

# MPEG-4 스트리밍을 위한 서버 디멀티플렉싱 기법

박 지현, 김 상욱  
경북대학교 컴퓨터학과

## The De-multiplexing Technique on Server Side for MPEG-4 Streaming

Ji-Hyun Park, Sang-Wook Kim  
Dept. of Computer Science, Kyungpook National University  
E-mail : jhpak@woorisol.knu.ac.kr, swkim@cs.knu.ac.kr

### 요 약

본 논문은 MPEG-4 스트리밍을 위한 서버 디멀티플렉싱 기법을 제안한다. 일반적인 스트리밍 서비스에서 클라이언트가 수행하는 디멀티플렉싱을 서버에서 수행한다. 이 기법은 클라이언트의 부하를 감소시킨다. 또한, 다수의 클라이언트에게 스트리밍 서비스를 제공할 경우 데이터 중복 전송에 대한 네트워크 자원 낭비를 막고, 서버 시스템의 부하를 줄일 수 있다. 그러므로, 서버 디멀티플렉싱 기법을 사용할 경우, 기존의 PC 환경에서 뿐 만 아니라 무선환경과 같이 자원이 부족한 환경에서 동작하는 멀티미디어 재생기 개발에 응용될 수 있다.

### 1. 서론

MPEG-4[1-3] 스트리밍은 데이터 전송에서 네트워크의 영향을 줄이고, 클라이언트의 부하를 감소시킴으로써 최종 사용자에게 고품질의 MPEG-4 재생을 지원해야한다[4-5].

MPEG-4 스트리밍에서 재생기는 디코딩 방법이나 프리젠테이션 방법을 개선함으로써 전송 받은 스트림의 처리 시간을 감소시킬 수 있다. 그러나 이러한 방법은 클라이언트의 부하를 감소시키기 어렵다. 그러므로 서버의 변형을 이용한 스트리밍 기법이 요구된다.

서버 디멀티플렉싱 기법은 클라이언트의 부하를 감소시킬 수 있도록 클라이언트에서 이루어지는 디멀티플렉싱을 서버에서 수행한다. 이 기법은 다수의 클라이언트에게 스트리밍 서비스를 제공할 때 발생할 수 있는 데이터 중복 전송으로 인한 네트워크 자원 낭비를 막고, 시스템 부하를 줄일 수 있다.

미디어 스트리밍 서비스에 대한 관련 연구로 Real

Media Technology[6], Windows Media Technology[7], Quick Time Technology[8] 등이 있다. 이러한 솔루션은 윈도우나 리눅스 환경에서의 스트리밍을 지원하지만, 자체의 멀티플렉싱 기술을 사용하여 동영상 콘텐츠를 제공한다. 각 솔루션의 콘텐츠 생성기는 Real Media 파일[6], ASF 파일[7], Quick Time 파일[8]을 생성하고 각각의 미디어 재생기에 전달한다. 멀티플렉싱 된 스트림은 각각의 디멀티플렉싱 방법으로 분리되고, 미디어 재생기는 이를 재생한다.

이와 같은 방법은 재생기에 의존적인 스트리밍 서비스를 제공하고, 클라이언트의 부하를 감소시키기 어렵다는 문제점이 있다.

### 2. MPEG-4 스트리밍 환경

본 장에서는 MPEG-4 스트리밍을 위한 서버와 클라

이언트 구조를 제시하고 MPEG-4 프리젠테이션에 관하여 기술한다.

그림 2-1은 스트리밍 환경이다.

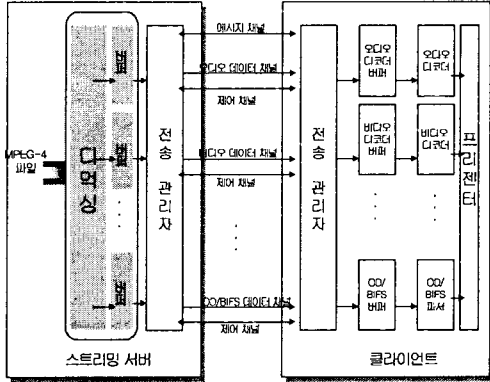


그림 2-1 MPEG-4 스트리밍 환경

## 2.1 서버 구조

스트리밍 서버는 전송 관리자를 통하여 전달된 미디어 재생기의 메시지를 처리하고, 미디어 스트림을 전송 관리자에게 전달하는 기능을 수행한다. 그리고 MPEG-1, MP3, MPEG-4와 같은 미디어 스트림을 지원한다.

그림 2-2는 서버 구조이다.

