

---

# HTTP상에서 동적 적응적 스트리밍 시스템 설계 및 구현

반태학\* · 정상호\* · 유소라\* · 김호겸\*\* · 정회경\*

\*배재대학교 컴퓨터공학과, \*\*한국전자통신연구원

## Design and implementation of a Dynamic Adaptive Streaming System over HTTP

Tae-hak Ban\*·Sang-ho Jung\*\*·So-ra Yu\*·Ho-gyom Kim\*\*·Hoe-kyung Jung\*

\*Dept. of Computer Engineering, Paichai University

\*\*ETRI(Electronics and Telecommunications Research Institute)

E-mail : \*{banth, tophojs, llllyusora, hkjung}@pcu.ac.kr, \*\*hokykim@etri.re.kr

### 요 약

최근 QoS/QoE 기술의 일환으로 HTTP에 기초한 적응적 스트리밍 기술이 주목받고 있다. 이에 본 논문에서는, HTTP에 기초한 동적 적응적 스트리밍 기술에 대해 알아본다. 이를 기반으로 HTTP상에서 동적 적응적 스트리밍 시스템을 설계 및 구현 하였다. 이 시스템은 MPEG2-TS 파일의 비트율별 변환, Segment 분할, MPD(Media Presentation Description)의 생성과 서버와 클라이언트 간의 동적이고 적응적인 네트워크 환경에서 MPD File의 분석을 통한 비트율별 플레이어로 구성된다. 이는 다양한 네트워크 환경에서 지속적이고, 원활한 영상의 재생을 위해 다양한 멀티미디어 분야에 활용될 것이다.

### ABSTRACT

Last QoS / QoE technology as part of the HTTP-based adaptive streaming technology has attracted attention. In this paper, HTTP-based adaptive streaming technology Find out about the dynamic. It is based on a dynamic adaptive streaming system over HTTP was designed and implemented. The system converts by the bit rate of MPEG2-TS files, Segment Split, MPD (Media Presentation Description) between servers and clients with the creation of a dynamic and adaptive analysis of network environments over MPD File consists bitrate's player. This diverse network environments, continuous and smooth playback of video will be used in various multimedia fields.

### 키워드

MPEG DASH, FFMPEG, Segment, MPD

### 1. 서 론

기존의 HTTP상에서 스트리밍 방식은 대상 동영상으로부터 한가지의 비트율을 클라이언트에 서비스하여 사용자로 하여금 일정한 비트율로 동영상 서비스를 받는 구조였다. 이러한 구조는 통신망 상에서의 간섭이나 트래픽이 일정순간 과도하게 생기다 보면 사용자는 버퍼링이나 화면의 끊김과 같은 불편함을 경험할 수 있었다. 이러한 문제점들을 해결하기 위해 최근 QoS/QoE 기술의 일환으로 HTTP에 기초한 MPEG DASH(Dynamic

adaptive streamin over HTTP) 프로그램의 연구와 개발이 주요 이슈화 되고 있다. HTTP상에서 동적 적응적 스트리밍이란, 동영상을 플레이하여 재생할 때, 통신망이나 트래픽 등의 간섭을 고려하여 현재의 네트워크 상황에 맞는 비트율의 화질을 서비스하여 제공하는 것을 말한다. 본 논문에서는 MPEG DASH를 이용한 프로그램 구현에 대해 소개하고, 구현내용에 대해 기술한다.

본 논문의 구성은 다음과 같다. 2장에서는 관련 연구에 대하여 기술하고, 3장에서는 DASH 구현내용에 대해 상세히 기술한다. 결론은 4장에서

기술한다.

