

# Smart TV 비디오 서비스를 위한 HTTP 적응적 스트리밍

하호진 | 권오훈 | 김선발  
삼성전자

## 요약

본고에서는 스마트 TV의 대표적인 서비스인 OTT(Over-The-Top)서비스의 전송 방법 중 최근에 가장 주목을 받고 있는 HTTP에 기초한 적응적 스트리밍 기술을 알아본다. 먼저, 최근 MPEG등 각 표준단체에서 진행되고 있는 HTTP 스트리밍 기술의 표준화 범위 및 진행 상황에 대해서 살펴보고, 표준화 범위 중 핵심이 되는 메타데이터 과일 구조 및 청크 단위의 전송 포맷 및 기존 시스템 포맷의 확장 사항 등에 대해서 설명한다. 마지막으로 MPEG DASH(Dynamic adaptive streaming over HTTP)의 basic on-demand profile에 기초한 구현 내용 및 결과를 소개한다.

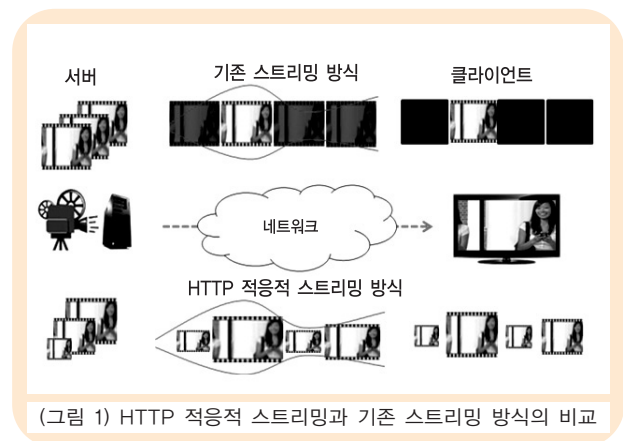
## 1. 서론

기존의 단순한 영상물을 보고 즐기는 TV에서 현재는 콘텐츠 플랫폼 기반으로 다양한 콘텐츠 및 어플리케이션을 제공하는 스마트 TV로 패러다임이 바뀌고 있다[1-2]. 기존의 지상파 및 케이블 영상에 대한 시청은 물론이고, 항상 네트워크에 연결이 되어 있다는 점에서 IPTV와 유사하지만, 스마트 TV는 콘텐츠 플랫폼을 기반으로 전세계 개발자들이 지속적으로 업로드하는 수 많은 어플리케이션을 자유롭게 사용자가 이용한다는 점에서 큰 차이가 있다고 할 수 있다.

또한 기존의 셋탑 박스를 통해서 스트리밍 서비스를 구축하던 콘텐츠 사업자들도 웹 모듈화를 통해서 다양한 스트리

밍 서비스를 제공 할 수 있다. 대형 TV 스크린으로 OTT(Over-the-top)서비스를 확장할 수 있다는 점과 발생 수익의 대부분이 OTT사업자가 가져갈 수 있다는 것이 장점으로 부각되면서, Hulu[3], Netflix[4], Block buster[5] 등은 OTT 스트리밍 서비스를 통해서 다양한 콘텐츠를 서비스를 확장해 나가고 있다. 특히 사용자의 특성을 고려한 미디어 스트리밍 서비스는 사용자가 갖고 있는 선호도나 이력을 기초로, 사용자에게 맞는 스트림을 선택을 할 수 있도록 하는 추천 서비스 및 네트워크 특성이나 디바이스 성능에 따라 다양한 화질의 콘텐츠를 끊김 없이 제공하는 서비스를 제공하고 있다[6-8].

본고에서는 스마트 TV의 대표적인 서비스인 OTT서비스의 전송 방법 중 최근에 가장 주목을 받고 있는 HTTP에 기초한 적응적 스트리밍 방법에 관한 내용을 기술한다[6-15]. 기존의 스트리밍 방식은 RTSP, RTMP, MMS등과 같은 전용



〈표 1〉 HAS 관련 표준 및 비 표준 현황

| 이름         | Smooth Streaming                    | IETF HTTP Live Streaming | Adaptive HTTP Streaming Rel. 9 | HTTP Adaptive Streaming Rel. 2 | Dynamic Adaptive Streaming over HTTP      |
|------------|-------------------------------------|--------------------------|--------------------------------|--------------------------------|---|
| 회사/표준화 단체  | Microsoft                           | Apple                    | 3GPP                           | OIPF                           | MPEG                                      |
| 서비스 형태     | VoD(Video on Demand)/ Live          |                          |                                |                                |   |
| 미디어 포맷     | Fragmented MP4                      | MPEG 2 TS                | Fragmented MP4                 | MPEG 2 TS, Fragmented MP4      | MPEG 2 TS, Fragmented MP4                 |
| 비디오/오디오 코덱 | H,264, VC-1/ MPEG 4 AAC             | H,264/ MPEG 4 HEAAC      | H,264/ MPEG 4 HEAAC            | H,264/MPEG 4 HEAAC             | H,264, SVC, Multiview Coding / MPEG 4 AAC |
| 메타데이터 포맷   | .ism, .isms (server) .ismc (client) | .m3u8                    | .mpd                           |                                |   |
| 서버 의존성     | IIS                                 | 일반 웹 서버                  |                                |                                |   |
| 클라이언트 의존성  | Silverlight                         | No                       |                                |                                |   |
| 트릭 플레이 지원  | 지원 가능                               | 지원 되지 않음                 | 지원 가능                          |                                |   |