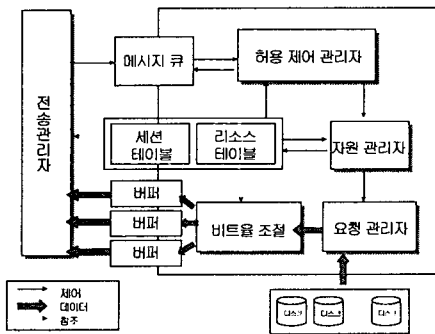


그림 2-2 서버 구조도

### ■ 전송관리자

서버 전송 관리자는 클라이언트의 요청에 대한 메시지를 분석하여 서버 응용프로그램에 메시지를 전달하고, 서버가 처리한 미디어 스트림을 클라이언트로 전

송한다.

제어 메시지 송수신 모듈, 데이터 전송 모듈, QoS(Quality of Service) 와 RTP/RTCP 처리 모듈로 구성되어 있다.

### ■ 허용 제어 관리자

세션을 설정, 관리하고 각각의 VCR 연산에 대하여 허용 제어를 수행한다. 세션 설정은 버퍼 및 디스크 자원을 할당하고 세션 테이블에 현재 세션의 상태를 기록한다.

### ■ 요청 관리자

세션에서 요구하는 데이터 양을 계산하고 버퍼링하여 전송 관리자에게 전달한다. 이때 MPEG-4 파일 포맷을 해석하여 비디오/오디오 등의 데이터를 분리하고 동기화 정보를 추출한다. 상호 배제 기능을 가지는 이중 버퍼를 사용하며, 비어있는 버퍼에서 필요한 양의 데이터를 파일시스템에게 요구한다.

### ■ 자원 관리자

세션 테이블, 리소스 테이블을 주기적으로 체크하여 갱신한다.

세션 테이블은 전송 관리자와 정보 공유가 필요한 정보를 관리하며 세션의 정보가 변경될 때마다 자원 관리자가 세션 테이블, 리소스 테이블을 변경한다.

### ■ 비트율 제어자

전송 관리자로부터 전달받은 QoS를 검사하여 QoS에 맞는 비디오/오디오 등의 데이터를 실시간으로 전송 관리자에게 전달한다.

## 2.2 재생기 구조

미디어 재생기는 MPEG-1, MP3, MPEG-4를 재생한다. 또한, 디스크에 저장된 파일, 실시간 스트리밍 파일, 그리고 멀티캐스팅/유니캐스팅 되는 스트림을 실시간으로 재생한다.

미디어 재생기의 구조는 그림 2-3과 같다.

### ■ 전송관리자

클라이언트 전송 관리자는 미디어 재생기가 스트리밍 서버에게 요구하는 메시지를 전달하고 서버로부터 전달받은 데이터를 버퍼에 저장한다. 또한 미디어 데이터에 대한 동기화, 순서제어, QoS 정보를 처리하는 모듈로 구성되어 있어 이에 해당되는 내용을 처리한

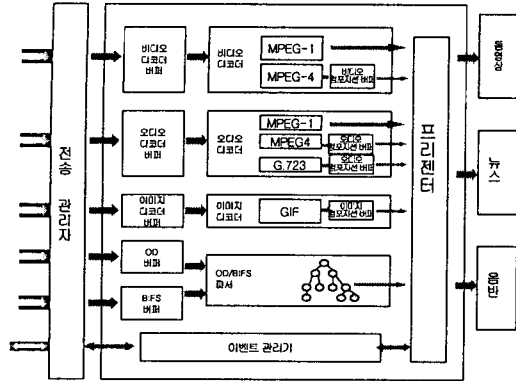


그림 2-3 미디어 재생기 구조

다. 클라이언트에 필요한 채널은 음성과 영상 등으로 나뉜 각각의 데이터 전송채널과 이에 해당하는 제어 정보 채널인 RTP채널로 이루어져 각각의 채널을 설정하게 된다.

■ 디코더/ 컴포지션 버퍼

서버에서 전송 받은 미디어를 저장하기 위해 버퍼가 필요하다. 버퍼는 원형 버퍼로 구현되며 원형 버퍼의 각 요소는 프레임이 된다. 각 프레임은 프리젠타가 읽기 위치와 쓰기 위치일 때 접근 제어를 할 수 있다.

■ BIFS/OD 파서

BIFS(Binary Format for Scene)는 장면을 구성하는 객체의 속성과 위치 정보를 갖고 있다. OD(Object Descriptor)는 객체의 데이터 정보를 갖고 있다.

BIFS/OD 파서는 BIFS/OD 데이터를 파싱하여 장면 그래프를 생성한다. 장면 그래프는 장면을 구성하는 객체의 속성과 위치정보, 데이터 정보를 갖는다.