## II. 관련연구

### 2.1 FFMPEG[1]

FFMPEG은 디지털 음성 스트림과 영상 스트림에 대해서 다양한 종류의 형태로 기록하고 변환하는 컴퓨터 프로그램이다. FFMPEG은 명령어를 직접 입력하는 방식으로 동작하며 여러 가지 자유 소프트웨어와 오픈소스 라이브러리로 구성되어 있다. 라이브러리 중에는 libavcodec도 들어있는데, 이 라이브러리는 음성/영상 코덱 라이브러리로 여러 프로젝트에 쓰이고 있으며, libavformat 라는 음성/영상 다중화, 역다중화 라이브러리도 있다. FFMPEG은 MPEG 영상 표준화 그룹에서 유래했고, "mpeg" 앞에 붙은 "FF"는 "fast forward"를 의미한다. FFMPEG은 리눅스 기반으로 개발되었지만, 대부분의 OS에서 컴파일 이 가능하다.

### 2.2 MPD(Media Presentation Description)[2]

통신망의 상태 변화에 신속히 대응하기 위해서는 전체 콘텐츠 파일을 적당한 크기의 조각으로 나누어 파일로 저장해 두어야 한다. 그리고 이렇게 조각난 파일들을 어떻게 순차적으로 가져와서 AV콘텐츠를 재생할 것인지에 대한 정보를 메타데이터로서 별도의 파일로 만들어 서버에 올려 두어야 한다. 이러한 메타데이터 파일을 MPD라고 부르는데, 여기에는 조각난 파일들에 상응하는 콘텐츠 시간 상의 위치, URL, 크기 등 클라이언트가 미리 알아야 할 정보가 들어 있다. 클라이언트는 MPD파일을 HTTP로 다운로드하여 분석한 후, 전송(Delivery) 포맷에 따라 콘텐츠가 저장된 조각 파일들을 순차적으로 HTTP로 다운로드하여 AV를 재생하게 된다. 통신망 환경과 단말 환경에 대한 분석과 MPD파일의 정보를 이용한 전송 최적화에 대한 부분은 클라이언트에서 담당한다.

## III. DASH 시스템 설계 및 구현

### 3.1 DASH 설계

MPEG DASH의 설계에서 서버에서는 크게 3 단계로 분류가 된다. 첫 번째로는 원본 영상에 대해 코덱정보, 해상도, 압축비트율에 대한 정보를 설정하여 레프리젠테이션을 설정하는 비트 스트림 생성, 두 번째는 비트 스트림 파일들에 대해 세그먼트 단위의 시간과 HTTP WebServer에 대한 저장장소를 설정해 주는 세그먼트 생성, 마지막으로 세그먼트를 기초하여 MPD문서를 만드는 메타데이터 생성 부분으로 구성된다.

클라이언트의 설계도 크게 4단계로 분류가 된다. 첫 번째로 서버측으로의 MPD문서를 요청하여 전달받은 메타데이터 분석 부분, 두 번째는

Segment file을 송 수신하는 Seamless스위치 부분, 세 번째로 전달받은 Segment file에 대해 초기화하여 분석하는 세그먼트 컨트롤러 부분, 마지막으로 MPD 문서안의 세그먼트들의 대한 정보를 분석하여 재생시켜주는 미디어 플레이어 부분으로 구성된다. DASH 설계는 그림1과 같다.

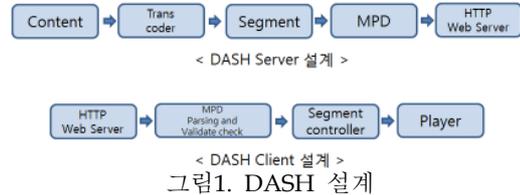


그림1. DASH 설계

### 3.2 MPD 설계

MPD문서는 여러 속성을 사용하여 Live와 OnDemand의 구분, 콘텐츠가 서버에서의 사용유무, 미디어의 플레이시간 등의 정보를 포함하고 있으며, 미디어 콘텐츠를 여러개의 Segment단위의 파일정보를 포함하고 있으며 이를 Period라고 한다. Period 같은 경우, 시작시간(Start Time)을 포함하고 있는데 라이브는 Period안의 시작시간과 MPD문서안의 시작시간을 합산하여 콘텐츠의 시작시간을 결정하고, OnDemand의 경우 MPD 문서안에 명시된 시작시간이 0이기 때문에 MPD안의 시작시간이 처음 콘텐츠의 재생시간이 된다.

