Applying NMOS to non-2110 Environments

Chris Lennon

NMOS is great

- Networked Media Open Specifications
- Terrific work by a collection of great minds in AMWA
- Crucial to 2110 interoperable control
- What if we extended NMOS to other non-2110 signal types?

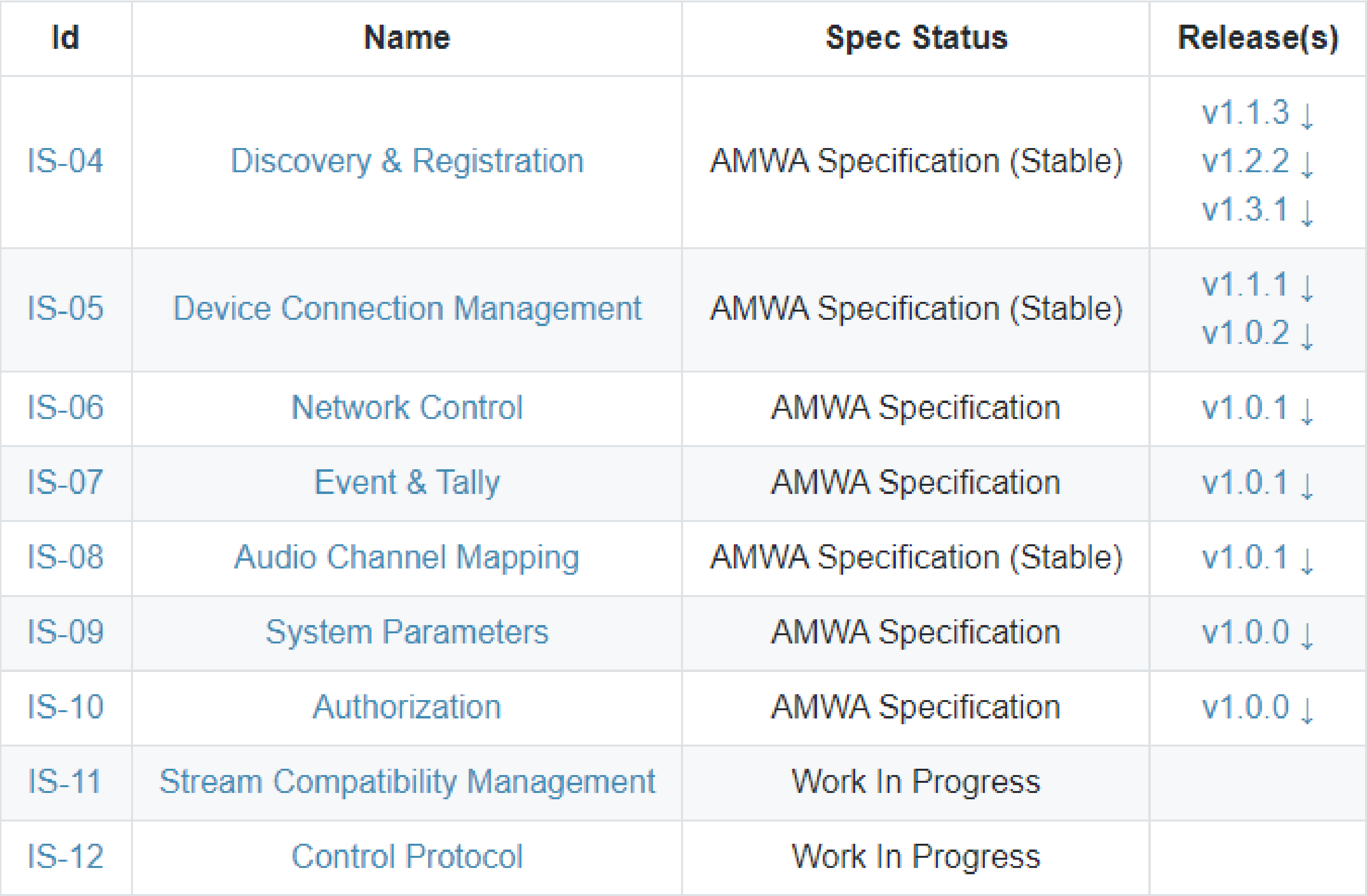
networked media NMOS open specifications

AMWA Networked Media. Working.

