IP Conversion of Broadcasting

Part 4 | Control

NMOS: What is it and why do we need it?

The content of this document is as of July 2020.
The company and product names in this document are trademarks or registered trademarks of their respective holders.

Table of Contents

Introduction

Standards organizations including SMPTE and AES have created specifications for streaming of uncompressed video and audio over IP including things the ST2110 and ST2022-6 standards. These standards use RTP for video, audio and ancillary data over IP. However, none of these standards completely specify the control or application planes, leaving significant additional work to be done to achieve useful interoperability in professional networked media environments. To address this a number of industry bodies came together within the Joint Task Force on Networked Media (JT-NM) to coordinate how this might happen.

This is where the Advanced Media Workflow Association (AMWA) comes in. AMWA is an industry group of manufacturers, developers and end users that is advancing a software-focused approach to support future professional media operations. Out of this the Networked Media Open Specifications (NMOS) family of specifications were created to support the professional AV media industry's transition to a "fully-networked" architecture. NMOS is a set of specifications for control of IP networks like ST2110 & ST2022-6. It enables the use of items such as router control panels much as we do in SDI networks or keeping track of video sources and receivers in the network, and much more.

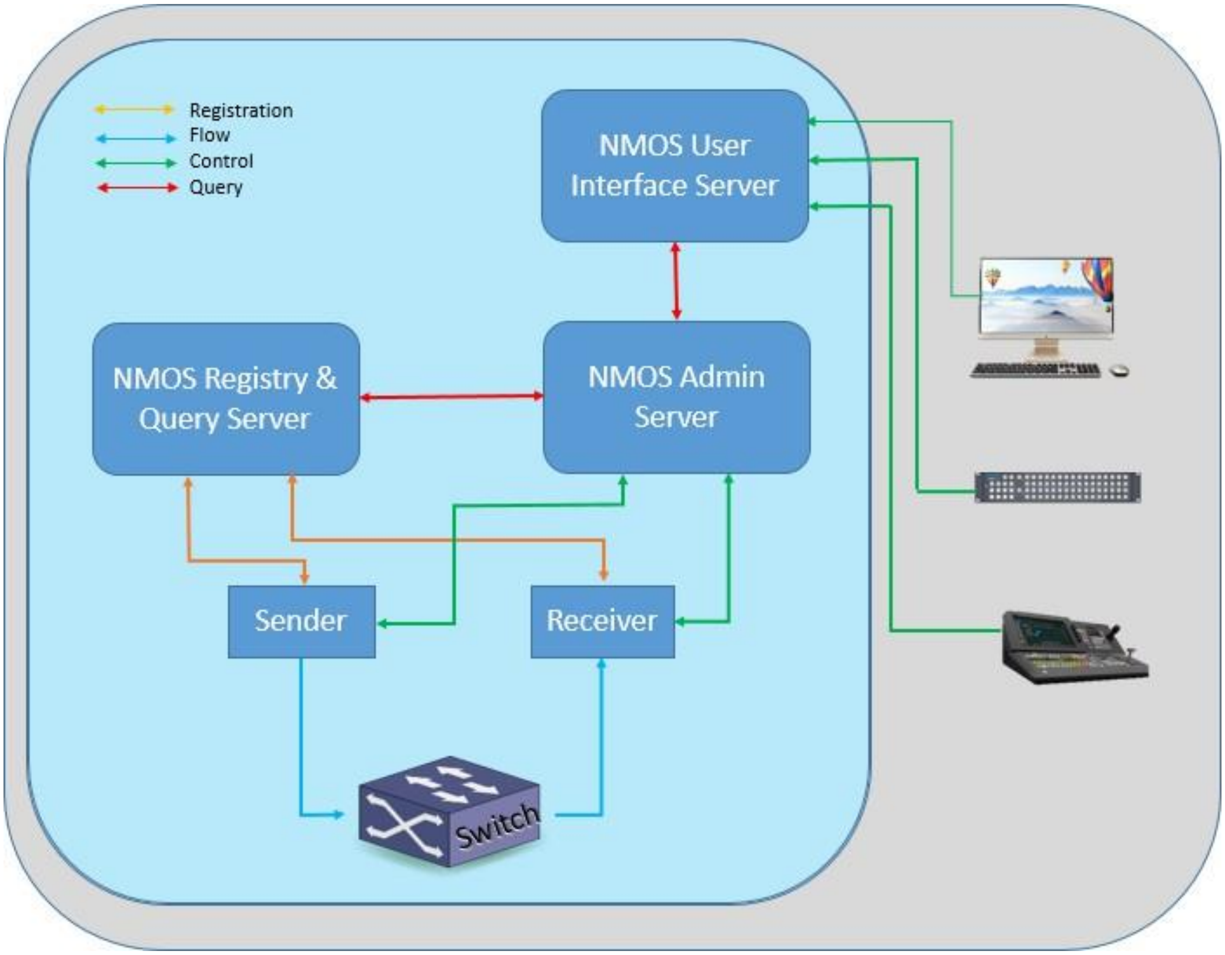
Glossary

In NMOS, several common terms have specific meanings that it helps to be aware of. Many of these correspond to the glossary of the JT-NM Reference Architecture.

API	Application Programming Interface provided over a protocol such as HTTP or WebSocket, defined in an AMWA NMOS Specification (IS-04, IS-05, IS-06, etc.).
AMWA	Advanced Media Workflow Association. AMWA is an industry group of manufacturers, developers and end users creating a software-focused approach to support future professional media operations.
Client	The entity that is using an API, for example: application using the IS-05 Connection API.
Controller	Client software that interacts with the NMOS APIs.
Device	A Device is a logical block of functionality within a networked media infrastructure; Sender (Camera), Receiver (Encoder, Waveform Monitor).