■ 장면 프리젠타

장면 프리젠타는 오디오, 비디오, 이미지 등으로 구성된 장면을 화면에 재생한다. 장면 렌더러, 드로어블 노드 리스트, 드로잉 모듈로 구성되어 있다.

■ 이벤트 관리자

이벤트 관리기는 서버에서 '재생', '일시정지', '종료'와 같은 스트림 관련 이벤트를 처리한다. 이러한 처리의 실시간성을 높이기 위해 이벤트 처리루틴이 콜백으로 결과 값을 주고, 시스템의 안정성을 높인다.

### 3. 서버 디멀티플렉싱 기법

본 장에서는 MPEG-4 스트리밍을 위한 서버 디멀티플렉싱 기법을 위한 허용 제어 관리, 요청 관리, 전송 관리, 멀티캐스팅 관리에 대해 설명한다.

그리고, 서버 디멀티플렉싱 기법을 위한 서버 동작 흐름에 대해 설명한다.

#### 3.1 허용 제어 관리

허용 제어 관리자는 세션에 대한 정보가 변할 때마다 세션 테이블에 세션번호, 세션 상태를 기록하여, 자원 관리자가 세션 자원을 관리할 수 있도록 한다.

전송 관리자가 전달한 DAI(DMIF Application Interface) 메시지를 검사하여 요청 관리자에게 사용자 명령을 전달한다.

공유 버퍼를 생성하여 전송 관리자와 요청 관리자가 공유하도록 한다. 공유 버퍼는 동시에 읽기/쓰기를 방지하기 위해 뮤텍스를 사용한다.

#### 3.2 요청 관리

서버 디멀티플렉싱 기법에서 서버가 MPEG-4 스트림을 각각의 미디어로 분리하는 과정이 여기에서 수행된다.

요청 관리자는 허용 제어 관리자가 보낸 명령어를 받아 해당 명령에 대한 작업을 수행한다. 그리고 자신의 상태를 허용 제어 관리자에게 전달한다.

요청 관리자는 미디어 스트림에 대한 오디오/비디오 컨트롤 기능을 제공한다.

- (1) 스트림의 시작 및 종결기능
- (2) 스트림의 재생, 정지 및 계속 기능
- (3) 스트림의 임의 위치에서 재생 가능하게 하는 점프기능

MPEG4 파일 포맷 디코더는 MPEG-4 파일을 구성하는 각 미디어 객체로 분리하기 위해 하일 헤더를 파싱한다.

MPEG-4 파일 포맷은 다음과 같은 형식이다.

MPEG-4 파일헤더	비디오 데이터1	오디오 데이터1	이미지 데이터1	비디오 데이터2	...	오디오 데이터 N	이미지 데이터 N
-------------	----------	----------	----------	----------	-----	-----------	-----------

MPEG-4 파일 헤더에는 현재 MPEG-4 파일을 구성하고 있는 미디어가 몇 가지이며, 가변 크기를 가지고 한 파일에 섞여 있는 각 미디어 데이터를 파일에서 인덱싱하기 위한 정보와 각 미디어 데이터를 재생하기 위해 필요한 타임스탬프 정보가 들어있다.

파일 헤더 정보를 읽고 분석해야 RTP를 통해 전송될 오디오/비디오 데이터를 추출할 수 있다.

서버 디멀티플렉싱 기법에서 서버의 요청 관리자가 MPEG-4 헤더를 파싱하고, 그 정보를 이용해 미디어를 분리하여 버퍼에 저장한다.

### 3.3 전송 관리

전송 관리자는 허용 제어 관리자와 DAI 메시지를 이용하여 통신하고, 클라이언트 전송 관리자와 DNI 메시지를 이용하여 통신한다.

전송 관리자와 요청 관리자는 허용 관리자가 생성한 더블 버퍼를 공유하여, 요청 관리자가 더블 버퍼에 저장해둔 데이터를 얻고, 각 미디어에 적절한 채널을 생성하여 클라이언트로 전송한다.

### 3.4 멀티캐스팅 관리

다수의 클라이언트에게 스트리밍 서비스를 제공하기 위해 멀티캐스팅 관리가 필요하다. 미디어 데이터가 처음 요구되었을 경우, 멀티캐스팅 채널을 설정한다. 멀티캐스팅 채널이 설정된 미디어에 대해 특정 시점에 들어오는 클라이언트 요구에 대해서는 패칭으로 가능한지 여부를 검사한다.

그림 3-2는 패칭 구조이다.

