발생하면 텍스트가 출력되는 것을 볼 수 있다. 그림 7은 이미지를

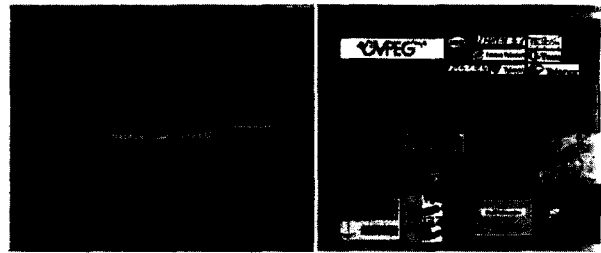


그림 5 RTSP URL 입력

그림 6 초기화면

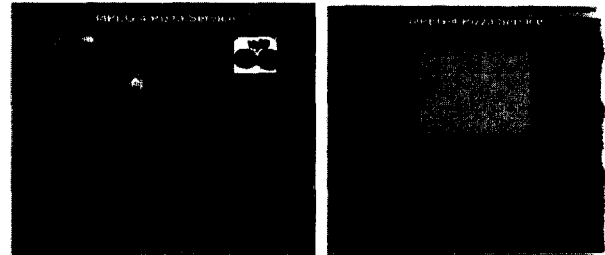


그림 7 사용자 상호작용에 의한 장면 전환 (1)

그림 8 사용자 상호작용에 의한 장면 전환 (2)

클릭했을 때 서버로부터 새로운 스트림을 전송받아 장면이 재구성된 모습이다. 그림 8은 그림 7에서 가운데 이미지를 클릭하여 서버로부터 비디오를 전송 받으며 재생하고 있는 장면이다.

3. 결론

본 논문에서는 RTSP에 기반한 MPEG-4 스트리밍 서버와 클라이언트의 구현에 대하여 설명하였다. 클라이언트의 전송계층을 RTSP로 구현함으로써 스트리밍시에도 사용자 상호작용을 문제없이 지원할 수 있다는 것을 확인하였다.

향후 연구과제로는 저작도구 구현과 스트리밍시 멀티플렉싱을 이용한 클라이언트 컨백션의 감소등이 있다.

참고문헌

[1] Atul Puri, Tsuhan Chen, *Multimedia Systems, Standards, and Networks*, Marcel Dekker, Inc., 2000.
 [2] ISO/IEC FCD 14496-1, Part 1: *Systems*, ISO/IEC/JTC1/SC29/WG11 N2201, May 1998.
 [3] H. Schulzrinne, A. Rao, R. Lanphier, RFC2326, *Real Time Streaming Protocol(RTSP)*, April 1998.
 [4] <http://drogo.cse.stet.it/ufv/im1>
 [5] ISO/IEC CD 14496-6, Part 6 : *Delivery Multimedia Integration Framework*, May 1998.
 [6] M. Handley, V. Jacobson, RFC2327, *Session Description Protocol(SDP)*, April 1998.