CABLE-TEC EXPO® 2017

SCTE · ISBE

THE NEXT BIG...

DEAL
CONNECTION
INNOVATION
TECHNOLOGY
LEADER
NETWORK

SCTE·ISBE CABLE-TEC EXPO 2017

DENVER, CO
OCTOBER 17-20

2017 Fall Technical Forum
SCTE·ISBE • NCTA • CABLELABS

SCTE · ISBE

Addressing IP Video Adaptive Stream Latency and Video Player Synchronization

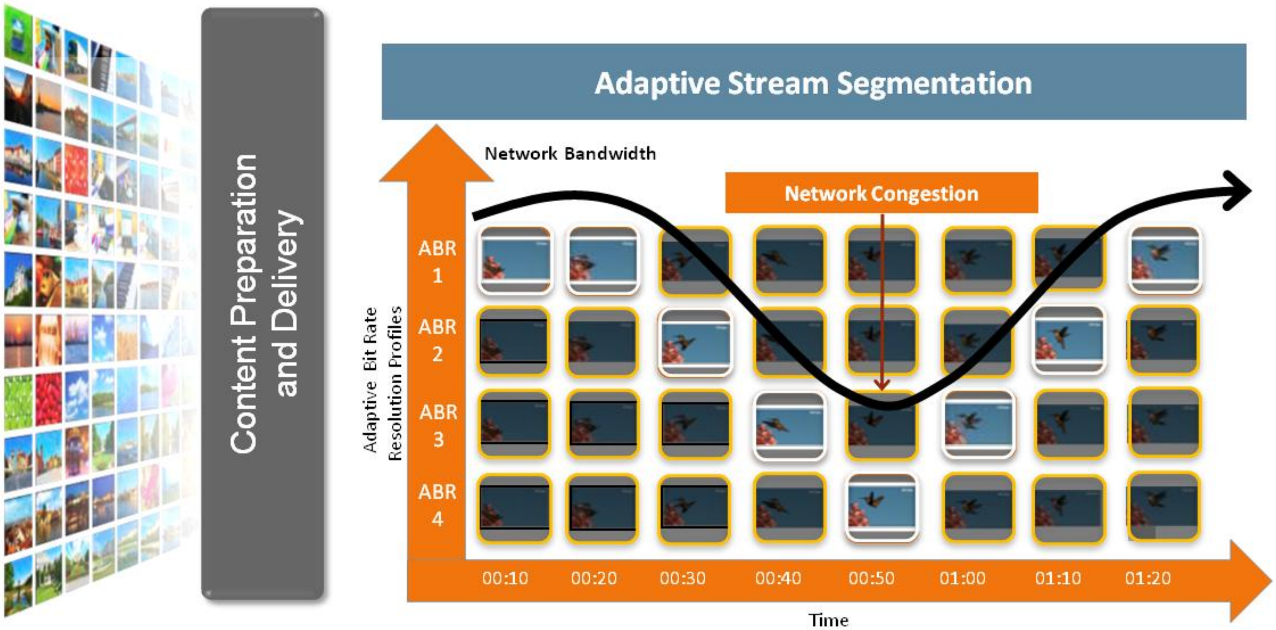
Jeff Tyre
Distinguished System Engineer
ARRIS

SCTE · ISBE CABLE-TEC EXPO 2017
DENVER, CO
OCTOBER 17-20

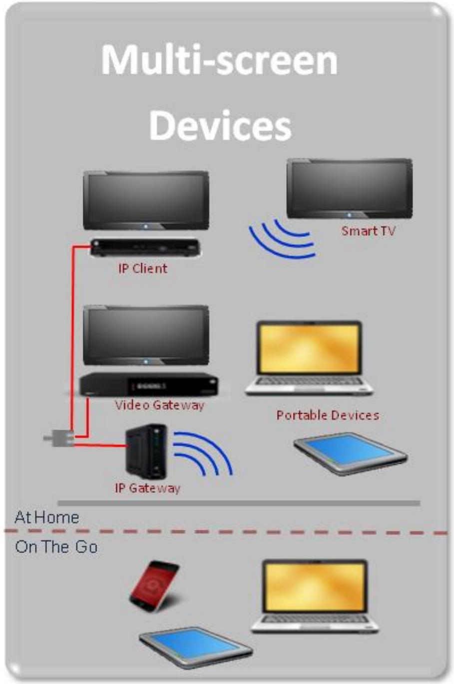
Multi-screen services driving new video technology adoption

- Multi-screen consumer devices use Web-based technologies
- Video is delivered over IP networks using Content Delivery Networks (CDN) based on HTTP / TCP
- Adaptive Bit-Rate (ABR) video streaming addresses “best-effort” delivery of IP networks
- Cloud-based service deployments can provide unification of subscriber, delivery, analytics and scalability management

Adaptive Bit Rate (ABR) video streaming



HTTP ABR profile package is created for each TV channel service



Gaps - Live Broadcast TV Service vs ABR Delivery

Live / Linear TV Services

A real-time timeline that is used by remote viewers, such as a sports event.