DNS	The Domain Name System (DNS). Each IP device on a network has a unique IP address which other machines use to find the device. DNS servers eliminate the need for humans to memorize IP addresses such as 192.168.1.1.
Flow	Flow refers to a sequence of video, audio or time-related data, the movement of Grains not the IP data itself in NMOS context.
GitHub	Is a code hosting platform for version control and collaboration. It lets you and others work together on projects from anywhere.
Grain	NMOS uses Grain as a convenient way of identifying a unit of video, audio or time-related data. For example, a video Grain could correspond to a frame of video.
IS-x	NMOS Interface Specifications IS-04, IS-05, IS-06, IS-07, IS-08, IS-09, IS-10, IS-11, IS-12.
Node	A Node is the host for one or more Devices.
NMOS	Networked Media Open Specifications. Is an open, free specification. NMOS provides for the management of video.
Post	The POST method requests the webserver to receive and process the data enclosed in the body of the POST message.
Receiver	A Receiver consumes a flow transmitted on the network by a sender, a monitor would be a Receiver for example.
Registry	Is a Registration & Discovery System (RDS) that provides the IS-04 Registration API for Nodes and the IS-04 Query API for Controllers.
REST	REST stands for representational state transfer. It is an application programming interface (API or web API) that allows for interaction with RESTful web services.
Sender	A Sender makes a Flow available on the network, a Camera for example.
Server	The entity that is providing an API, for example a Registry implementing IS-04 Registration and Query AP.
Source	Represents the logical origin of one or more Flows. It may be the Video, Audio, and Tally of a Sender.
User	Either human or automated operators driving a control system.
Query	A request for data, usually between Devices, Controllers, and Clients.

(This glossary complied from AMWA, Leader sources

Simplified block diagram of an NMOS system



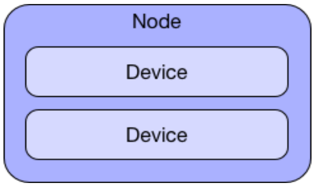
NMOS Overview

NMOS stands for ‘Networked Media Open Specifications’. NMOS is an open and free set of specifications. NMOS provides for the management of video, audio and data streams through specifying how to discover and control, and connect to network elements, how to send tally information, control audio channels and much more. NMOS is not software but a specification of how to do these things through an API that can be incorporated into a user’s software bundle. NMOS lets you find, connect and configure media devices with in your Audio Video networks.

