Using AMWA IS-06 for Flow Control on Professional Media Networks

Rob Porter
Sony Europe B.V.

Sachin Vishwarupe
Cisco Systems Inc.

Overview

- What are the AMWA NMOS specifications?
- What is AMWA IS-06?
- IS-06 v1.0 specification
  - Topology discovery
  - Setting up media flows
- Future work
  - Telemetry and monitoring
  - Securing the APIs
  - Grouping
- Proposed extension for Network Address Translation (NAT)
  - Use cases
  - Proposal

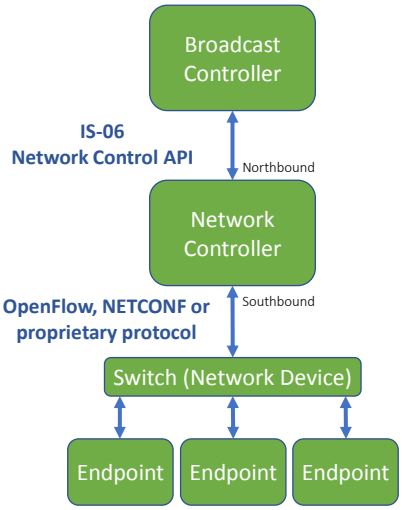
What are the AMWA NMOS Specifications?

- Networked Media Open Specifications
  - Set of **open APIs** for managing devices on professional media networks
  - Allow **interoperability** between different manufacturers' devices
  - Use standard **RESTful APIs** using HTTP GET, PUT, PATCH, DELETE with JSON payloads and WebSockets for notifications of updates

- IS-04 – Discovery and Registration
- IS-05 – Connection Management
- IS-06 – Network Control
- IS-07 – Event and Tally
- IS-08 – Audio Channel Mapping
- IS-09 – System
- IS-10 – Authorisation
- BCP-002 – Grouping
- BCP-003 – Security

What is AMWA IS-06?

- IS-06 defines the **Network Control API**
- **Northbound API** exposed by Network Controller
- Used by Broadcast Controller to "**reserve and secure network services**"
- Allows Broadcast Controller to:
  - Discover **network topology**
  - Set up **media flows** between senders and receivers (endpoints) on the network
    - Reserve bandwidth for media flows
    - Enforce network security by only allowing authorised flows, senders and receivers
- Supports use of **Software Defined Networking** (SDN)



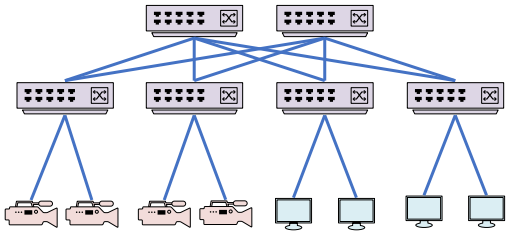
Topology Discovery

- Broadcast Controller is able to request network topology information from Network Controller using two API calls:

  IS-06 Network Device API

  IS-06 Network Link API

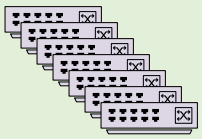
- Broadcast Controller can use these to get information to represent network topology on its UI



IS-06 Network Device API

GET /network-devices

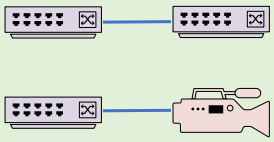
- Returns all the **networks devices** (switches) on the network
- Information returned includes:
  - **device_id**
  - **chassis_id**
  - **mgmt_ip**
  - **mtu**
  - array of **interfaces**
- Information returned for each **interface** includes:
  - **port_id**
  - **admin_status / oper_status**
  - **speed**



IS-06 Network Link API

GET /network-links

- Returns all the links:
  - Between connected network devices
  - Between network devices and endpoints



- In both cases, information returned includes:
  - For each of the two peers in the link:
    - **device_id**
    - **port_id**
  - NB: port_id must be null for endpoints