HEAAC : High Efficiency AAC  
 SVC : Scalable Video Coding  
 IIS: Internet Informational Server

스트리밍 프로토콜을 사용하고, CDN(Content delivery network)의 전용 회선을 이용하여, 현 시점에서 필요한 영상을 미디어 서버로부터 받는 방식이었다. 이는 마치 서버와 클라이언트가 하나의 논리적인 회로를 설정해주는 스테이트풀 방식으로 생각할 수 있다[16-18].

이와는 달리, 최근에는 YouTube[19], Vimeo[20], MySpace [21], 그리고 MSN Soapbox[22]등에서는 웹 표준 프로토콜인 HTTP를 이용하여 미디어 콘텐츠를 버퍼링하고 플레이를 하는 프로그레시브 다운로드 방식이 널리 이용되고 있다 [18]. 기존의 스트리밍 방식과는 달리, 웹 표준 프로토콜을 이용한 전송방식은, HTTP를 지원하는 종래의 구조적 체계나 아키텍처를 그대로 이용하여, 언제 어디서나 인터넷을 이용하여 미디어 서비스에 접근이 가능하다는 점과 인터넷에 존재하는 캐시 또는 프록시 서버 등을 이용하여 네트워크 효율적으로 사용할 수 있다. 또한 RTP/RTSP와 같은 전송 프로토콜의 경우처럼 NAT/firewall과 같은 이슈가 발생하지 않는다는 장점이 있기 때문에, 오늘날 웹 상의 비디오 전송 방식으로 널리 사용되고 있다[6][13].

또한 최근에는 HTTP 프로토콜의 장점과 필요한 영상을 바로 서버로부터 요청해서 플레이 하는 스트리밍 방식의 장점을 살린 하이브리드 전송 기술 방식으로써, HTTP 적응적 스트리밍(HTTP Adaptive Streaming, HAS) 기술이 주목을 받고 있다[8-15]. HTTP 상에서, 기존의 스트리밍 전송방식이

갖고 있는 전용 회선의 특징을 갖기 위해서, 전송 파일을 작은 단위의 세그먼트를 사용하여 미디어 콘텐츠를 구성하고, HTTP상에서 미디어 데이터를 전송하는 방식이다. 즉 회선이 좋지 않은 상황에서, 고정된 화질을 맞추기 위해서 일정량을 미리 버퍼링 하는 과정을 피하기 위해서, 미디어 콘텐츠는 여러 개의 비트 율에 대해서, 일정한 시간단위의 청크로 나누어 저장된다. 네트워크의 특성을 고려하여, HAS 클라이언트는 다양한 비트 율을 적응적으로 요청함으로써, 미디어를 연속적으로 재생하게 된다. (그림 1)은 HTTP 적응적 스트리밍과 기존 스트리밍 방식의 차이를 보여주고 있다.

본 논문은 본문에서는 현재 여러 표준화 단체에서 진행중인 HAS의 주요 표준 범위 및 각 표준단체의 스펙 범위에 대해서 소개한다. 또한 HAS 스펙 중, 메니페스트, 미디어 세그먼트 포맷에 대한 특징, 그리고 MPEG DASH basic on-demand profile 에 기초한 구현 내용 및 결과를 소개한다. 마지막으로 3장에서 결론을 맺는다.

## II. 본 론

### 1. 표준화 현황

(그림 2)는 HTTP에 기초한 적응적 스트리밍 기술의 표준

화 범위를 나타내고 있다. 서버에서는, 망 상태 변화에 적응적으로 대응하도록, 전체 콘텐츠 파일을 적당한 크기의 조각으로 나누어 파일로 저장한다. 조각난 AV파일들에 대한 정보는 메타데이터의 별도의 파일로 저장한다. 클라이언트에서는, 전달된 메타데이터 상의 파일 정보를 통해서, 서버에 저장된 파일의 정보를 알고, HTTP 요청을 통해서 AV 콘텐츠가 전달되고, 재생된다. 조각난 파일을 효과적으로 전송하기 위한 전송 포맷 및 조각난 파일에 관련된 시간 정보 및 URL등의 메타데이터 정보의 구성 등이 표준화 범위에 포함된다.