하나의 Period안에는 하나이상의 레프리젠테이션(Representation) 들로 구성될 수 있고, 레프리젠테이션의 구성된 하나이상의 콘텐츠에 대해 코덱의 종류, 영상의 비트율, 음성의 비트율, 해상도, 언어 등을 포함 할 수 있다. 레프리젠테이션이 하나 이상인 이유는 통신망 상에서 간섭이나 트래픽의 이유로 버퍼링이나 끊김이 발생 할 수 있기 때문에 서로 대체 가능한 콘텐츠를 서비스 하기 위해서 이다. MPD설계는 그림2와 같다.

```
<?xml version="1.0" encoding="UTF-8"?>
<MPD xmlns:xsi="http://www.w3.org/2001/XMLSchema"
      xmlns="urn:mpeg:mpegB:schema:DASH:MPD:DS2011"
      xsi:schemaLocation="urn:mpeg:mpegB:schema:DASH:MPD:DS2011"
      profiles="urn:mpeg:mpegB:profile:dash:isoff-basic-on-demand:ca"
      type="onDemand"
      mediaPresentationDuration="PT10M.00S"
      minBufferTime="PT10.00S">
  <Period>
    <Group segmentAlignmentFlag="true" mimeType="video/3gpp"
      <Representation bandwidth="249867" id="test0">
      <segmentInfo duration="PT10.00S">
        <initializationSegmentURL sourceURL="http://203.250.143.6/test/MIE-1.ts"/>
        <url sourceURL="http://203.250.143.6/test/MIE-2.ts"/>
        <url sourceURL="http://203.250.143.6/test/MIE-3.ts"/>
        <url sourceURL="http://203.250.143.6/test/MIE-4.ts"/>
        <url sourceURL="http://203.250.143.6/test/MIE-5.ts"/>
        <url sourceURL="http://203.250.143.6/test/MIE-6.ts"/>
        <url sourceURL="http://203.250.143.6/test/MIE-7.ts"/>
        <url sourceURL="http://203.250.143.6/test/MIE-8.ts"/>
        <url sourceURL="http://203.250.143.6/test/MIE-9.ts"/>
        <url sourceURL="http://203.250.143.6/test/MIE-10.ts"/>
        <url sourceURL="http://203.250.143.6/test/MIE-11.ts"/>
        <url sourceURL="http://203.250.143.6/test/MIE-12.ts"/>
      </segmentInfo>
    </Representation>
  </Group>
</Period>
</MPD>
```

그림2. MPD 설계

### 3.3 구현 내용

MPEG DASH의 구현 내용은 다음의 표 1과 같은 환경에서 개발되었다.

표 1. MPEG DASH 구현

구성 내용	MPEG DASH (basic on-demand profile)
서비스 형태	VoD (Video on Demand)
미디어 포맷	M2TS or MP4
비디오/오디오 코덱	H.264 / AAC
비디오 / 오디오 레프리제이션	비디오 3개 (512, 1024, 2048kbps) 오디오 1개 (128kbps)
세그먼트 크기(초)	10 초
미디어 세그먼트	TS 패킷
SegmentAlignmentFlag	True

3.4 DASH 서버 구현

MPEG DASH의 서버구현에서는 원본 콘텐츠를 불러와서 미리 설정해둔 AV의 비트율별로 각각의 Trascoding을 실행하게 된다.

이 단계에서는 Trascoding된 콘텐츠들이 저장되는 경로와 제목을 설정해 줄 수 있다.

또한 Trascoding된 콘텐츠들을 처음 설계시와 같은 세그먼트단위(10초)로 나눌 수가 있는데 이 부분도 Trascoding부분과 마찬가지로 저장경로와 제목을 설정할 수 있고, 이렇게 비트율별로 변환되고 세그먼트로 나뉜 조각들의 정보를 하나의 MPD 문서로 생성할 수 있다. 생성된 MPD문서를 대상으로 Validation 체크 후 서버로 저장된다. DASH 서버 구현의 Form은 그림3과 같다.