SONY CISCO

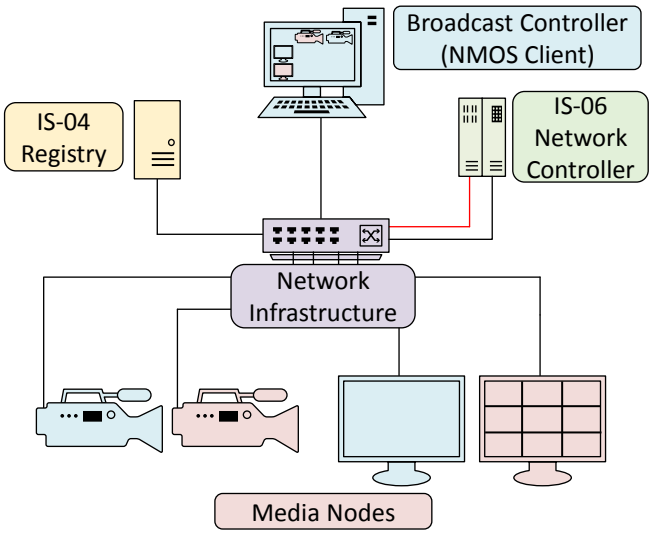
Setting up Media Flows

- Broadcast Controller uses two additional API calls to set up media flows:

IS-06 Endpoint API

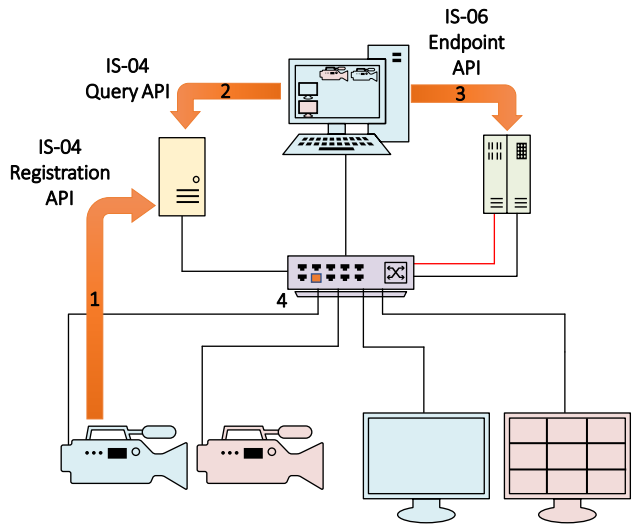
IS-06 Network Flow API

- Let's consider how these fit into the IS-04 / IS-05 workflow…



SONY CISCO

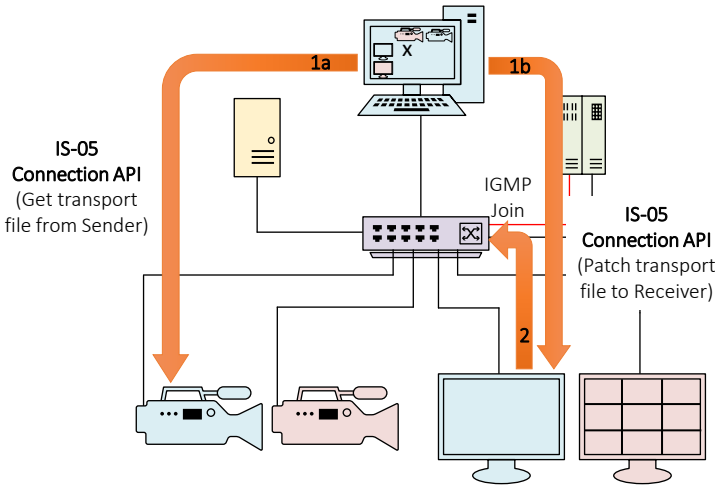
On Connecting a Media Node to the Network…



1. Node discovers **IS-04 Registration API** and registers itself and its sub-resources with the Registry
2. Client gets updated list of registered resources from Registry via **IS-04 Query API** WebSockets subscription
3. Client adds Node to Network Controller using **IS-06 Endpoint API**
4. Network Controller determines switch and port to which Node is connected using Node's **chassis_id** and **port_id**