Viewing delay caused by content acquisition, editing, processing and delivering, as well as mandatory regulation delay request. Minimum delay is desired.

Content is delivered to a large number of viewers simultaneously.

Consistent viewing experience for all viewers is a key goal.

However, ABR Services are Based On

HTTP-based adaptive bit rate streaming was not designed for Live / Linear TV service.

Segment length design, segmentation delay and relationship to video coding efficiency.

ABR players act independently and lack consistent or synchronized content playout.

Key challenges for ABR video in support of Live TV services

ABR Content Preparation and Delivery Latency

ABR processing latency due to ABR content package segment duration, typically 2-10 seconds

ABR content playout latency due to ABR player buffering of ABR content segments

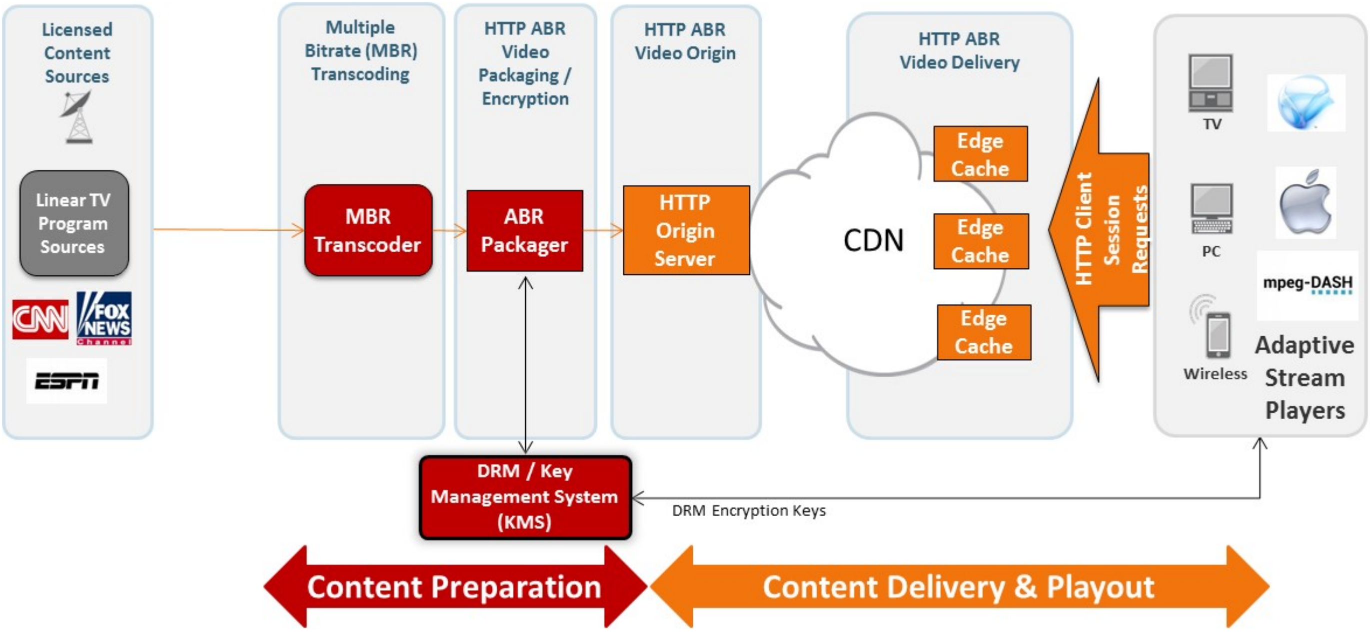
Video Playout Synchronization

ABR players use HTTP sessions to independently request ABR content from origin

HTTP typically based on a PULL content delivery model

ABR player sessions are inherently un-synced

ABR Content Preparation and Delivery – HTTP client “Pull”



Primary sources of latency

Content Preparation

- Encoder / MBR Transcoder
  - < 2s, most segment sizes
- ABR Segment Packaging
  - ~6-30s, per segment size