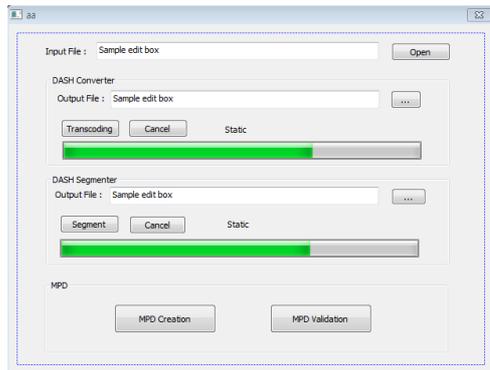


그림3. DASH 서버 구현

3.5 DASH 클라이언트 구현

MPEG DASH의 클라이언트(Player)구현에서는 서버로 MPD를 요청하여 전달받아 콘텐츠를 재생하여준다. 서버에 저장된 MPD문서의 URL을 입력시켜 주게 되면 클라이언트에서는 전달받은 MPD의 Validation 검증을 통해 MPD안에 있는 하나 이상의 Group에서의 각각의 Segment들을 Reading하여 플레이 하게 된다. 이 부분에서 일정시간동안 버퍼를 생성하게 되는데 망간섭이나 트래픽으로 인하여 버퍼에 저장되는 시간보다 서버에서 클라이언트로 저장되는 시간이 적으면 낮은 비트율의 동영상을 재생하게 되고, 통신망에서의 전송이 수월하게 진행하게 되면 높은 비트율의 영상을 재생하게 된다. 클라이언트(Player)에서

는 서버에 위치한 MPD의 URL을 입력하게되면 MPD에 대한 Validation 후 Segment에 대한 정보를 불러와 망상황에 맞는 영상을 재생하게 된다. DASH 클라이언트 구현의 Form은 그림4와 같다.

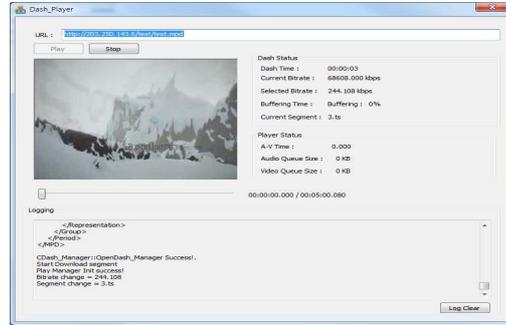


그림4. DASH Client 구현

IV. 결 론

최근 네트워크상에서 콘텐츠를 제공받아 영상을 시청할 때, 보다 유연한 영상을 서비스하기 위해 HTTP에 기초한 동적 적응적 스트리밍 기술이 개발 중이다. 망간섭이나 갑작스런 트래픽으로 인하여 높은 비트율의 콘텐츠를 제공할 때, 사용자는 버퍼링이나 화면 끊김과 같은 불편함을 경험해야 하였다. 이러한 문제점을 개선하기 위하여, 여러 화질의 비트율로 변환하여 네트워크 상황에 맞는 서비스를 제공하고 플레이하는 방법을 제시하였다.

참고문헌

- [1] <http://www.ffmpeg.org/>, 2011.08
- [2] ISO/IEC JTC1/SC29/WG11MPEG2010/M18403 : CODING OF MOVING PICTURES AND AUDIO, October 2010, Guangzhou, China
- [3] ISO/IEC JTC1/SC29/WG11 N12167 : CODING OF MOVING PICTURES AND AUDIO, July 2011, Torino, Italy
- [4] 하호진, 권오훈, 김선발, "Smart TV 비디오 서비스를 위한 HTTP 적응적 스트리밍", 삼성전자