NMOS
Networked Media Open Specification
The Open AV over IP RESTful API
Andrew Starks
Director of Product Management, Macnica
2019/06/12
I am...

- Andrew Starks
- Director of Product Management for
  
- Formerly Co-Founder of Tightrope Media Systems (20 Years)
- *Extremely* happy to be here!
### SMPTE ST-2110 / NMOS: Layered Architecture

<table>
<thead>
<tr>
<th>Layer</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>Registration Discovery IS-04</td>
<td></td>
</tr>
<tr>
<td>Subscription IS-05</td>
<td></td>
</tr>
<tr>
<td>Network Control IS-06</td>
<td></td>
</tr>
<tr>
<td>...</td>
<td></td>
</tr>
</tbody>
</table>

#### Networked Media Open Specification (NMOS)

Network-Transparent (RESTful) Application Programming Interface (API)

<table>
<thead>
<tr>
<th>Layer</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>Timing and Synchronization -10</td>
<td></td>
</tr>
<tr>
<td>Uncompressed Video -20</td>
<td></td>
</tr>
<tr>
<td>Uncompressed Audio -30 (AES67)</td>
<td></td>
</tr>
<tr>
<td>Ancillary Data -40</td>
<td></td>
</tr>
</tbody>
</table>

#### ST 2110

Low Latency, Synchronized Media Stream Transport
SMPTE ST-2110 / NMOS: Layered Architecture

Registration Discovery IS-04
Subscribe IS-05
Network Control IS-06
...

Networked Media Open Specification (NMOS)
Network-Transparent (RESTful) Application Programming Interface (API)

Timing and Synchronization -10
Uncompressed Video -20
Uncompressed Audio -30 (AES67)
Ancillary Data -40

Transport

Low Latency, Synchronized Media Stream Transport
## SMPTE ST-2110 / NMOS: Layered Architecture

### Networked Media Open Specification (NMOS)
Network-Transparent (RESTful) Application Programming Interface (API)

### Timing and Synchronization
- Uncompressed Video
- Uncompressed Audio
- Ancillary Data

### IS-04
- Registration Discovery

### IS-05
- Subscribe

### IS-06
- Transport Control

### ST 2110
Low Latency, Synchronized Media Stream Transport
<table>
<thead>
<tr>
<th>ID</th>
<th>Name</th>
<th>Status</th>
</tr>
</thead>
<tbody>
<tr>
<td>IS-04</td>
<td>Discovery and Registration</td>
<td>Stable</td>
</tr>
<tr>
<td>IS-05</td>
<td>Device Connection Management</td>
<td>Stable</td>
</tr>
<tr>
<td>IS-06</td>
<td>Network Control</td>
<td>AMWA Specification</td>
</tr>
<tr>
<td>IS-07</td>
<td>Event &amp; Tally</td>
<td>AMWA Specification</td>
</tr>
<tr>
<td>IS-08</td>
<td>Audio Channel Mapping</td>
<td>AMWA Specification</td>
</tr>
<tr>
<td>IS-09</td>
<td>System</td>
<td>Work in Progress</td>
</tr>
<tr>
<td>MS-04</td>
<td>ID &amp; Timing Model</td>
<td>Finalized, Approved</td>
</tr>
<tr>
<td>BCP-002-01</td>
<td>Natural Grouping</td>
<td>Finalized, Approved</td>
</tr>
<tr>
<td>BCP-003-01</td>
<td>API Security: Communications</td>
<td>AMWA Specification</td>
</tr>
<tr>
<td>BCP-003-02</td>
<td>API Security: Authorization</td>
<td>Finalized</td>
</tr>
<tr>
<td>N/A</td>
<td>Parameter Registers</td>
<td>Continuing</td>
</tr>
</tbody>
</table>
- NMOS Node API
- NMOS Registration API
- NMOS Query API
  - m/DNS Search and Discovery
  - Peer-to-Peer Search and Discovery
NMOS Node

![Diagram showing a flow between a sender and a receiver with a flow identifier: B1847B0-98F7-4B9F-A197-3DA3918225B]
NMOS Model

- **Devices:** do things
  - Play, capture, transform, display, etc.

- **Nodes:** their logical hosts
  - Could be a server or other container
Other Specifications

- IS-04 NMOS Node
- IS-05 Connection Manager
- IS-07 Tally & Event
- IS-08 Audio Mapping
Devices can have Senders and/or Receivers which are logical inputs/outputs.
Flows are time stamped (synchronous) sequences of video/audio/ancillary data that pass from a Sender to a Receiver(s).
Devices, Sources, Flows, Senders, Receivers - they are Resources

- **Uniquely identified** and addressable parts of a NMOS environment
- A Node exposes its Resources via its IS-04 Node API
NMOS Example: HDMI Gateway

HDMI Source to 2110 Gateway

HDMI Input
- video sender
- audio sender
- anc data

HDMI to 2110 Gateway

Video Flow 2110-20
Audio Flow 2110-30
ANC Flow 2110-40

Video device
- receiver

AV device
- video receiver
- audio receiver
- anc receiver (CC, CEC, USB)
NMOS API: Web Friendly

- HTTP/S
- Websockets
- JSON
- UUIDs v4

JSON representation of a Device resource
IS-04 Registration Service

node
device

source
flow
receiver
sender

registry

Registration API
Query API
Discovery IS-04 Services

- Registration and Query: **mDNS announcements** (Bonjour, Avahi etc) or through **DNS entry**
- **Multicast / Unicast** supported
- Finding Registration service announcements:
  - Finds endpoint via **_nmos-register._tcp**
  - Register: **POST** request to a Registration API
  - Persist: **POST**ing time to Registration API / health endpoint every 5 sec
  - Register its Resources with further **POSTs** to the Registration API /resource
Peer-to-Peer

- The Node announces itself with `_nmos_node._tcp`
- Clients find its resources directly through the NMOS IS-04 Node API
- Continue to looking for Registry announcements and use registry instead, if available! (it’s better)
Provides a general mechanism for connecting senders and receivers:

- Supports unicast/multicast
- Provides feedback on success/failure
- Best used in conjunction with IS-04
NMOS IS-05: Connections
• **NMOS IS-06**: Switch control API

• **Broadcast Controller**: Plays the role of routing switcher software
  • Touches IS-04/05/06

• **Timing**: PTP Clocks keep things source synchronized. Work being done to support destination synchronization.

• **Multicast & Unicast**: Fully Supported

• **MUCH MORE TO COME!**
Thank you