Content Delivery / Playout

- HTTP Origin / CDN Cache
  - ~6-30s, per segment size
- ABR Player buffering
  - ~2-30s, per segment size

Content preparation and delivery design goal – Push, don't Pull

Legacy video broadcast and IP multicast transport streams (TS) are bandwidth efficient

- Minimum end to end latency based on a true video streaming function => TS "Push" to player
- IP multicast provides a content delivery framework supporting video player synchronization

Challenge is that next gen Multiscreen / ABR players don't support TS or IP multicast

HTTP sessions are file-based transactions, naturally inclined to introducing processing latency

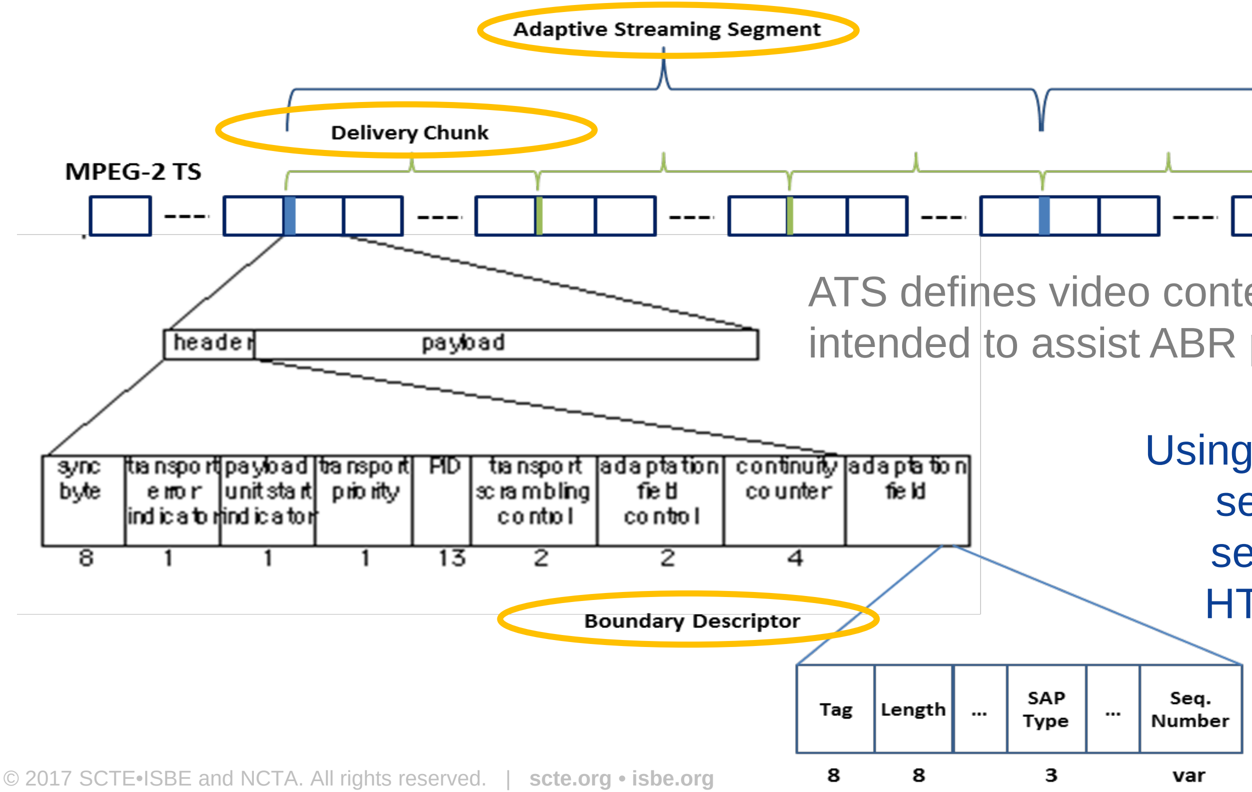
HTTP CDNs are based on multiple PULL transactions of ABR segments, typically 2s size or more

Latency reduction goal is to process smaller content packages

Goal is to define an ABR PUSH content preparation and delivery model

"*Minimizing Delivery Latency of Linear Video Service via Adaptive Transport Stream and HTTP Chunked Transfer Encoding*" – Wendell Sun, SCTE Cable-Tec Expo, 2016 technical paper

Adaptive Transport Streams (ATS) – SCTE and MPEG / ISO



ATS defines video content boundary descriptors intended to assist ABR packaging

Using video chunk-level delivery and segment-level ABR profile bitrate selection can help address ABR / HTTP latency delivery challenges

HTTP Chunk Transfer Encoding (CTE)

CTE is a data transfer mechanism in HTTP 1.1, in which data is sent in a series of "chunks".

