WebRTC-based Premium Streaming Ecosystem

DASH-IF, 28 May 2021
Presenters

Bill Wishon - Chief Product Officer - Phenix Real Time Solutions

Ofer Shem Tov - Distinguished Architect - Verizon Media

Nate Burr - Sr. Principal Software Development Engineer - Verizon Media
Agenda

1. Use Cases
2. Ecosystem & Gaps
3. Service Discovery & Negotiation
4. Other Ecosystem Topics
   a. DRM
   b. Ad Insertion
5. Next steps
Streaming Latency: It’s a continuum

**Streaming latency continuum**

- **High latency**
- **Typical latency**
  - For non-HLS HTTP adaptive streaming formats
- **Typical latency**
- **Reduced latency**
  - Using traditional streaming and tuned HTTP adaptive streaming
- **Low latency**
  - Live sports events
  - Second screen experiences
  - Breaking news
- **Very-low latency**
  - Video surveillance
  - Trivia/quiz
- **Real time**
  - Voice chat
  - Video gaming
  - Gambling
  - Betting
  - Online auctions
  - Web conferencing

---

**Latency Continuum**

- HLS
- Smooth streaming, HDS, DASH
- RTMP
- RTP/RTSP, WebRTC
- HLS/DASH CMAF with LLC

<table>
<thead>
<tr>
<th>Latency Level</th>
<th>Time (seconds)</th>
</tr>
</thead>
<tbody>
<tr>
<td>High</td>
<td>60+</td>
</tr>
<tr>
<td>Typical</td>
<td>45</td>
</tr>
<tr>
<td>Typical</td>
<td>30</td>
</tr>
<tr>
<td>Reduced</td>
<td>18</td>
</tr>
<tr>
<td>Low</td>
<td>05</td>
</tr>
<tr>
<td>Very-low</td>
<td>02</td>
</tr>
<tr>
<td>Real time</td>
<td>500 ms</td>
</tr>
</tbody>
</table>
Interactivity Requires Real-time

Examples of User Experiences

- Multi-angle user-selectable content, synchronized in real-time
- Conversations between hosts and viewers
- Co-watching of content with a group of friends (VOD & Live)
- Interactive news, where viewers add their own content to the story

Industry Verticals

- Gambling
- Gamification
- Sports
- Trivia / Online Quiz Games
- Auctions
- Online shopping

Interactivity

- host ⇔ viewers
- user ⇔ user
Premium Video in Production Use Cases
Quality & Scale in Real-time with WebRTC

Cheltenham Festival
Achieved 478,000 peak concurrent viewers

Grand National
10 races topped 100,000 peak concurrent users
Two topped 175,000 concurrent users
Two topped 225,000 peak concurrent users

Who Wants To Be A Millionaire
Mobile app companion
Over 100,000 peak concurrent players each week

The Oscars
Total views: 1.4M
Peak Concurrent Users = 114,000
Stream Join Rate = 80,000/sec

Q12 Trivia Game
Delivered 220,000 concurrent users on iOS and Android
10 weeks of 200,000+ concurrent users
What is WebRTC?

- Originally, created for real-time communication for the Web
- W3C Standard
- Supports video, audio & data streaming
- Built into all modern browsers across desktop and mobile devices
- Increasingly used today for real time streaming of premium content
Premium Streaming Ecosystem

Acquire
- SMPTE
- SDI
- SRT
- RTSP/RTP
- MPEG
- Transport Stream
- RTMP
- RIST
- NDI
- HDR

Adapt
- H.264
- MPEG-4 AVC
- AV1
- H.265
- VP8
- VP9
- DOLBY
- VISION • ATMOS
- DOLBY
- MPEG-H AUDIO
- DTS
- SMPTE 2095-1
- MIIO

Enhance
- SCTE
- n
- CEA
- MBR
- mpeg-DASH
- RTMP

Deliver
- WebRTC
- RTMP
- HLS

Consume
- Mac
- iOS
- Apple TV
- Windows
- XBOX
- Android
- PlayStation
- Linux
Ecosystem Gaps

- How a viewer discovers and joins an experience
- Session Negotiation
- Captions / subtitles
- Timed Metadata
- Ad Insertion
- Digital Rights Management (DRM)
- Advanced Audio & Video Codecs

Bridging these gaps with standard solutions will maximize interoperability and streamline development and adoption.
From Discovery to Streaming

Current and future states of how a viewer discovers and joins an experience

The manifest contains multiple ‘adaptation sets’ for camera angles, languages, etc.

Current state of Real-Time Streaming

Goal for Real-Time Streaming
Session Negotiation

**Today:** The specific transport method for signaling and session negotiation is left up to each application developer

**What’s needed:** Standard and specific transport method for signaling and session negotiation

**Goals:**

- Increase adoption by allowing participants to easily publish and subscribe to each other
- Minimize session establishment time while maintaining flexibility
Current State - No Specific Signaling or Transport Method

Publishing Peer

Create Offer()

Hello

SDP Offer

set remote description (SDP)

Answer SDP

Negotiate ICE

ICE Candidate

Send ICE candidate

Test ICE candidates

Connection established

Media Server

Signaling

Hello

SDP Offer

set remote description (SDP)

Create Answer

Answer SDP

Send ICE Candidate

ICE candidate

Test ICE candidates

Connection established

Publishing Peer

Signaling

Media Server
WebRTC HTTP-base Ingest Protocol (WHIP)