3GPP(3 Generation Partnership Project)는 AHS (Adaptive HTTP Streaming)라는 이름으로 2010년 3월에 릴리즈 9 버전 9.2.0을 완료하였고, 현재 버전 9.4.0를 수정 중이다. 주요 내용은 메타데이터 구성과 ISOBMFF(ISO base media file format)를 이용한 전송 세그먼트의 포맷에 관련된 내용이 포함된다. 추가되는 사용자 시나리오에 대응하기 위해서 릴리즈 10의 표준화가 현재 진행 중이다[9-11].

OIPF(Open IP TV Forum)는 릴리즈 2가 2010년 9월에 HAS(HTTP Adaptive Streaming)라는 이름으로 완료되었다. 3GPP 릴리즈 9 버전 9.2.0을 기초로 메타데이터가 구성되어 있으며, 미디어 세그먼트 포맷으로 MPEG 2 TS 및 ISOBMFF를 지원한다[12].

MPEG은 현재 2011년 7월에 FDIS(Final Draft International Standard)를 목표로 표준화가 진행 중이다. 3GPP의 릴리즈 9 버전 9.4.0을 기초로 추가되는 사용자 시나리오에 대응하기 위해서 메타데이터를 확장 중에 있으며, 지원되는 미디어 세그먼트 포맷으로 MPEG 2 TS 및 ISOBMFF를 포함한다. 주요 확장 내용은 다양한 비트율 및 코덱 정보에 따른 그룹핑 정보, 스케일러블 비디오 코딩 및 다 시점(Multiview) 비

디오 코딩에 대한 지원 등을 포함하고 있다[13].

비 표준 HAS 기술로서, 마이크로 소프트에서는 Smooth Streaming방식을 사용하고 있다. 메타데이터는 서버 단의 ism과 isms, 그리고 클라이언트 단의 ismc 파일을 필요로 하게 되며, 서버는 일반 HTTP 웹 서버가 아닌 IIS (Internet Informational Server)의 전용 서버를 사용해야 한다. 미디어 세그먼트 파일은 ISOBMFF및 MP4를 사용하는 Fragmented MP4를 구성하여 전송하는 방식을 사용한다[14]. 애플 사의 IETF(Internet Engineering Task Force) HTTP Live Streaming은 인터넷 표준으로 제안 중에 있다. 메타데이터는 확장된 m3u8을 사용하고 있으며, 미디어 세그먼트 파일은 MPEG 2 TS를 기반으로 구성된다. 서버는 일반적인 HTTP 웹 서버를 이용하여 구현될 수 있다[15]. HAS 관련 표준 및 비 표준관련 자세한 내용은 표1에 정리되어 있다[10][12-15].

## 2. 표준의 주요 내용: MPD 및 세그먼트 구성

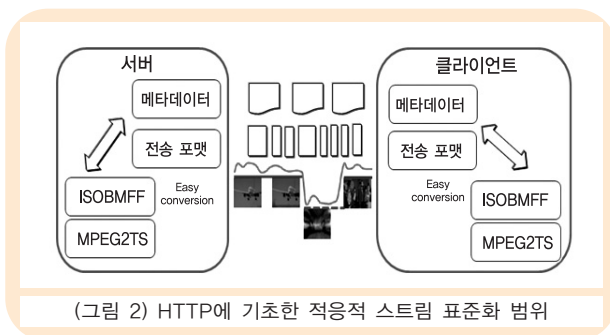
### 2.1 MPD 구성

조각난 파일에 관련된 시간 정보 및 URL등의 정보를 클라이언트에 효과적으로 전달하기 위해서, 메타데이터 구성에 대한 표준이 진행되고 있고 이를 MPD(Media presentation description)로 명명한다. (그림 3)은 4개의 레벨로 구성되어 있는 MPD의 전체구조를 나타낸다. : MPD, 피리어드, 레프리젠테이션, 세그먼트 레벨[10-13].

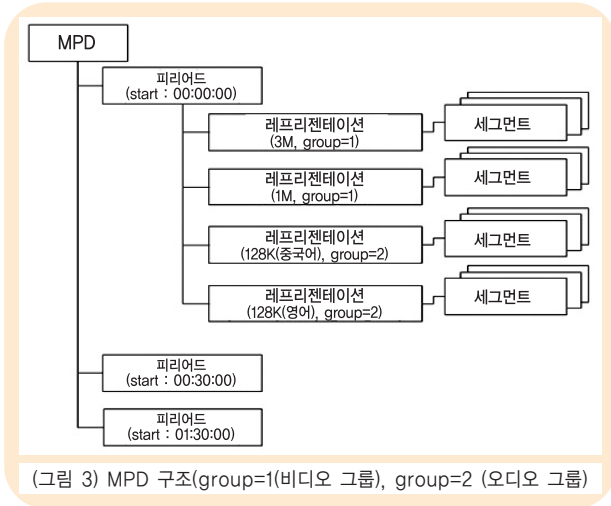
MPD 레벨은 여러 속성을 이용하여 CoD(Content on demand)와 라이브 서비스의 구분, 미디어 콘텐츠가 서버단에서 언제 이용 가능한 지, 언제 이용할 수 없는지, 그리고 미디어의 전체 플레이 시간 등의 정보를 제공한다.