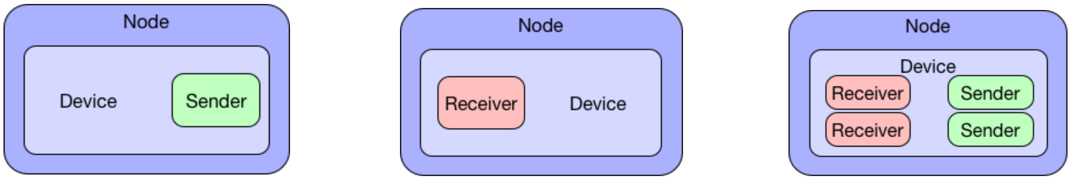
Building an NMOS-based system eliminates the need to manually share among all devices current IP addresses of every source and receiver on the network. It likewise removes the needed to manage the network solely through complex tables of IP addresses tracked on a spreadsheet. This can also help eliminates the confusing time-consuming UI configuration of Source, Destination addresses, and ports required on each device. Being able to discover what video resources (senders, receivers) are available gives you and devices the information needed to route Senders to Receivers. This system can also provide receivers the metadata needed to decode and present the data or content correctly using Session Description Protocol (SDP) files. Nonetheless there often still remains a need to give the server tied to the system’s user interface additional network detail because not all sources or receives provide the human-readable names and locations we want to match our naming conventions or topology.

NMOS Model and Terminology

In NMOS a Device represents a logical block of functionality, and a Node is the host for one or more Devices. Devices are used to represent "logical" things.



Devices have Logical input and outputs called Senders and Receivers.



Source: AMWA

A device can have one or more Senders or Receivers, if a device can receive 3 Video flows then there will be 3 Video Receivers. There will be Senders and/or Receivers for each Video, Audio and Anc flow.

NMOS Interface Specifications (IS)

AMWA followed a number of general principles when creating the NMOS specifications. They used Web-friendly protocols: NMOS uses REST HTTP API-based approaches and GitHub repositories to publish the specifications. These specifications are made public and are available at no cost to the industry.

Professional media has to work in many different environments, requiring a range of types of equipment. This means that NMOS specifications have been designed to work on many types of platforms including, low-power devices, rack-mounted, and virtualized.

Below are links to the AMWA site for each item. The latest information can be found at https://specs.amwa.tv/nmos/.

ID	Function / Name	Snapshot
IS-04	Discovery & Registration (stable)	Setup Sender and Receivers in the NMOS registry server (RDS)
IS-05	Device Connection Management (stable)	Provide the connection information I.E (SDP) File
IS-06	Network Control	Network bandwidth control in a managed network
IS-07	Event & Tally	Provide event information to Senders and Receivers
IS-08	Audio Channel Mapping (stable)	Change the channel mapping of Receivers or Senders
IS-09	System Parameters	Obtain global configuration parameters
IS-10	Authorization	Allows an API server to accept or reject requests depending authorization
IS-11	Stream Compatibility Management (work in progress)	Configure Sources, Flows and Senders
IS-12	Control Protocol (work in progress)	Setup and control Devices

In this document we will primarily looking cover the IS-04 and IS-05 portions of the specifications. These are the parts in most common use today. There is a call for more use of the IS-08 portion of the specification, and also the IS-07 specification as we move into IP expanding outside of the routing core.