CTE uses the Transfer-Encoding HTTP header in place of the Content-Length header.

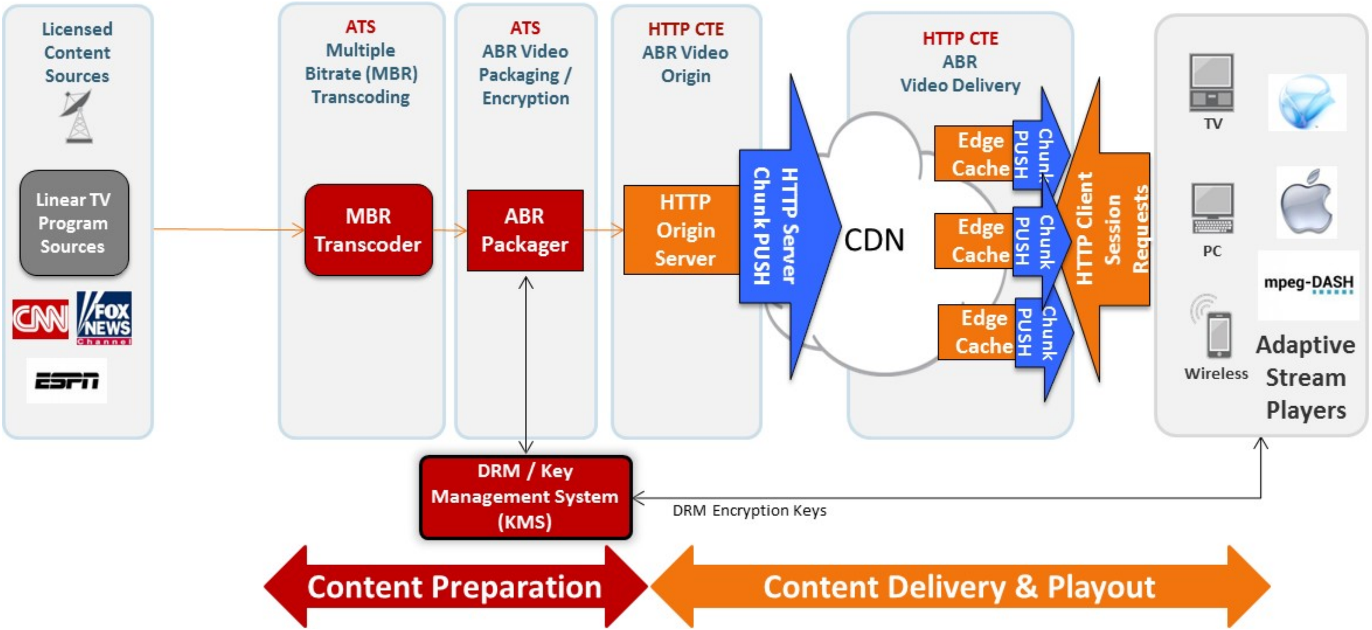
- Since the Content-Length header is not used, the sender does not need to know the length of the content before it starts transmitting a response to the receiver.
- Senders can begin transmitting dynamically-generated content before knowing the total size of that content.
- Size of each chunk is sent right before the chunk itself so that the receiver can tell when it has finished receiving data for that chunk. Data transfer is terminated by a final chunk of length zero.

HTTP CTE is a potential optimization technology to use for ABR / HTTP video content delivery

Compliments ABR player HTTP Segment “Pull” with a HTTP CTE Server Chunk “PUSH”

ABR video content delivery can be very similar to IP video transport streaming in terms of latency

ABR Content Preparation and Delivery – HTTP CTE server "Push"



Primary sources of latency

Content Preparation

- Encoder / MBR Transcoder
  - < 2s, most segment sizes
- ABR Segment Packaging
  - < 1s, per chunk size

Content Delivery / Playout

- HTTP Origin / CDN Cache
  - < 1s, per chunk size
- ABR Player buffering
  - < 1s, per chunk size

ABR Manifest File Formats and Unsynchronized Playback

Existing HTTP based adaptive streaming use manifest file and segmented content downloading

- Media being prepared / conditioned into well aligned multi-bit rate streams
- Each of the multibit rate streams is packaged to small segments and placed on HTTP origin server
- A manifest file is created to describe structure, location and availability of these media segments
- ABR player first requests the ABR manifest file, then starts to pull ABR media segment one after the other adaptively per available bandwidth
- Each pulling is a separate HTTP file transfer transaction; each ABR player acts independently

For Live / Linear TV services, all ABR players should request the same media segment simultaneously, supporting a synchronized viewing experience.