미디어 콘텐츠는 여러 개의 시간 단위로 구성되고, 이를 피리어드(Period)로 명명한다. 피리어드는 시작 시간(start)을 갖는다. 라이브의 경우에는 피리어드 내의 시작 시간과 MPD내의 시작 시간(UTC, Universal Time Coordinated)을 합하여, 각 피리어드의 시작 시간을 결정한다. 반면, CoD의 경우에는 MPD내의 시작 시간은 항상 0이기 때문에, 처음 피리어드의 시작 시간이 콘텐츠의 시작시간이 된다.

하나의 피리어드가 나타내는 시간 범위 안에는, 하나 이상의 레프리젠테이션(Representation)들이 구성 될 수 있다. 레프리젠테이션의 구성은 하나의 미디어 콘텐츠에 대해서, 코덱의 종류, 비트율, 해상도, 프레임 율, 그리고 사용된 언어



(그림 2) HTTP에 기초한 적응적 스트림 표준화 범위



등에 따라서 다양한 구성을 가질 수 있고, 이를 컴포넌트라고 명명한다. 이러한 컴포넌트의 구성 정보는 레프리젠테이션 레벨의 내부 속성(예를 들면, mimetype, bandwidth, resolution)들에 의해서 명시되고 전송된다. 대표적인 레프리젠테이션의 예는, 다양한 망의 채널 상태에 따라서, 사용자가 알맞은 비트 율의 청크를 선택하도록 하기 위해서, 다양한 비트 율의 레프리젠테이션들이 구성될 수 있다. 또한 레프리젠테이션 내에서는 그룹 속성을 이용해서 여러 개의 레프리젠테이션을 그룹화한다. 같은 그룹에 포함된 레프리젠테이션은 서로 대체 가능한(alternative) 컴포넌트들로 구성된다.

만약 그룹 값이 0이면, 여러 개의 컴포넌트가 하나의 레프리젠테이션으로 구성된 경우로써, 여러 개의 레프리젠테이션을 조합하지 않아도 미디어 콘텐츠를 구성할 수 있다. 그러나 그룹 값이 0이 아니면, 서로 다른 0이 아닌 그룹의 레프리젠테이션 중 하나를 뽑아서 그들의 조합을 구성하면, 여러 개의 미디어 콘텐츠가 생성된다.

세그먼트는 피어어드를 구성하는 시간 단위의 조각으로써, MPD상에서 HTTP-URL[19]에 의해서 참조 되는 단위로 정의 된다. 이는 망의 상태 변화에 신속히 대응하기 위해서, 전체 콘텐츠 파일에서 적당한 크기의 조각으로 나누어, 파일로 저장되고, 세그먼트 레벨의 HTTP 질의에 의해서 클라이언트에 전달된다.

세그먼트 인덱싱을 위한 URL은 상대(relative) 주소와 절대(absolute) 주소를 사용하여 세그먼트에 접근한다[19]. 상대

주소의 구성은 MPD내 각각의 레벨에 존재하는 주소를 baseURL에 붙임으로써 생성된다. 물론 절대 주소는 세그먼트 레벨에 지시하려는 세그먼트의 주소를 직접 기입함으로써, 각 세그먼트에 접근할 수 있다. 또한 세그먼트 URL은 템플릿을 이용하여 구성될 수도 있다.

템플릿의 구성은 레프리젠테이션 레벨에서의 id와 피어어드와 세그먼트 레벨의 템플릿을 이용하여 구성된다. 그림. 4는 템플릿을 사용하여 세그먼트를 인덱싱 하는 예를 보여주고 있다. 첫 번째 세그먼트의 http-URL은 “http://168.219.193.204/SPIDER\_MAN\_III\_1920x1080/1/T1\_V1.ts”로 구성된다.

```
<Period start="PT0S" segmentAlignmentFlag="True">
  <SegmentInfoDefault duration="PT10S"
  sourceUriTemplatePeriod=http://www.example.com/SPIDER_MAN_III_1920x1080/$RepresentationId/$T1_V$Index$.ts startIndex="0"/>
  <Representation mimeType="video/mpeg
  codec=mp4v.20.9,mp4a.E1" bandwidth="4000000"
  id="1" startWithRAP="True" width="1280"
  height="720" group="1">
    <SegmentInfo <UriTemplate endIndex="22"/>
  </SegmentInfo>
</Representation>
</Period>
```