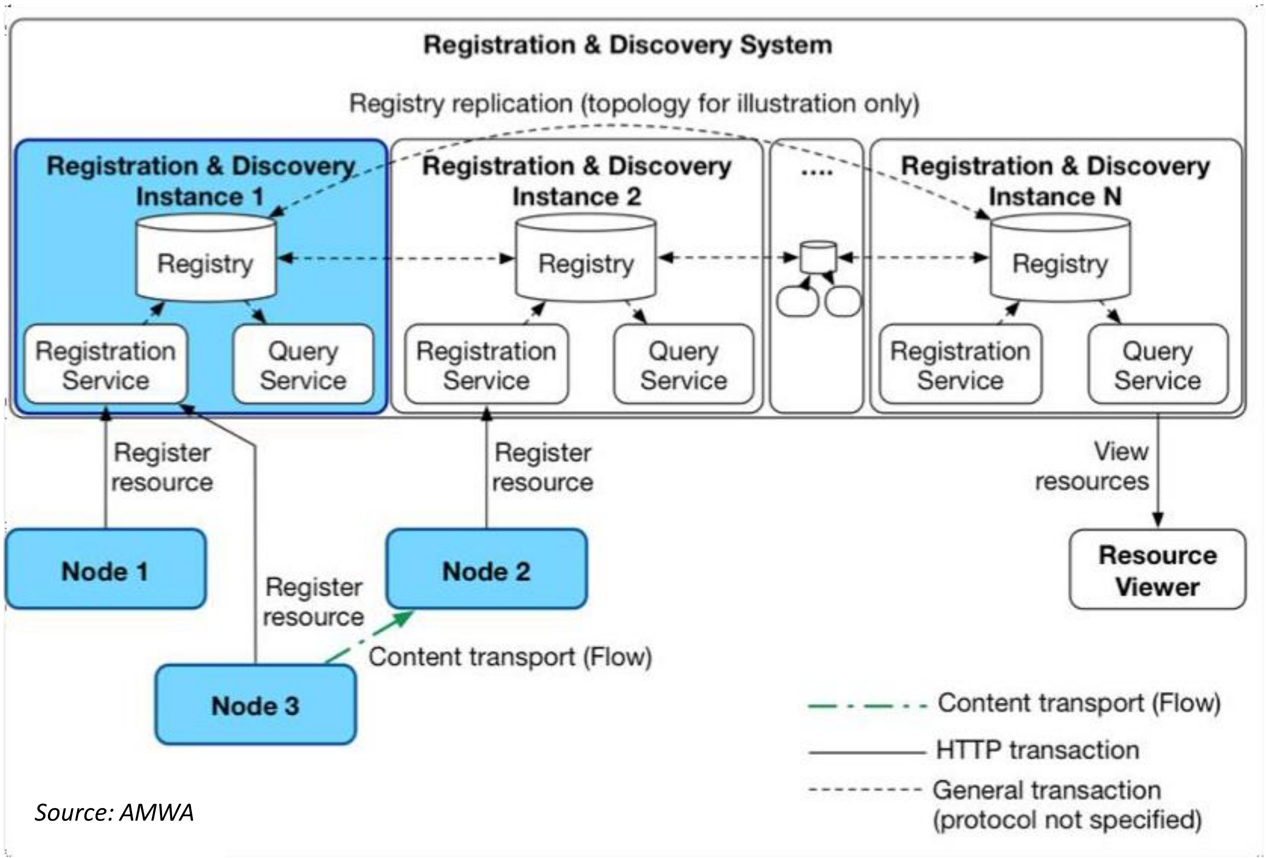
IS-04 Discovery and Registration

IS-04 is used for finding NMOS Nodes and their devices (Senders and Receivers) on the network, describing their capabilities, and advertising the locations of other (IS-05) APIs. The nodes will send their Session Description Protocol (SDP) file to the NMOS Registration & Discovery System (RDS). The SDP file has the all of the data needed to be used by the NMOS system (IS-05) for connecting a Sender to a Receiver.

The Registration and Discovery Specification describes two mechanisms for the discovery of Nodes and their resources: peer-to-peer and registered. These two mechanisms MAY co-exist on the same system, although most systems use a registry system. Registry and discovery takes place using RDS server, which can be a single server or be modular and distributed. An RDS is composed of one or more Registry & Discovery Instances (RDIs). Peer-to-peer can be used on a small system that is all on the same sub-net where nodes interact with each other directly without the need for an RDS.

Each RDI provides:

- A Registration Service
- A Query Service
- A Registry storage backend.



Source: AMWA

The Registration Service connects to the Nodes, the Registration service holds the SDP files sent by the nodes, the Query service makes the SDP files available to the control system (IS-05)

Registration & Discovery System (RDS)

There are 3 processes happen within the Registration & Discovery System (RDS):

1. The Registration API allows a Node to register its resources.
2. The Query API allows querying of registered resources.
3. The Node API is used to find resources on a Node, and is used for peer-to-peer

discovery The Specification covering the RDS includes:

JSON Schema definitions, with supporting JSON examples. The AMWA GitHub has more details for this and all aspects of NMOS.