SONY CISCO

To Set Up a Connection…



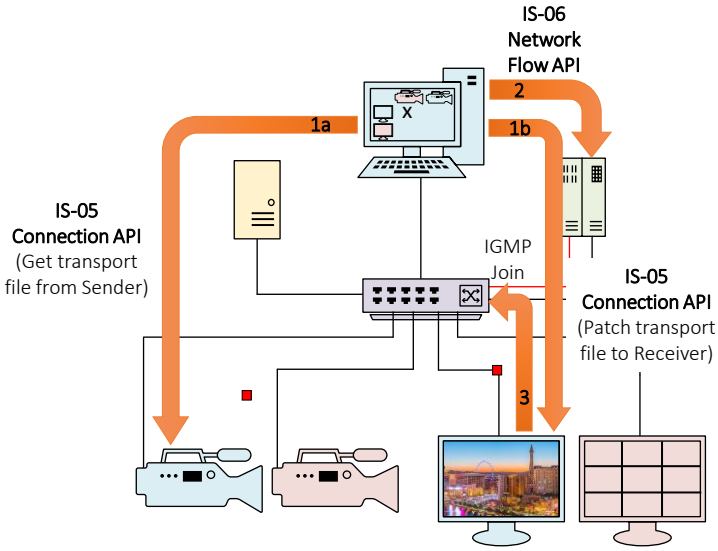
1. Client uses Node's **IS-05 Connection API**
   a. Get transport file from Sender
   b. Patch transport file to Receiver
2. **Without IS-06**:
   - Receiver issues IGMP join and connection is made

But…

… no authorisation or bandwidth control!

SONY CISCO

To Set Up a Connection…



1. Client uses Node's **IS-05 Connection API**
   a. Get transport file from Sender
   b. Patch transport file to Receiver
2. **With IS-06**:
   - Client makes explicit request to Network Controller to set up flow using **IS-06 Network Flow API**
   - If bandwidth available, Network Controller sets up flow from Sender to Receiver
3. Optionally, IGMP Join can still be used to allow Receiver to signal timing of flow set-up

Benefits of IS-06 for Flow Control

- Flow Set-up
  - Flow is only set up **if explicitly requested by Client**
    - i.e. all flows are authorised by Broadcast Controller
    - No rogue IGMP Joins allowed
  - Network Controller can **ensure sufficient bandwidth is available** before setting up flows
    - Ensures network is not oversubscribed
  - It's an **open multi-vendor approach**
    - Clients only have to support one protocol

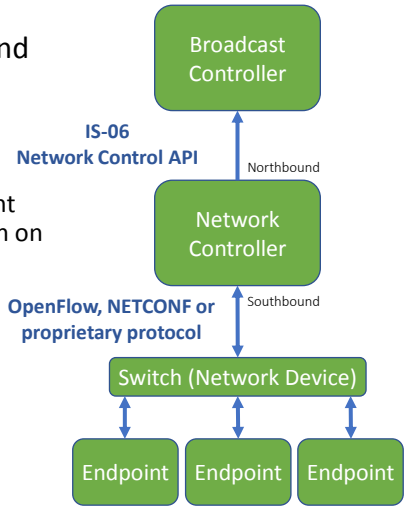
IS-06 Future Work – Monitoring

- **Provisioning Feedback** – Current IS-06 captures user's intent. Network controller may or may not be able to set up flows (insufficient bandwidth, no route available from sender to receiver, no management connectivity etc.)
- **Flow Statistics** – Get active vs dropped traffic
- **Logical Flow Topology** – End to end network topology for a given flow
- **Flow Classification** – Get active, sender-only, receiver-only, inactive flows in a network
- **Endpoint Classification –** Get list of endpoints discovered via (IGMP, ARP, Statically Configured etc.)
- **Network State –** Get current status of involved network devices, links.
- **Network Usage –** Aggregate link utilization per switch. Set node/link level utilization thresholds

SONY CISCO

IS-06 Future Work – Streaming Telemetry

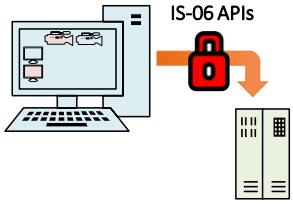
- Telemetry Interface will provide up-to-date network and flow state to northbound consumers (broadcast controllers)
- **Flow Telemetry**
  - Flow Established/Tear down. Consumer can make subsequent REST calls for additional information e.g. get logical flow path on flow established event
  - Flow Denied due to secure flow policy, insufficient BW etc.
  - Flow Statistics active vs dropped traffic
- **Network Changes**
  - Node/device up/down events
  - Link up/down events
  - Threshold crossing events