(그림 4) 템플릿을 사용하는 세그먼트 인덱싱의 예

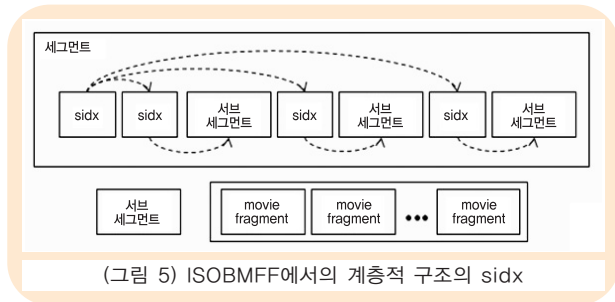
## 2.2 세그먼트 구성

표준 스펙에서의 세그먼트는 크게 MPEG 2 TS [24]와 ISOBMFF [25]을 지원한다. 먼저 MPEG 2 TS는 초기화 세그먼트와 미디어 세그먼트로 구성 된다. 초기화 세그먼트는 엘리멘터리 스트림을 구별하기 위한 PAT (Program association Table)/PMT (Program mapping table) 정보를 포함한다. 또한 하나의 초기화 세그먼트에 모든 레프리젠테이션의 PID(Packet ID) 값이 포함될 수 있고, 각각의 레프리젠테이션이 각각의 초기화 세그먼트를 포함할 수도 있다. 또한, 콘텐츠 보호를 위해서, PSI(Program Specific Information)에 CAT(Conditional Access Table)가 포함될 수 있고, 스크램블된 세그먼트의 TS 패킷을 위해서, 디스크램블을 위한 정보 또한 포함될 수 있다.

이와는 달리 미디어 세그먼트는 PAT/PMT를 제외한 실제 미디어 데이터를 포함한다. 미디어 세그먼트의 세부 구조는

MPD내의 세그먼트 구성관련 속성들에 의해서 구성된다. : *StartWithRAP*, *SegmentAlignmentFlag*, 그리고 *Bitstream switchingFlag*

*StartWithRAP*가 설정되면, 세그먼트의 첫 번째 액세스 유닛, 즉 TS 패킷은 랜덤 액세스 포인트 (RAP, Random Access Point)로 구성된다. 해당 액세스 포인트에서 시간 확인 후 바로 점프 및 스위칭이 가능하도록, 세그먼트의 첫 번째 PES(Presentation time stamp)의 시간 정보를 포함하여 구성된다. 스위칭이 가능하도록, 세그먼트의 첫 번째 PES (Packet Element Stream) 패킷은 PTS(Presentation Time Stamp)의 시간 정보를 포함하여 구성된다. 또한 미디어 컴포넌트를 구성하는 코덱의 종류에 따라서, RAP에는 디코딩 관련 정보가 포함될 수도 있다. 예를 들면 ISO/IEC 14496-10 [26]의 경우에는 SPS (Sequence Parameter Set)/PPS (Picture Parameter Set)의 정보가 포함되어야 한다.



*SegmentAlignmentFlag*가 설정되면, 해당 피리어드 내의 모든 레프리젠테이션에 포함된 세그먼트들은 시간 축으로 정렬이 된다. 즉 세그먼트의 시작과 끝의 시간이 일치하게 된다. 이를 위해서, 첫 번째 TS 패킷은 시간 정보를 포함하는 PES로 구성된다. 세그먼트 사이의 Seamless 스위칭을 위해서, 미디어 세그먼트는 분리되지 않은 PES 패킷으로 구성된다.

*BitstreamswitchingFlag*값이 설정되면, 각 레프리젠테이션 내의 세그먼트를 통해서 구성된 여러 비트 스트림들은 스위칭이 가능하게 된다. 이를 위해서 각 세그먼트들은 타임 정보를 포함해야 한다.

ISO/BMFF(ISO based media file format)의 경우에, 초기화 세그먼트는 ftyp (File Type), moov 그리고 *pdin* (Progressive download information)로 구성된다. *moov*는 시작 시간을 줄

이기 위해서, 샘플에 관련된 세부 정보인, *stbl* (sample table box) 등을 포함하지 않으므로써, 작은 크기를 유지한다. 또한 *mvex* (movie extends box)를 이용해서, 미디어 세그먼트가 fragmented로 구성 됨을 표시한다. 또한 *moov*에는 미디어 컴포넌트 코덱 관련 내용들이 포함된다. 예를 들면, *moov*에 ISO/IEC 14496-10 [25] 초기화와 관련된 SPS/PPS의 정보가 포함된다.

미디어 세그먼트는 하나 이상의 movie fragment 로 구성된다. *trun* (track run)을 포함하는 *moof* (movie fragment)와 미디어 데이터를 포함하는 *mdat* (media data)로 구성된다. *moof*는 하나 이상의 track fragment를 포함할 수 있다 [11][25].