NMOS and DNS-SD

End-points in the NMOS environment need to be able to find the Registration and Discovery Instances (RDIs) that make up the Registration and Discovery System (RDS). This can be accomplished by hard coding endpoints with the IP address of the RDS, but a simpler and more flexible method is to use service discovery via DNS (DNS-SD).

DNS (Domain Name System) provides a mechanism that allows end-points to resolve IP addresses from hostnames. In a web browser when you type GOOGLE.COM a DNS server looks up the domain Google and finds the IP address to connect you to Google using that IP address. DNS-SD provides the additional ability to discover hosts that provide specific network services from service type records.

The control network for NMOS and the RDS system are quite often on different networks or subnets in SMPTE 2110 systems, so we need a way to know the address of the RDS server on the other network/subnet to be able to register a device in the RDS.

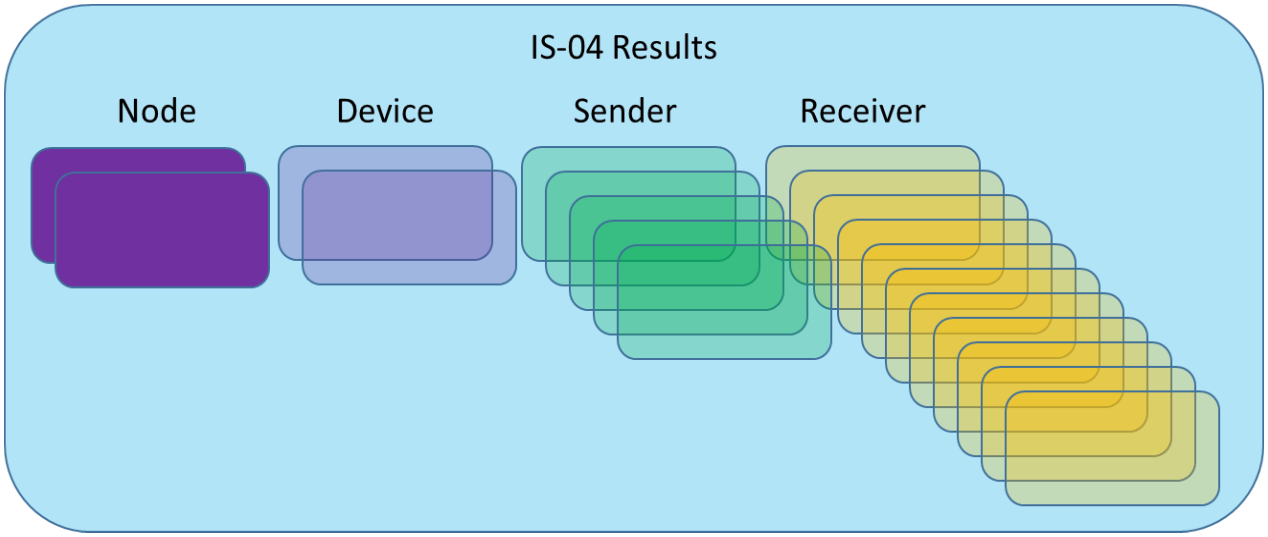
DNS-SD can be achieved using mDNS (multicast DNS) which is a very simple serverless peer-to-peer technology. This works well for small flat (layer 2) networks, but does not scale well to larger layer 3 networks that we generally see for ST2110 implementations.

An alternative to mDNS is to use a unicast, server-based name resolution service DNS-SD, with requests and queries being provided over routable unicast transport.