WHIP Scope
- Publish

WebRTC HTTP Session Negotiation Protocol (WHSNP)

WHSNP Scope
- Publish
- Subscribe
- TURN
- ICE
- RTC Config

Diagram:
- Peer
  - Create Offer
  - Negotiate ICE
  - Send ICE candidate
  - Test ICE candidates
  - Connection established

- Signaling
  - Hello
  - Answer SDP
  - SDP Offer
  - set remote description (SDP)
  - Create Answer

- Media Server
  - Hello
  - Answer SDP
  - SDP Offer
  - set remote description (SDP)
  - ICE candidate
  - Send ICE Candidate
  - Test ICE candidates
From Discovery to Streaming

Current and future states of how a viewer discovers and joins an experience

The manifest contains multiple ‘adaptation sets’ for camera angles, languages, etc.

Current state of Real-Time Streaming

Goal for Real-Time Streaming
Can we borrow your MPD for a minute?

Why would we want to use the MPD?

- Widely adopted as an industry standard
- Solves many perceived needs
- Allows code reuse
- Adds new use cases to DASH

DASH extension for real time interactive streaming!
WebRTC Representation

- Mime type: application/webrtc
- URL for session negotiation endpoint with required content ID
- Includes audio and video
- Audio language is specified using `lang=` in the adaptation set
  - There is not resource duplication since audio streams can be separated from video
- Codecs are signaled in WebRTC SDP (session description protocol)
  - Scalable video codecs are starting to be supported
- Captions can use CEA-608 and languages specified in adaptation set
- Do we need multiple representations?
  - Compatibility
  - Manual bitrate selection
Events and Timed metadata

Current state:

- Delivery of ad insertion markers, captions, other timed data is not specified
- Vendors use WebRTC's reliable bi-directional data channel to send this data via proprietary messages

Goal:

- Use standard DASH in-band events and MPD events, such as SCTE-35 events, sent over the data channel
- Custom events continue to use a standard format including schemeldUri, etc.
DASH and WebRTC use cases

- Initiate a complex WebRTC session with multiple streams
  - Using WebRTC representations with multiple adaptation sets for camera angles, etc.
- Fallback from WebRTC to HTTP streaming
  - Include Adaptations for both WebRTC and HTTP streaming
  - Players can prefer WebRTC and fallback to HTTP when WebRTC does not work
- Linear channels with interactive programs
  - Use WebRTC periods for interactive programs that require real time streaming
- Co-watching synchronized streams with audio/video chat
  - Live streams over WebRTC/HTTP or VOD over HTTP
  - Use MPD events for synchronization to fix timing issues caused by personalized SSAI
  - MPD will include information about connecting to the WebRTC-based interactive endpoint
Current WebRTC Security

Current Options:

- Secure Real-time Transport Protocol (SRTP) - Provides encryption, message authentication and integrity, and replay attack protection to the RTP data in both unicast and multicast applications

Future Options:

- Insertable Stream to E2EE -
  https://webrtc.github.io/samples/src/content/insertable-streams/endtoend-encryption/
DRM

Current Real-Time Options:

- WebRTC DataChannel to MSE/EME
- WebSockets to MSE/EME (*head-of-line* blocking)

Future Options:

- Insertable Stream to MSE/EME
- WebTransport to MSE/EME

**MSE / EME limitations:**

- CMAF isn’t ideal for real-time streaming. It’s a storage container format vs streaming format
- Current MSE/EME has some latency issues in different browsers.
DRM Cont.

Recommended Options:

- WebCodecs + EME (without CMAF packaging).
- DataChannel or WebTransport or Insertable Streams to WebCodecs and EME (without CMAF packaging).
- Client side demuxing is possible.
- AES-128 CBCS - Only Elementary stream is encrypted.
- Can follow a single encode and encryption path and multi-package approach compatible with DASH
Ad Insertion

Client-side

- SCTE event via Data Channel
- Ads via DASH or HLS and switch from WebRTC

Server-side

- Ad inserted directly into the WebRTC stream
Server-Side Ad Insertion in Real-Time

SSAI Situation Matrix

<table>
<thead>
<tr>
<th>Ad Inventory Lead Time</th>
<th>Days</th>
<th>Seconds</th>
<th>None</th>
</tr>
</thead>
<tbody>
<tr>
<td>Retrieval</td>
<td>Pre-fetch</td>
<td>Just-in-time</td>
<td>Real-time</td>
</tr>
<tr>
<td>Transcoding</td>
<td>Batch</td>
<td>Just-in-time</td>
<td>Real-time</td>
</tr>
<tr>
<td>Caching</td>
<td>Pre-cached</td>
<td>Deliver &amp; Cache</td>
<td>Delivered &amp; Cache</td>
</tr>
</tbody>
</table>

Cache Miss => Default Ad

Phenix SSAI Architecture

- Real-Time & Batch Transcoding
- Ad Engine
- Streaming Engine
- Cache

Ad Servers:
- Google
- FreeWheel
- In House

Viewers & Ad Beacons

Video & Ad Content
Next steps

- Building industry awareness
- Identify standards development organizations for these efforts
- Industry review & feedback
  - Session Negotiation
  - DASH Interop
  - WebRTC Security
Questions?