SONY CISCO

IS-06 Future Work – Security & Grouping

- Securing the APIs
  - Use of AMWA NMOS BCP-003 Best Practice recommendations for securing the IS-06 APIs
    - **BCP-003-01** for secure **communications** between Broadcast Controller and Network Controller
    - **BCP-003-02** for secure **authorisation**, to ensure Broadcast Controller and its operator have the required privileges to access the Network Controller
- Grouping
  - Ideally user may want to switch/set-up audio, video and ancillary data multicast streams together. Provide transactional APIs to achieve the same.
  - Provide batching for ”salvo” use case
  - Explore possibility of leveraging BCP-002 grouping tag construct



SONY CISCO

IS-06 Proposed Extension for Network Address Translation (NAT) – Use Cases

- AS A facility owner / broadcaster I NEED **an open method of applying network and port address translation in my network fabric** SO THAT…
  - I can support **sharing of resources between facilities in a flexible way**.
  - I can **resolve problems caused by address conflicts when sharing resources between facilities**.
  - I can flexibly support **contribution encoders and decoders which are set to fixed multicast addresses**.
  - I can support **legacy devices that are tuned to a specific multicast address** which cannot be changed using any non-proprietary protocols.
  - I can **redirect network flows to general network monitoring / packet capture equipment**.
- AS A manufacturer I NEED the API for applying network and port address translation to be open SO THAT I **do not have to implement and maintain several different network vendor-specific proprietary protocols**.

SONY CISCO

NAT Design Goals

- Leverage current IS-06 Models
- Model ”translation” as an independent object so IS-06 provider may or may not support it
- Support both IP and port address translation (NAT & PAT) for source & destination
- Support both TCP & UDP
- NAT policy scope can be network device or per interface
- Flexible
  - Support conditional translation
  - Selective translation e.g. source and/or destination (IP address and/or Port)
  - Multiple conditional & selective translation in single object
- Vendor Neutral

Example Policy 1 – Wildcard Translation


```
{
  "id": "57066e00-b777-4c7f-b8ea-e16b3eacb325",
  "description": "This policy translates source & destination if all match conditions are true",
  "match_source_ip": "10.1.2.3",
  "match_source_port": 10000,
  "match_destination_ip": "30.1.2.3",
  "match_destination_port": 30000,
  "translated_source_ip": "20.1.2.3",
  "translated_source_port": 20000,
  "translated_destination_ip": "40.1.2.3",
  "translated_destination_port": 40000,
  "receiver_endpoint_ids" : [
    "ac7c36a5-cd6d-44bb-a1ac-6e328c018214",
    "ac7c36a5-cd6d-44bb-a1ac-6e328c018215"
  ]
}
```


Demonstrates all possibilities of conditions and translations.
User can pick them selectively for matching and translation purpose

Example Policy 2 – Translation on Conditional Match


```
{
  "id": "58066e00-b777-4c7f-b8ea-e16b3eacb325",
  "description": "This policy translates only destination ip by default as no match conditions are provided (wildcard)",
  "translated_destination_ip": "50.1.2.3",
  "receiver_endpoint_ids" : [
    "dc7c36a5-cd6d-44bb-a1ac-6e328c018215"
  ]
}
```


Example of all incoming flows to these receiver endpoints are always translated to fixed IP and/or Port.

Policy Scope

- NAT URL
  - Global Scope
    - http://<network-controller-ip>/x-nmos/netctrl/v1.1/network-devices/<device-id>/nat/<nat-policy-id>
  - Per Interface Scope
    - http://<network-controller-ip>/x-nmos/netctrl/v1.1/network-devices/<device-id>/interfaces/<interface-id>/nat/<nat-policy-id>

Thank you

Rob Porter, Sony Europe B.V.
Rob.Porter@sony.com

Sachin Vishwarupe, Cisco Systems Inc.
vsachin@cisco.com

Thank you to our Media Partners