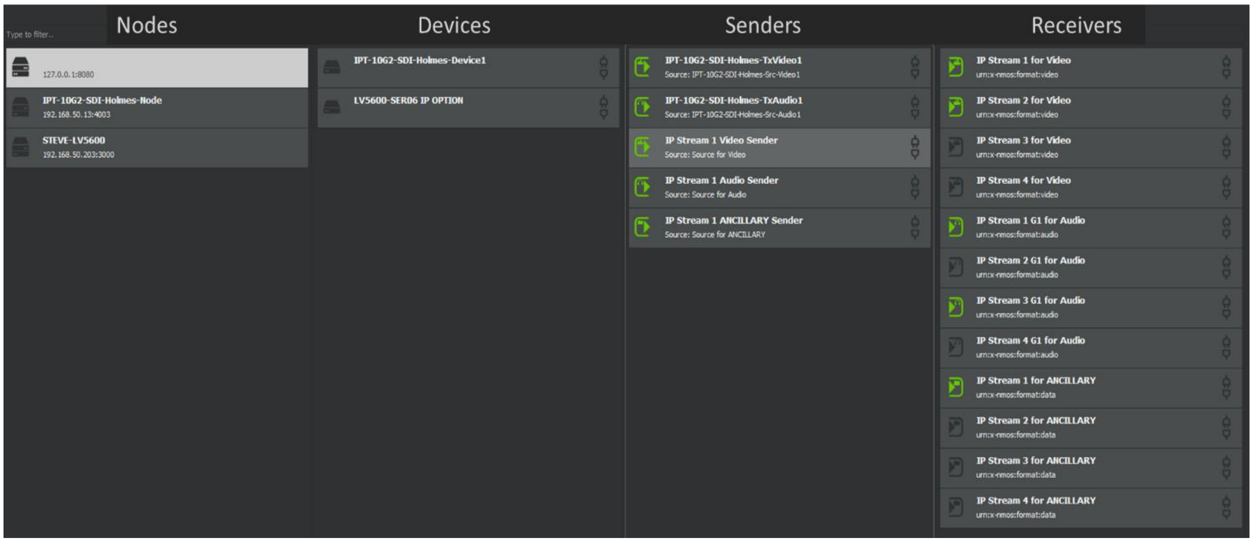
For most ST2110 systems, a unicast DNS-SD methodology provides the most appropriate choice. DNS- SD's unicast, routable characteristics provide scalability, and removes the inherent mDNS limitation of operation only within a subnet.

IS-04 Example

- RDS executes an mDNS or DNS announce
- Node executes a Query for the RDS
- Once the RDS is found, then
- Node does a Post (Target =/resource, type “node”, data)
- Post (Target =/resource, type “device”, data)
- Loop for all senders
    - Post (Target =/resource, type “sender”, data)
- Loop for all receivers
    - Post (Target =/resource, type “receivers”, data)



A view of the results from Riedel’s NMOS Explore



Type to filter...