ISO/BMFF는 상대적인 오프셋 및 시간정보를 사용하기 때문에, 스위칭 및 점프를 위해서는 절대 시간 및 오프셋을 위한 정보가 필요하다. 이를 위해서, HAS 표준에서는 *sidc* (segment index box) 라고 하는 해당 세그먼트에 대한 정보 박스를 제공한다[10-11]. 이 박스는 세그먼트를 여러 개의 서브 세그먼트로 구성하고, 이를 계층적인 구조로 관련된 정보를 제공한다.

〈표 2〉 구현 내용

| 구성 내용                       | MPEG DASH (basic on-demand profile)       |
|-----------------------------|---|
| 서비스 형태                      | VoD (Video on Demand)                     |
| 미디어 포맷                      | MPEG 2 TS                                 |
| 비디오/오디오 코덱                  | H,264/ MPEG-4 AAC                         |
| 비디오 레프리젠테이션                 | 5개로 구성(kbps)<br>(778,1128,1628,2628,5128) |
| 오디오 레프리젠테이션                 | 1개로 구성(128kbps)                           |
| 세그먼트 크기(초)                  | 10 초                                      |
| 초기화 세그먼트                    | PAT/PMT                                   |
| 미디어 세그먼트                    | TS 패킷                                     |
| <i>StartwithRAP</i>         | True                                      |
| <i>SegmentAlignmentFlag</i> | True                                      |

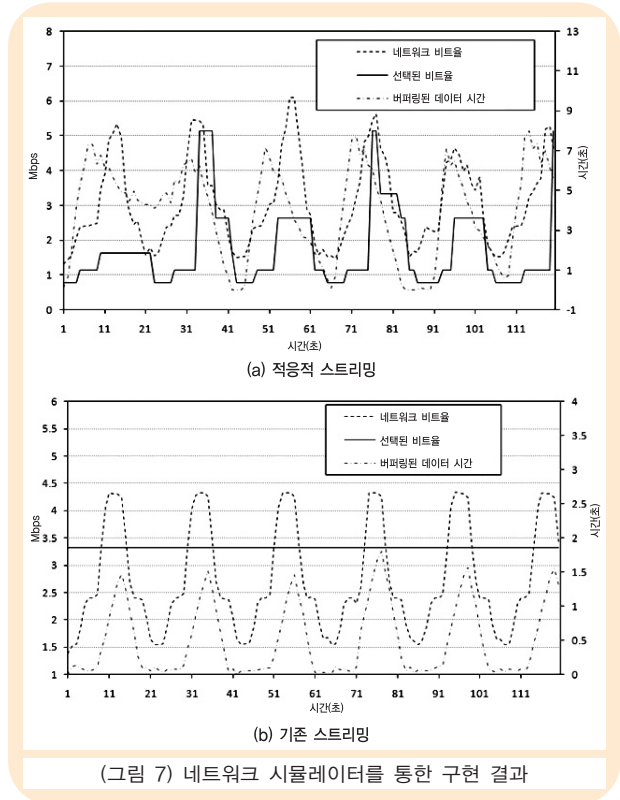
여러 개의 *sidc*박스 중, 제일 처음에 위치하는 박스는 해당 세그먼트의 디코딩 시간이 절대 시간으로 제공된다. 따라서 스위칭 시에, 클라이언트는 *sidc* 박스내의 시간 정보를 이용

해서, 연속적인 재생을 위한 가능한 세그먼트인지를 체크할 수 있다. 또한 그 다음에 존재하는 sidx 박스는 세그먼트를 구성하는 서브 세그먼트들에 대한 duration, RAP 존재 여부, 그리고 RAP의 시간정보 등을 포함하게 된다. 전체적인 구조는 (그림 6)에 나타나 있다.

SegmentAlignmentFlag가 설정되면, 해당 피리어드내의 모든 레프리젠테이션에 포함된 세그먼트 들은 일정한 시간 간격을 갖게 된다. 만약 이 값이 설정되어 있지 않으면, movie fragment는 정렬이 보장되지 않기 때문에, 각 세그먼트는 sidx에 포함된 시간정보 등을 이용해서, 스위칭을 수행한다. StartwithRAP와 Bitstreamswitching Flag 와 값은 위의 MPEG 2 TS의 경우와 일치한다[10][13].

### 3. MPEG DASH의 basic on-demand profile 구현내용 및 결과

본 장에서는 basic on-demand profile[27]에 기초한 MPEG DASH의 구현 내용을 설명한다. 구현 환경은 <표 2>와 같다.