But in practice, this does not occur when individual ABR players initiate their HTTP session requests from the ABR manifest playlist published for managed TV services.

ABR Manifest File Formats and Unsynchronized Playback

Example using 10s ABR segments

At the moment of 12:00:00	At the moment of 12:00:07	At the moment of 12:00:10
#EXTM3U	#EXTM3U	#EXTM3U
#EXT-X-TARGETDURATION:10	#EXT-X-TARGETDURATION:10	#EXT-X-TARGETDURATION:10
#EXTINF:10,	#EXTINF:10,	#EXTINF:10,
./segment**3511**.ts	./segment**3511**.ts	./segment**3512**.ts
#EXTINF:10,	#EXTINF:10,	#EXTINF:10,
./segment**3512.**ts	./segment**3512**.ts	./segment**3513**.ts
#EXTINF:10,	#EXTINF:10,	#EXTINF:10,
./segment**3513**.ts	./segment**3513**.ts	./segment**3514**.ts
#EXT-X-ENDLIST	#EXT-X-ENDLIST	#EXT-X-ENDLIST

Proposed ABR Manifest File Format for Synchronized Playback

For a Live / Linear TV services, a manifest file only lists different ABR bit rate profile representations, but not individual ABR media segment URLs.

A single "virtual" segment URL can be defined for selection of each ABR bit rate profile.

For example:

```
#EXTM3U
#EXT-X-TARGETDURATION:10
./segment-720p-3000kbps.ts
#EXTINF:10,
./segment-480p-800kbps.ts
#EXTINF:10,
./segment-320p-500kbps.ts
#EXTINF:10,
./segment-0kbps.ts
#EXT-X-ENDLIST.
```

Proposed ABR Manifest File Format for Synchronized Playback Combined with HTTP CTE Server with ATS Defined Chunks

HTTP CTE server receives ATS prepared media segments and maintains the status of available ABR media segments and their chunks per ABR player tiered bit rate requests.

At any moment, there is only one ABR media segment currently available to any ABR player requesting a given ABR bit rate stream requested, matching the current media presentation time, for each ABR stream's bit rate profiles

For example 720p-3000kbps, 480p-800kbps, 320p-500kbps.

This "current segment" definition moves along with the timeline of media presentation.

In the example next slide, the segments with the "*" mark are the current segment.

Proposed ABR Manifest File Format for Synchronized Playback

Example using 10s ABR segments

Current segment 720p profile

Current segment 480p profile

Current segment 320p profile

At the moment of 12:00:00	At the moment of 12:00:07	At the moment of 12:00:10
#720p-3000bps	#720p-3000bps	#720p-3000bps
./segment3511.ts	./segment3511.ts	./segment3512.ts
./segment3512.ts *	./segment3512.ts *	./segment3513.ts *
./segment3513.ts	./segment3513.ts	./segment3514.ts
#480p-800bps	#480p-800bps	#480p-800bps
./segment3511.ts	./segment3511.ts	./segment3512.ts
./segment3512.ts *	./segment3512.ts *	./segment3513.ts *
./segment3513.ts	./segment3513.ts	./segment3514.ts
#320p-500bps	#320p-500bps	#320p-500bps
./segment3511.ts	./segment3511.ts	./segment3512.ts
./segment3512.ts *	./segment3512.ts *	./segment3513.ts *
./segment3513.ts	./segment3513.ts	./segment3514.ts

Note that ABR Players select bit-rate tier at segment level;
HTTP CTE deliver ABR content at ATS-defined chunk level

Key summary points

Adopting ATS and HTTP CTE, ABR latency is no longer a segment size issue since it can be supported at the chunk level with very small video content units / video frame

- Leverages a PUSH model for ABR content delivery similar to Transport Streams, addressing ABR delivery latency
- ABR playout from the local player buffer is also minimized by processing chunks, not segments

Removing the specific segment URLs from ABR manifest and replacing with a tiered ABR bitrate selection playlist separates the two key functions of ABR content delivery

- ABR players retain the function of ABR profile bitrate selection per network condition at the ABR segment selection level
- Moves the function of ABR media delivery management to the network server allowing ABR media playout synchronization to be supported across a large population of ABR players

THANK YOU!

Jeff Tyre
jeffrey.tyre@arris.com
408-940-2095

Wendell Sun
wendell.sun@viasat.com
760-893-5577