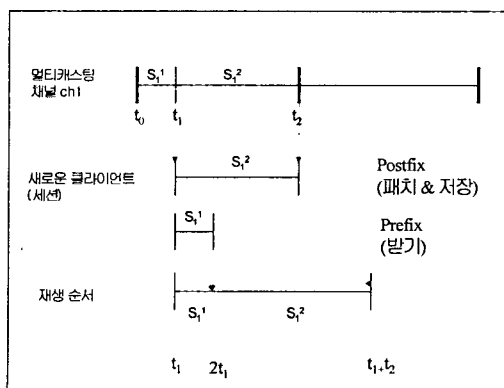


그림 3-2 패칭 구조

멀티캐스팅 되고 있는 미디어 데이터에 대해 t1시점에서 새로운 클라이언트의 세션 요청이 이루어진 경우 다음과 같이 처리한다.

(1) 기존에 존재하는 Ch1에 대해 t1 시점부터는 멀티캐스팅 채널에 대한 패칭 작업이 이루어져서 클라이언트의 로컬 디스크에 저장된다.

(2) t0부터 t1시점까지의 부족분  $S_1^1$ 에 대해서 별도의 채널을 열어 프리젠테이션 한다.

(3) Prefix에 대한 재생은  $2*t_1$ 시점에서 종료되고 채널이 해제되며, 그이후 패칭으로 디스크에 저장된  $S_1^2$ 가 재생된다.

(4) 멀티캐스팅 스트림에 대해서는 항상 t에 해당되는  $S_1^1$  만큼의 데이터 량이 클라이언트의 디스크에 저장된다. 저장되는 데이터의 양은 VCR 연산이 일어나지 않는 동안에는 계속되며, VCR 연산에 의해서 값이 늘거나 줄게 된다.

### 3.5 서버 동작 흐름

서버 디멀티플렉싱 기법을 이용한 스트리밍 서비스에서 허용 제어 관리자, 요청 관리자, 전송 관리자와 사이의 동작은 다음과 같다.

허용 제어 관리자는 서버 전송 관리자에게서 서비스 요청을 받고 세션을 설정한다. 더블버퍼를 생성하고 요청 관리자를 동작시킨다. 요청 관리자는 요구하는 데이터를 각 미디어 객체로 분리하여 공유 메모리에 저장한다. 전송 관리자는 공유 메모리에 있는 데이터를 클라이언트로 전송한다. 그림 3-3은 서버 디멀티플렉싱 기법 이용 시 서버의 전체적인 흐름이다.

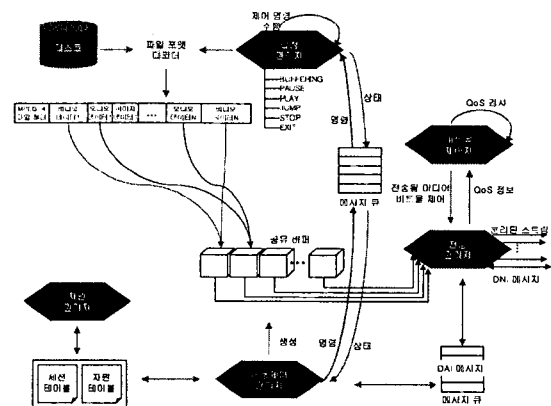


그림 3-3 서버의 전체적인 흐름

#### 4. 구현 및 성능평가

스트리밍 서버와 미디어 재생기는 리눅스 환경에서 동작한다.

본 논문에서 구현된 서버와 미디어 재생기는 C로 구현되며, 미디어 재생기는 Gtk+ 라이브러리를 이용하여 프리젠테이션 한다.

미디어 서버는 MPEG-4 스트림을 분리하고, 다수의 채널을 통해 재생기로 전송한다. 재생기는 전송 받은 스트림을 디코딩하여 프리젠테이션 한다.

그림 4-1은 서버 디멀티플렉싱 기법을 이용하여 MPEG-4 미디어 스트리밍 한 화면이다.



그림 4-1 MPEG-4 재생화면

클라이언트는 서버가 전송한 미디어 스트림을 디코딩하여 프리젠테이션 한다. 재생기에서 미디어 스트림을 처리하는데 사용되는 자원의 양과 부하를 줄이기 위해 서버 디멀티플렉싱 기법을 이용한다.

서버에서 디멀티플렉싱을 수행함으로써 클라이언트는 다중화 된 스트림을 저장하고, 처리하는데 사용되는 자원의 사용량을 줄일 수 있다.

그림 4-2는 클라이언트 디멀티플렉싱을 이용한 스트리밍 기법과 서버 디멀티플렉싱을 이용한 스트리밍 기법의 경우에 대해 각각 클라이언트의 메모리 사용량을 비교한 그래프이다.