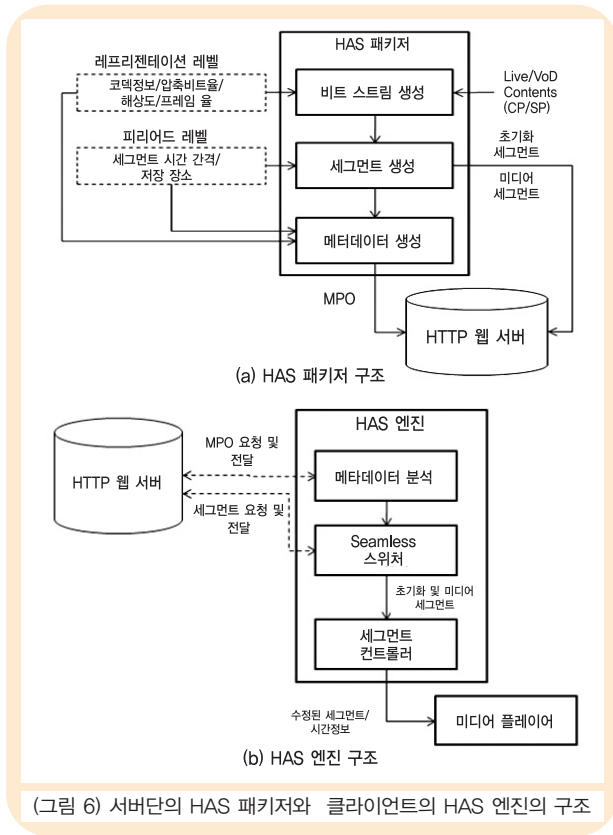


(그림 7) 네트워크 시뮬레이터를 통한 구현 결과

(그림 6)의 (a)와 (b)는 서버단의 HAS 패키지와 클라이언트의 HAS 엔진의 구조를 나타내고, 암호화 모듈은 생략하였다.

서버는 크게 3단계로 구분되어 있다. 비트 스트림 생성/세그먼트 생성/메타데이터 생성으로 구분된다. 비트 스트림 생성 단계에서는 CP/SP에서 제공되는 콘텐츠를 비디오/오디오 코덱을 이용하여 트랜스 코딩을 실시하여 비트 스트림을 생성한다. 이때 입력되는 정보로는 레프리젠테이션 레벨의 속성에 해당되는 코덱 정보, 비트율, 해상도 등이 해당된다. 그리고 생성된 비트 스트림을 미리 입력된 세그먼트 구조에 관련된 정보(세그먼트 크기/ 저장 장소)를 이용하여, 스트림 파일을 전송을 위한 작은 조각으로 나눈다. 마지막으로 입력된 정보를 기초로 하여 MPD를 생성한다.

클라이언트의 구조는 메타데이터 분석, seamless 스위처, 그리고 세그먼트 컨트롤러 구분된다. 메타데이터 분석은 서버에 존재하는 세그먼트의 구조 및 위치, 그리고 서비스 형태 등의 정보를 수집한다. 또한 seamless 스위처는 네트워크 상태의 분석을 통해서 망의 상태에 가장 적당한 비트 율의



(그림 6) 서버단의 HAS 패키지과 클라이언트의 HAS 엔진의 구조

세그먼트를 HTTP-url을 이용하여 요청한다. 마지막으로 전달 받은 세그먼트들을 끊김 없이 플레이 될 수 있도록 시간 정보의 체크 및 continuity counter등을 조정하여 미디어 플레이어로 전달하는 역할을 세그먼트 컨트롤러가 수행한다.

(그림 7 (a)와 (b))는 네트워크 에뮬레이터를 통한 HAS의 구현 결과를 보여주고 있다. (그림 7(a))는 네트워크 비트율 및 버퍼링 데이터의 시간에 따라, HAS 엔진은 5개의 레프리 쉐이션 레벨을 적응적 스트리밍을 보여주고 있다. 반면 (그림 7 (b))은 네트워크의 변화에 적응적 변화를 보이지 못하고, 버퍼의 고갈을 가져와 화면이 끊기면서 재생될 수 밖에 없음을 보여주고 있다.

### III. 결 론

본고에서는 스마트 TV의 OTT서비스에서 주목을 받고 있는 HTTP 적응적 스트리밍에 대한 기술을 소개하였다. 메타 데이터 및 체크 단위의 전송 포맷을 중심으로 3GPP, OIPF 그리고 MPEG등에서 표준화가 진행되고, 다양한 CP/SP 및 TV 메이커들에 의해서 활발한 사업화가 진행되고 있다. 따라서 국제 표준화에 특허를 반영하는 노력과 더불어서, 표준에 부합하는 솔루션을 조속히 개발하여 시장을 선점하려는 노력이 필요하다고 사료된다.

