

MP4 파일 포맷 기반 MPEG-DASH 클라이언트 구현

*박민규 *송해봉 *신아영 **김용한
 ,서울시립대학교
 **yhkim@uos.ac.kr

Implementation of MP4 File Format Based MPEG-DASH Client

*Park, Minkyu *Song, Heifeng *Shin, Ayoung **Kim, Yong Han
 ,University of Seoul

요약

MPEG-DASH(Dynamic Adaptive Streaming over HTTP)는 MPEG(Moving Picture Experts Group)에서 표준화 중인 HTTP를 이용한 적응형 비디오 스트리밍 기술이다. MPEG-DASH는 인터넷 방화벽에 의해 차단되는 RTP/UDP를 이용한 AV 스트리밍 대신 현재 HTTP/TCP를 이용하여 AV 스트리밍을 하는 추세에 맞춰 유무선 인터넷에서 망 환경 및 단말 환경에 동적 적응적으로 AV 스트리밍을 달성하기 위한 기술에 대한 표준이다. HTTP를 이용한 AV 스트리밍은 서버의 부담을 최소화하고 클라이언트의 지능적 처리에 전적으로 의존하는 스트리밍 방식이다. MPEG-DASH 클라이언트는 HTTP의 파일 전송 요청 또는 파일 일부 전송 요청만을 사용하여 스트리밍을 달성한다. 2011년 7월 제97차 MPEG 회의에서 MPEG-DASH DIS(Draft International Standard) 2차 버전이 승인되었다. 본 논문에서는 이 DIS 2차 버전에 따라 MP4 파일 포맷 기반의 MPEG-DASH 클라이언트를 구현하고 그 동작을 실험을 통해 검증하였다.

1. 서론

MPEG-DASH[1]-[3] 표준화를 위해 2010년 4월 드레스덴에서 개최된 제92차 MPEG 회의에서 기술제안요청서[4]를 공표하였고, 같은 해 7월 제네바에서 개최된 제93차 MPEG 회의에서 제안서들을 접수하였으며, 같은 해 10월 광저우에서 개최된 제94차 회의에서 CD(Committee Draft)가, 2011년 1월 대구에서 개최된 제95차 회의에서 DIS(Draft International Standard)가 승인되었으며, 2011년 7월 토리노에서 개최된 제97차 회의에서 DIS 2차 버전[5]이 승인되었다. MPEG-DASH의 공식 표준 번호는 ISO/IEC 23009-1이다

MPEG-DASH 클라이언트는 HTTP의 '파일 전송 요청' 또는 '파일 일부 전송 요청' 기능만을 사용하여 스트리밍을 달성한다. 따라서 망의 전송률의 변화에 적응적으로 대응하기 위해서는 같은 콘텐츠에 대해서 여러 가지 전송률로 압축된 파일들을 서버에 올려 두어야 한다. 또 이러한 망의 상태 변화에 신속히 대응하기 위해서는 전체 콘텐츠 파일을 적당한 크기의 파일 조각으로 나누어 저장해 두어야 한다. 또 파일 조각들을 순차적으로 가져와서 AV 콘텐츠를 재생할 수 있게 하는 메타데이터를 별도의 파일로 만들어 서버에 올려 두어야 한다. 이러한 메타데이터 파일을 매니페스트(Manifest) 또는 MPD(Media Presentation Description)라 부르는데, 여기에는 파일 조각들에 상응하는 콘텐츠 시간상의 위치, URL, 크기 등 클라이언트가 미리 알아야 할 정보가 들어있다. MPEG-DASH 클라이언트는 먼저 MPD 파일을 다운로드하여 분석한 후, 전송(delivery) 포맷에 따라 콘텐츠가 저장된 파일 조각들을 순차적으로 다운로드하여 AV를 재생하게 된다. 망 환경과 단말 환경에 대한 분석과 MPD 파일의 정보를 이용한 전송 최적

화는 전적으로 클라이언트의 몫이다. 현재 인터넷에서는 RTP에 의한 AV 스트리밍 대신, HTTP에 의한 AV 스트리밍이 널리 사용되고 있으므로, 이에 대한 MPEG-DASH 표준은 그 영향력이 매우 클 것으로 예상된다.

MPEG-DASH에서는 MPEG-2 TS(Transport Stream)[6] 또는 MP4 파일 포맷[7]을 기반으로 하는 두 가지 형태의 서버 콘텐츠를 지원할 수 있다. 본 논문에서는 제99차 회의에서 승인된 DIS 2차 버전을 기반으로 MP4 파일 포맷 기반의 MPEG-DASH 클라이언트를 구현하고 그 기능을 실험을 통해 검증하였다.