그래프는 다중화 된 스트림의 임시 저장버퍼의 제거와 디멀티플렉싱을 수행에 필요한 자원 감소로 클라이언트의 전체 메모리 사용량이 감소됨을 보여준다.

또한, 클라이언트/서버 디멀티플렉싱 기법을 이용하여, 재생기가 스트림을 처리하여 프리젠테이션 하는데 걸리는 시간을 측정하였다.

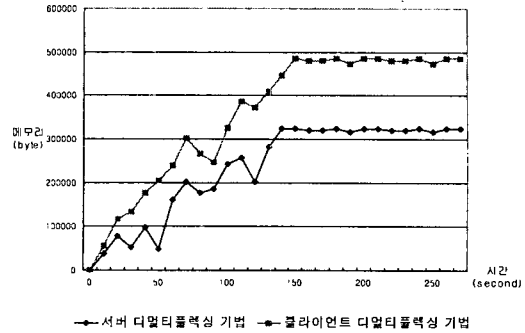


그림 4-2 메모리 사용량 비교

측정 결과, 서버에서 디멀티플렉싱 과정을 수행한 경우, 재생기에서 스트림을 재생하는데 걸리는 시간이 약 10% 감소되었다.

#### 5. 결론

본 논문은 MPEG-4 스트리밍을 위한 서버 디멀티플렉싱 기법을 제안한다. 스트리밍 서비스에서 클라이언트가 수행하는 디멀티플렉싱 과정을 서버에서 수행한다.

서버는 MPEG-4 스트림을 디멀티플렉싱하여 독립적인 미디어로 분리하고 각각의 버퍼에 저장한다. 전송 관리자는 미디어에 독립적인 채널을 설정하여 각 미디어에 최적화된 채널을 통해 데이터를 전송한다.

서버에서 디멀티플렉싱을 수행함으로써 콘텐츠의 파일 포맷에 의존적인 스트리밍 서비스와는 달리 재생기에 독립적인 서비스를 가능하게 한다. 분리된 미디어는 독립된 채널을 이용해 전송되고, 각 채널은 미디어에 최적화 되어있다. 이는 임의의 미디어가 전송되고 있는 채널에 대한 네트워크의 연결이 중단될 경우, 다른 채널로 전송되는 미디어는 클라이언트로 전달 가능하므로 최종 사용자에게 고품질의 스트리밍 서비스를 제공할 수 있다. 또한, 재생기의 자원 사용량을 감소시키며, 스트림의 처리 시간을 단축시킨다.

그러므로, 서버 디멀티플렉싱 기법을 사용할 경우, 기존의 PC 환경에서 뿐 만 아니라 무선환경과 같이 자원이 부족한 환경에서 동작하는 멀티미디어 재생기 개발에 응용될 수 있다.

참고문헌

- [1] ISO/IEC JTC1/SC29/WG11 N2201, "Coding of Moving Pictures and Audio," May 1998
- [2] S. Battista, F. Casalino and C. Lande, "MPEG-4: A Multimedia Standard for the Third Millennium. 1," IEEE Multimedia , Vol. 6, No. 4, pp. 74 -83, Oct.-Dec. 1999
- [3] S. Battista, F. Casalino and C. Lande, "MPEG-4: A Multimedia Standard for the Third Millennium. 2," IEEE Multimedia , Vol. 7, No. 1, pp. 76 -84, Jan.-March 2000
- [4] H. Kalva, Li Tang, J.-F. Huard, G. Tselikis, J. Zamora, Lai-Tee Cheok and A. Eleftheriadis, "Implementing Multiplexing, Streaming, and Server Interaction for MPEG-4," Circuits and Systems for Video Technology, IEEE Transactions on , Vol. 9, No. 8 , pp. 299-1311, Dec. 1999
- [5] J. Kneip, B. Schmale and H. Moller, "Applying and Implementing the MPEG-4 Multimedia Standard," IEEE Micro, Vol. 19, No. 6 , pp. 64 -74, Nov.-Dec. 1999
- [6] Real Technical White Paper  
<http://www.real.com/player/index.html>
- [7] Windows Media Technologies  
<http://www.microsoft.com/windows/windowsmedia-/en/default.asp>
- [8] Quick Time Streaming  
<http://developer.apple.com/techpubs/quicktime/-qtdevdocs/PDF>
- [9] Q. Zhang, W. Zhu and Ya-Qin Zhang, "Resource Allocation for Multimedia Streaming over the Internet," Multimedia, IEEE Transactions on, Vol. 3, No. 3, pp. 339-355, Sep., 2001