#### 참 고 문 헌

- [1] 황준호, “스마트 TV가 방송시장에 미치는 영향,” 정보통신 정책 연구원 리포트, 2010-03.
- [2] Google TV: Paradigm shift in TV and telecom; a new era is arriving, J.P. Morgan, 2010-May.
- [3] <http://www.hulu.com/>.
- [4] <http://www.blockbuster.com/>.
- [5] <http://www.netflix.com/>.
- [6] YouTube serves up 100 million videos a day online. [Online]. Available:[http://www.usatoday.com/tech/2006-07-16-youtube-views\\_x.htm](http://www.usatoday.com/tech/2006-07-16-youtube-views_x.htm)
- [7] X. Cheng, C. Dale, J. Liu, “Understanding the characteristics of internet short video sharing: YouTube as a case study,” Technical report, Cornell Univ., Jul. 2007.
- [8] R. S. Cruz, M.S. Nunes, and J.P. E. Goncalves, “A Personalized HTTP Adaptive Streaming WebTV,” in Proceedings of the First International Conference on User Centric Media Workshop, UCMedia '09. Venice, Italy, Dec. 2009.
- [9] P. Frojdh, U. Horn, M. kampmann, A. Nohlgren, and M. Westerlund, “Adaptive Streaming within the 3GPP Packet-Switched Streaming Service,” IEEE Networks, Mar 2006.
- [10] 3GPP TS 26,234, “Transparent End-To-End Packet-Switched Streaming Service (PSS): Protocols and Codecs (Release 9)”.
- [11] 3GPP TS 26,244, “Transparent End-To-End Packet-Switched Streaming Service (PSS): 3GPP File Format (3GP) (Release 6)”.
- [12] OIPF-T1-R2-Specification-Volume-2a-HTTP-Adaptive-Streaming-V2\_0-2010-09-07[1].
- [13] Information technology - MPEG systems technologies - Part 6: Dynamic adaptive streaming over HTTP (DASH), ISO/IEC CD 23001-6: Oct. 2010.
- [14] A. Zambelli, “IIS Smooth Streaming Technical Overview,” Microsoft Corporation, Mar. 2009.
- [15] R. Pantos, “Internet-Draft: HTTP Live Streaming: draft-pantos-http-live-streaming-03” Apr. 2010.
- [16] H. Schulzrinne, A. Rao, and R. Lanphier, “Real-Time Streaming Protocol (RTSP),” IETF RFC 2326, Apr. 1998.
- [17] H. Schulzrinne et al., “RTP:A Transport Protocol for Real-Time Applications,” IETF RFC 3550, Jul. 2003.
- [18] T. Friedman, R. Caceres, and A. Clark, “RTP Control Protocol Extended Reports (RTCP XR),” IETF RFC 3611, Nov. 2003.
- [19] Berners-Lee, et al, “URI Generic Syntax” IETF RFC 3986, Jan. 2005.
- [20] <http://www.youtube.com/>.

- [21] <http://www.vimeo.com/>.
- [22] <http://www.myspace.com/>.
- [23] [http://video.joinmsn.com/?mkt=ko-kr&tab=soap box](http://video.joinmsn.com/?mkt=ko-kr&tab=soap%20box).
- [24] ISO/IEC: Information technology-Generic coding of moving pictures and associated audio information:Systems, Tech, Rep. ISO/IEC 13818-1:2005.
- [25] ISO/IEC: Information technology-Coding of audio-visual objects-Part 14:MP4 file format, Tech, Rep, ISO/IEC 14496-14:2005.
- [26] ISO/IEC: Information technology-Coding of audio-visual objects-Part 10: Advanced Video Coding, Tech, Rep, ISO/IEC 14496-10:2005.
- [27] Information technology - MPEG systems technologies - Part 6: Dynamic adaptive streaming over HTTP (DASH), ISO/IEC FCD 23001-6: Jan, 2011.

**약 력**



1998년 명지대학교 제어계측공학과 학사  
 2000년 한양대학교 제어계측공학과 석사  
 2009년 연세대학교 전기 및 전자공학과 박사  
 2000년 ~ 현재 삼성전자 디지털 미디어 및 통신 연구소  
 책임연구원

관심분야: 멀티미디어 신호 처리 및 네트워킹

**하 호 진**



1995년 포항공과대학교 전자계산학과 학사  
 1997년 포항공과대학교 컴퓨터공학과 석사  
 1997년 ~ 2000년 포항공과대학교 첨단공학연구소 연구원  
 2007년 포항공과대학교 컴퓨터공학과 박사  
 2007년 ~ 현재 삼성전자 디지털 미디어 및 통신 연구소  
 책임연구원

관심분야: 스마트TV, 웹서비스, 콘텐츠 전송기술

**권 오 훈**



2003년 한남대학교 수학과 학사  
 2003년 ~ 현재 삼성전자 디지털 미디어 및 통신 연구소  
 선임연구원

관심분야: Protocol, Codec관련, 보안, Embedded System & Platform

**김 선 발**