2. MPEG-DASH 서버 준비 및 시험 콘텐츠 제작

실험을 위해 서울시립대학교 영상통신연구실(이하 '연구실'이라 칭함)의 PC에 마이크로소프트의 윈도우즈 XP OS를 설치하고, 프리웨어인 APMSETUP을 설치하여 MPEG-DASH 서버를 마련하였다. 시험 콘텐츠는 프리웨어인 X264와 Advanced Audio Coder AAC를 이용하여 ETRI에서 제공한 '금강산' 시험 영상의 비디오와 오디오를 각기 H.264와 AAC로 압축 부호화하고, 그 결과를 MP4 파일 포맷으로 변환하여 제작하였다.

MP4 파일 포맷은 크게 파일 내 데이터 구성 및 속성을 알려주는 메타데이터 부분('moov' 박스)과 실제 압축된 미디어 데이터를 저장하는 부분('mdat' 박스)으로 구성된다. MP4 파일 내용 중 'moov' 박스의 내용은 초기화 세그먼트(Initialization Segment, IS)로 구성하고, 'mdat' 박스의 내용을 여러 미디어 세그먼트(Media Segment, MS)로 분할한다. MP4 파일 포맷에서는 압축된 미디어 데이터를 저장하는 방

법으로 두 가지를 제공한다. 하나는 'mdat' 박스 내에 비디오 압축 데이터를 한꺼번에 모두 먼저 넣고 이후에 이에 부수된 오디오 압축 데이터를 한꺼번에 넣도록 하는 방법이며, 다른 하나는 동기화되어 함께 재생하여야 하는 미디어들 즉, 비디오와 이에 부수된 오디오의 압축 데이터를 시간 순에 맞게 일부씩만 함께 담은 미디어 파편(media fragment)들로 나누어 순차적으로 여러 'moof' 박스들에 넣는 방법이다. 후자는 실시간으로 미디어를 저장할 때 유용한 포맷이다. 본 논문에서도 상기 후자에 해당하는 MP4 파일 포맷을 기반으로 'mdat' 박스의 내용물을 여러 세그먼트로 분할하였다.

상기 형식으로 제작된 콘텐츠의 한 MS의 재생시간은 5 초로 설정하였다. 비디오와 오디오를 분리하여 별도의 MS로 저장하였으며, 비디오와 오디오의 MS 파일 수는 각기 36 개이다. 따라서 시험 콘텐츠의 총 재생 시간은 180 초이다.

3. MPD 작성

MPD는 XML 형식으로 작성되며, 각각의 노드들은 그림 1과 같이 계층 구조를 갖는다. 그림 1은 일반적인 MPD의 계층도이다. 하나의 MPD에는 여러 구간(Period)들이 존재하며 각 구간은 미디어 콘텐츠 구성요소끼리 상호 대체가 가능한 여러 적응 집합(AdaptationSet) 지식 노드들로 구성되어 있다. 또 이 적응 집합은 이상의 미디어 스트림을 포함하고 있는 표현(Representation)으로 구성되어 있다.

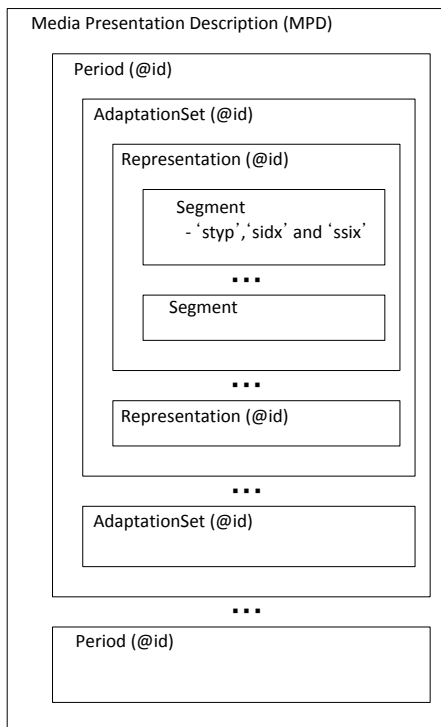


그림 1 MPEG-DASH MPD의 계층도

본 논문에서는 실험에서 사용한 MPD의 프로파일을 'ISO Base media file format On Demand'로 설정하였다. 재생시간 180 초에 해당하는 한 개의 구간(Period)은 지식 노드로서 비디오 적응 집합 노드와 오디오 표현 노드를 갖는다. 비디오 적응 집합 노드는 500 kbps 및 100 kbps로 압축된 비디오의 두 개 표현을 그 지식 노드로 갖는다. 오