Extending to other signal types

- Today we lack unified open discovery and connection management across signal types
  - Systems are more complex than they should be
  - Multiple control systems are required
  - Slows us down
  - Costs more

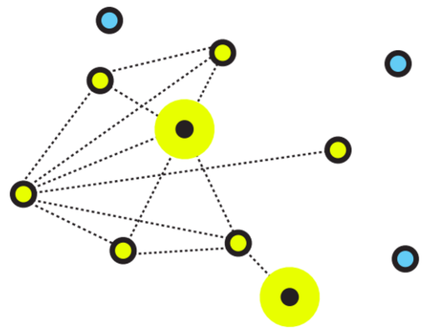
NMOS Interface Specifications



Id	Name	Spec Status	Release(s)
IS-04	Discovery & Registration	AMWA Specification (Stable)	v1.1.3 ↓ v1.2.2 ↓ v1.3.1 ↓
IS-05	Device Connection Management	AMWA Specification (Stable)	v1.1.1 ↓ v1.0.2 ↓
IS-06	Network Control	AMWA Specification	v1.0.1 ↓
IS-07	Event & Tally	AMWA Specification	v1.0.1 ↓
IS-08	Audio Channel Mapping	AMWA Specification (Stable)	v1.0.1 ↓
IS-09	System Parameters	AMWA Specification	v1.0.0 ↓
IS-10	Authorization	AMWA Specification	v1.0.0 ↓
IS-11	Stream Compatibility Management	Work In Progress	
IS-12	Control Protocol	Work In Progress	

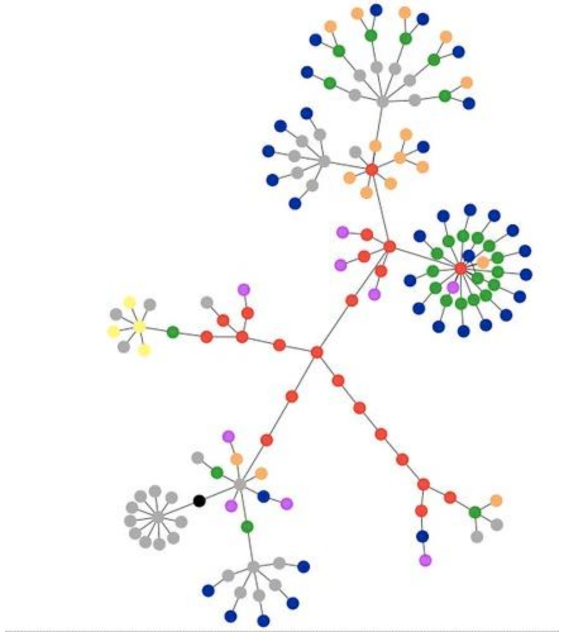
Discovery and Registration (IS-04)

- This is a critically important capability
- Devices can describe 2110 streams they have available so that NMOS-capable controllers can manage them.
- This is how you automate setup of networked systems, and dynamically deploying devices



Device Connection Management (IS-05)

- In conjunction with IS-04, this is the key to dealing with devices in a 2110 environment
- Provides transport-independent way of connecting media nodes via 2110 (RTP, WebSocket, and MQTT)
- Fills a gap left in ST2110



AMWA Implementation Specifications

- Fill gaps in the ST2110 suite

- How are streams controlled?

Is 2110 all you do?

- Probably not
- What to do for media on the cloud, for instance?
- What about other environments, particularly in cases where you have many transports you need to support?

IS-04 and IS-05

- In hybrid environments, you ideally want to be able to discover, register and manage all devices available to you.

Transports Currently Supported

- In v1.3, NMOS supports:
  - Real-time Transport Protocol (RTP)
  - RTP Multicast
  - RTP Unicast
  - Dynamic Adaptive Streaming (DASH) over HTTP
  - Message Queuing Telemetry Transport (MQTT)
  - WebSocket

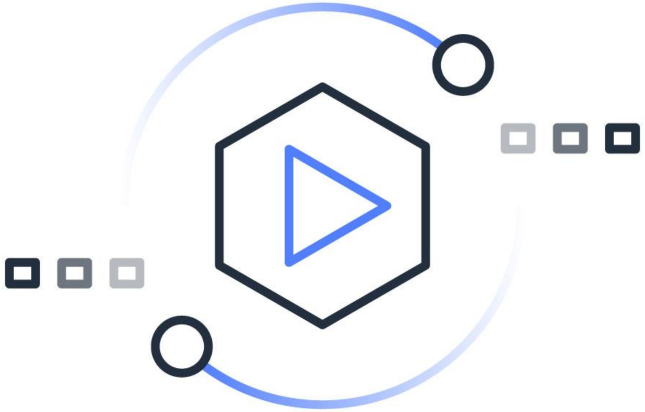
NDI – Network Device Interface

- Free SDK available
- Rapid and wide adoption
- Not a standard, but becoming a de facto standard