디오 표현 노드는 96 kbps로 압축된 오디오를 갖는다. 이렇게 작성된 MPD는 그림 2와 같다.

```
<MPD xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
  xmlns="urn:mpeg:DASH:schema:MPD:2011"
  xsi:schemaLocation="urn:mpeg:DASH:schema:MPD:2011
  http://brasspounder.com:8087/pub/MPEG-DASH-2011v34.xsd" type="static"
  mediaPresentationDuration="PT180S" minBufferTime="PT10.00S"
  profiles="urn:mpeg:dash:profiles:isoff-main:2011">
  <BaseURL>http://lena.uos.ac.kr/</BaseURL>
  <BaseURL>http://172.16.165.74/</BaseURL>
  <Period start="PT0.00S" duration="PT180.00S">
  <BaseURL>mp4/</BaseURL>
  <SegmentList>
  <Initialization sourceURL="init.mp4"/>
  </SegmentList>
  <AdaptationSet mimeType="video/mp4" codecs="avc1.640828">
  <Representation id="1" bandwidth="500000">...</Representation>
  <Representation id="2" bandwidth="100000">...</Representation>
  </AdaptationSet>
  <AdaptationSet mimeType="audio/mp4" codecs="mp4a.40.2">
  <Representation id="3" bandwidth="80000">...</Representation>
  </AdaptationSet>
  </Period>
</MPD>
```

그림 2 실험에서 사용한 MPD

4. MPEG-DASH 클라이언트 구현

MPEG-DASH 클라이언트는 연구실에서 자체적으로 구현하여 사용하고 있는, MP4 파일 포맷을 지원하는 AV 재생 프로그램에 MPEG-DASH 기반 HTTP 스트리밍 서비스 지원 기능을 추가함으로써 구현하였다. 추가로 구현된 부분은 DIS 2차 버전에 따라 작성된 MPD를 분석할 수 있는 부분, 망의 전송률 변동에 따라 적응적으로 MS를 바꿔 가며 다운로드하는 부분, 기존 AV 재생 프로그램과 이벤트를 주고받는 부분 등이다.

상기와 같이 구현한 MPEG-DASH 클라이언트의 전체 구조도는 그림 3과 같다. MPD를 서버로부터 다운로드하여 비디오 서비스를 하기 위한 과정을 그림 3에 포함된 번호에 따라 설명하면 다음과 같다.

- ① MPEG-DASH 서버로부터 XML로 작성된 MPD를 다운로드하여 XML 영역에서 이를 분석한다.
- ② XML 영역에서 분석된 MPD가 MPEG-DASH DIS 2차 버전에서 정의한 바와 일치하는지 검사한다. MPD에서는 상위 계층으로부터 상속받아 정의된 노드들이 다수 존재할 수 있다. 상위 계층으로부터 상속 받은 노드의 속성의 기본 값은 상위 계층 노드의 속성이 되는데, 만약 하위 계층 노드의 값이 별도로 설정되어 있다면 기본 값은 무시된다.
- ③ 상기 처리 결과, MPD에서 추출된 IS의 URL 주소를 사용하여 MPEG-DASH 서버로부터 IS를 다운로드하여 콘텐츠 정보를 상세하게 확인한다.
- ④ MPD와 IS를 분석한 결과를 이용하여 트랙 제어부(track control)를 초기화한다. 이때 MPEG-DASH 서버로부터 MS를 다운로드 드할 쓰레드(thread)를 MPD에 적시된 적응 집합 개수만큼 생성하고, IS에 규정된 트랙 정보를 바탕으로 트랙별로 샘플 데이터를 저장할 메모리 공간을 할당한다.
- ⑤ 초기화를 마친 트랙 제어부는 기존의 MP4 재생부와 이벤트를 주고받을 수 있도록 연결하며, MPEG-DASH 파서(parser)는 MPEG-DASH 트랙 리더(track reader) 쓰레드들을 생성함으로써 콘텐츠를 재생할 수 있는 준비를 마친다.

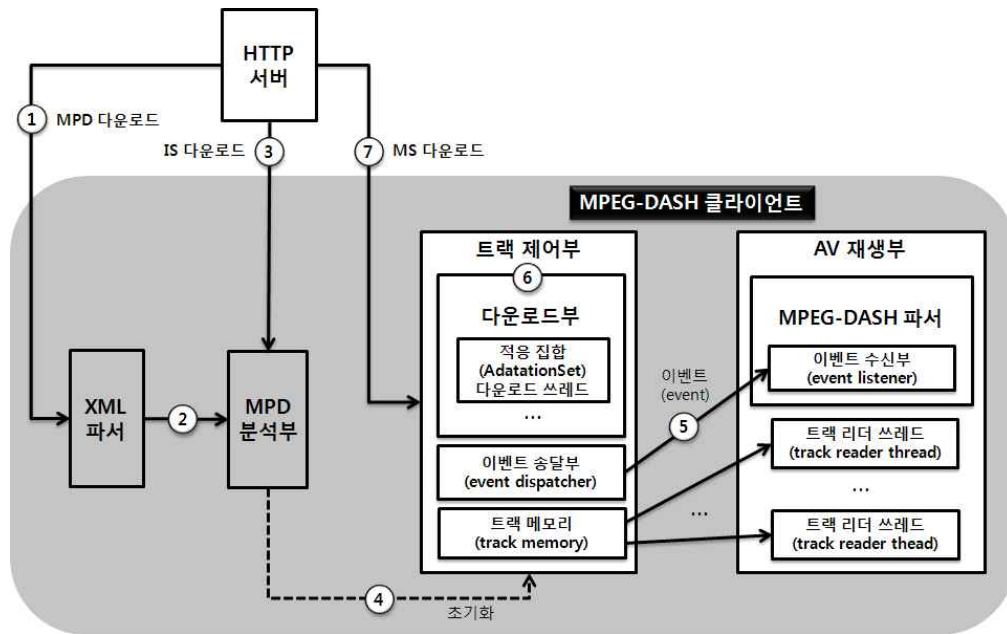


그림 3 MPEG-DASH 클라이언트의 구조 및 동작 흐름도

- ⑥ MPD에 적시된 적응 집합에 정의되어 있는 표현들을 바탕으로 망의 전송률에 맞춰 MPEG-DASH 서버로부터 MS를 다운로드하기 시작한다.
- ⑦ 다운로드 된 MS를 분석하여 미디어 샘플을 얻고, 이를 상기 단계 ④에서 생성된 트랙별 메모리 공간에 샘플 단위로 저장한다.

MPEG-DASH 클라이언트 개발에 사용된 운영체제는 마이크로소프트 윈도우즈 XP이고, 프로그램 개발 도구는 마이크로소프트 비주얼 C++ 6.0이다.

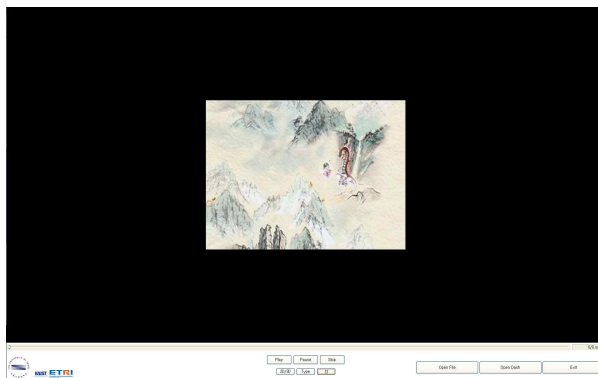


그림 4 MPEG-DASH 클라이언트의 재생 화면

5. 실험 결과

2절과 3절에서 설명한 콘텐츠와 MPD를 이용하여 4절에서 구현한 MPEG-DASH 클라이언트에 대한 기능 검증 실험을 시행하였다. MPEG-DASH 서버는 연구실 내에 설치하였으며, MPEG-DASH 클라이언트는 연구실 외부 PC에 설치한 후, MPEG-DASH 스트리밍을 시도하였다. 실험 결과, 망의 전송률에 따라 서로 다른 표현들을 바꿔가며 MS를 다운로드하고, 이를 정상적으로 재생함을 확인하였다. 그림 4는 개발된 MPEG-DASH 클라이언트의 재생 화면이다.

6. 결론

본 논문에서는 MPEG-DASH DIS 2차 버전 기반의 MPEG-DASH 클라이언트를 구현하였다. 시험용 세그먼트들을 제작하고, 이에 적합한 MPD를 작성하여, 개발된 MPEG-DASH 클라이언트의 기능을 시험하였다. 그 결과, 개발된 MPEG-DASH 클라이언트가 기능적으로 정상 동작함을 검증하였다.

참고문헌

- [1] ISO/IEC JTC 1/SC 29/WG 11 N11337, "HTTP Streaming of MPEG Media Context and Objectives," April 2010, Dresden, Germany.
- [2] ISO/IEC JTC 1/SC 29/WG 11 N11339, "Use Cases for HTTP Streaming of MPEG Media," April 2010, Dresden, Germany.
- [3] ISO/IEC JTC 1/SC 29/WG 11 N11340, "Requirements on HTTP Streaming of MPEG Media," April 2010, Dresden, Germany.
- [4] ISO/IEC JTC 1/SC 29/WG 11 N11338, "Call for Proposals on HTTP Streaming of MPEG Media," April 2010, Dresden, Germany.
- [5] ISO/IEC JTC 1/SC 29/WG 11 N12166, "Text of ISO/IEC 2nd DIS 23009-1 Dynamic Adaptive Streaming over HTTP," July 2011, Torino, Italy.
- [6] ISO/IEC 13818-1:2007, Information technology - Generic coding of moving pictures and associated audio information: Systems, 3rd Edition.
- [7] ISO/IEC 14496-14:2003, Information technology - Coding of audio-visual objects - Part 14: MP4 file format.