CDI – Cloud Digital Interface

- Amazon Web Services' answer to transporting uncompressed video inside of their cloud

- With AWS' widespread usage in the media industry, this is important



IPMX

IP MEDIA EXPERIENCE

- ProAV has long been devoid of standardization, due to domination by a handful of vendors, all using HDbaseT
- Whereas broadcast has embraced fixed video formats (e.g. 720p, 1080i, 2160p), ProAV includes everything from VGA to 8K and everything imaginable in between
- The situation for audio is similar

IPMX

IP MEDIA EXPERIENCE

- Users are often untrained, as opposed to professional engineers in the broadcast space
- Many, many use cases and verticals
- Traditionally, very little cross-vendor interoperability

IPMX

IP MEDIA EXPERIENCE

- ST2110 is not seen as the answer for ProAV
- However, 2110 with compression and a few more ProAV-specific features fits the bill nicely
- Attempts to provide standardized means of transporting audio and video over IP in the ProAV space (one which has resisted standardization thus far)
- ST2110 and NMOS form the foundation for transport and control infrastructure

What about other transports?

SRT

SECURE
RELIABLE
TRANSPORT

RIST

RELIABLE INTERNET STREAM TRANSPORT

Can we use IS-04 and IS-05 for these other transports?

Making IS-04/IS-05 work with NDI…

- Create new transport type:
    - urn:x-nmos:transport:ndi

- Sender:

```
[
"transport_params": [{
"connected": false,
"sender_name": "NDI Sender 2"
}]
]
```

Making IS-04/IS-05 work with NDI…

- Receiver (can receive one or more essence types):

```
[
"transport_params": [{
"allow_video_fields": true,
"bandwidth": "highest",
"color_format": "fastest",
"sender_name": "Camera01",
"rec_anc": false,
"rec_audio": false,
"rec_video": true
}]
]
```

What about non-IP streaming formats?

- Hold that thought*

*does require IP connectivity on the endpoints

What about non-IP transports?

- IS-11 Stream Compatibility Management (in progress)
  - Inputs associated with Senders
  - Outputs associated with Receivers
  - Can config parameters of Senders/Inputs with Receivers/Outputs
- Applicable to SDI devices, even HDMI devices
- Designed to be used alongside IS-04 and IS-05

Authorization (IS-10)

- Without security, what do we have?
- Applicable to securely control devices, despite the media transport they use underneath, allowing for a secure remote control of the infrastructure described by it
- Being widely embraced as a great approach to making sure that only authorized requests are allowed

AUTHORIZED

IS-10 - Authorization

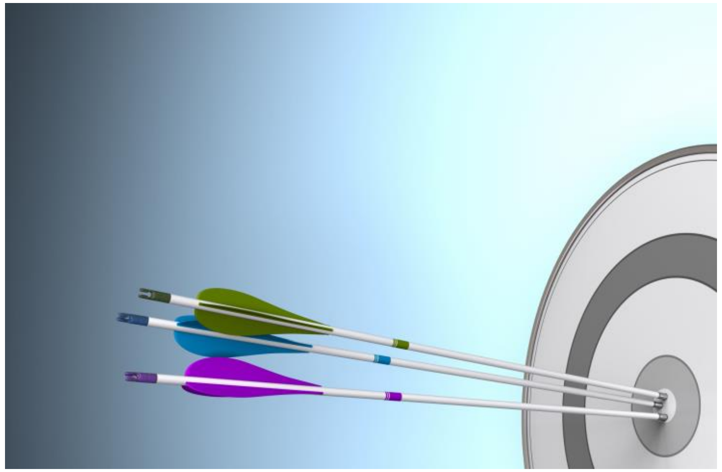
- Based on Oauth 2.0 and JSON Web Tokens
  - Widely accepted industry standards

- Secures NMOS portion of your ecosystem

- Need to think about other endpoints and securing those (e.g. NDI devices)

Conclusion

- NMOS has the potential to provide unified control over signal flows of all types
- Simplifies system design
- Flexibility to work on prem and in the cloud
- Move faster, save complexity and money



Any Questions?

IP SHOWCASE™