Nodes

127.0.0.1:8080

IPT-10G2-SDI-Holmes-Node
192.168.50.13:4003

STEVE-LV5600
192.168.50.203:3000

Devices

IPT-10G2-SDI-Holmes-Device1

LV5600-SER06 IP OPTION

Senders

IPT-10G2-SDI-Holmes-TxVideo1
Source: IPT-10G2-SDI-Holmes-Src-Video1

IPT-10G2-SDI-Holmes-TxAudio1
Source: IPT-10G2-SDI-Holmes-Src-Audio1

IP Stream 1 Video Sender
Source: Source for Video

IP Stream 1 Audio Sender
Source: Source for Audio

IP Stream 1 ANCILLARY Sender
Source: Source for ANCILLARY

Receivers

IP Stream 1 for Video
urn:x-nmos:format:video

IP Stream 2 for Video
urn:x-nmos:format:video

IP Stream 3 for Video
urn:x-nmos:format:video

IP Stream 4 for Video
urn:x-nmos:format:video

IP Stream 1 G1 for Audio
urn:x-nmos:format:audio

IP Stream 2 G1 for Audio
urn:x-nmos:format:audio

IP Stream 3 G1 for Audio
urn:x-nmos:format:audio

IP Stream 4 G1 for Audio
urn:x-nmos:format:audio

IP Stream 1 for ANCILLARY
urn:x-nmos:format:data

IP Stream 2 for ANCILLARY
urn:x-nmos:format:data

IP Stream 3 for ANCILLARY
urn:x-nmos:format:data

IP Stream 4 for ANCILLARY
urn:x-nmos:format:data

IS-05 Device Connection Management

IS-05 is the control part of NMOS, IS-05 is an API that provides the means to create a connection between a Device in an NMOS compatible system and other Devices by the use of Senders and Receivers. Under control of the NMOS admin server/user interface, the server will send an SDP file to the receiver for the sender to connect to. In a conventional COTS IP system the receiver will issue a multicast join to the multicast network switch for the entity to be connected to the receiver. The IP switch will then forward the requested multicast IP flow to the receiver. In a system where a COTS switch is not involved the SDP file will be sent to the receiver to set up the receiver's local interface (IP address and Port) for the flow to be sent to the receiver. The user interface server API will configure the network element to send the flow to the appropriate receiver.

Single & Bulk Interfaces

The API provides a mechanism to modify settings for an individual Sender or Receiver. This is referred to as the 'single' interface.

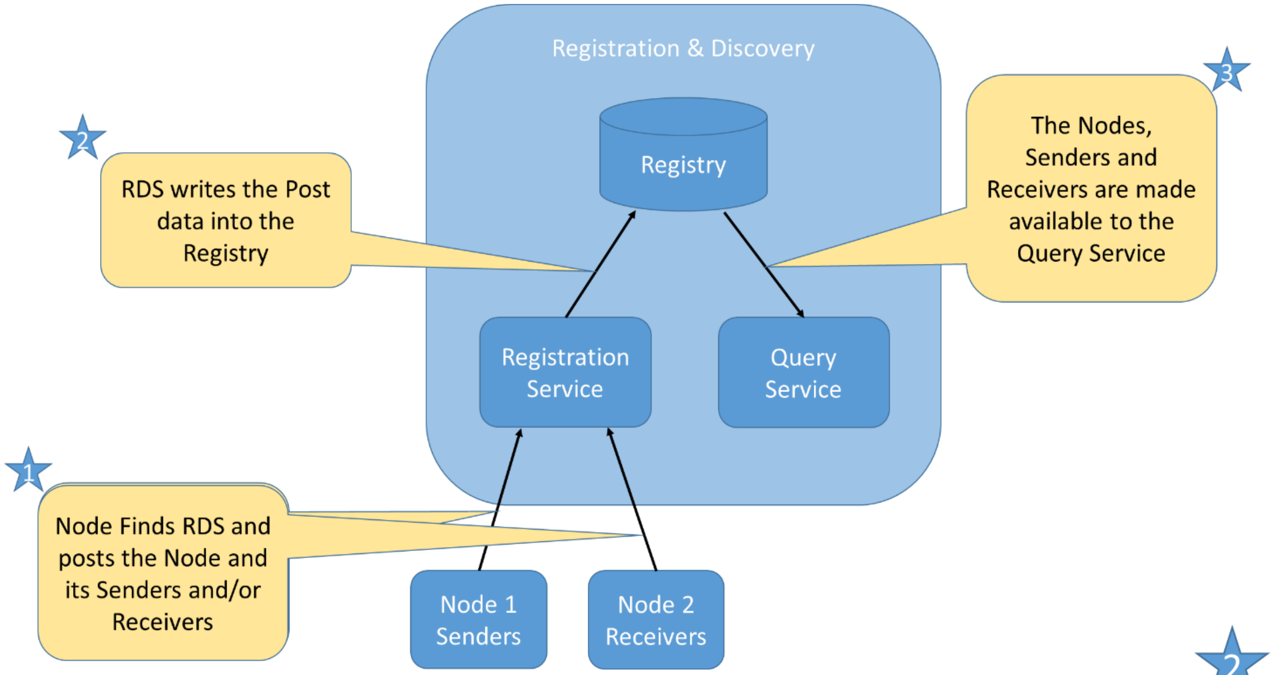
The API also provides a mechanism to modify settings for many Senders or Receivers at once which sit within the scope of the API implementation (typically contained by a Node or Device). This is referred to as the 'bulk' interface, and can be used to support 'salvo' operations in capable Devices.

HTTP APIs for telling senders where to send content

HTTP APIs for telling receivers where to get content

Trading SDP Files from Senders to Receivers

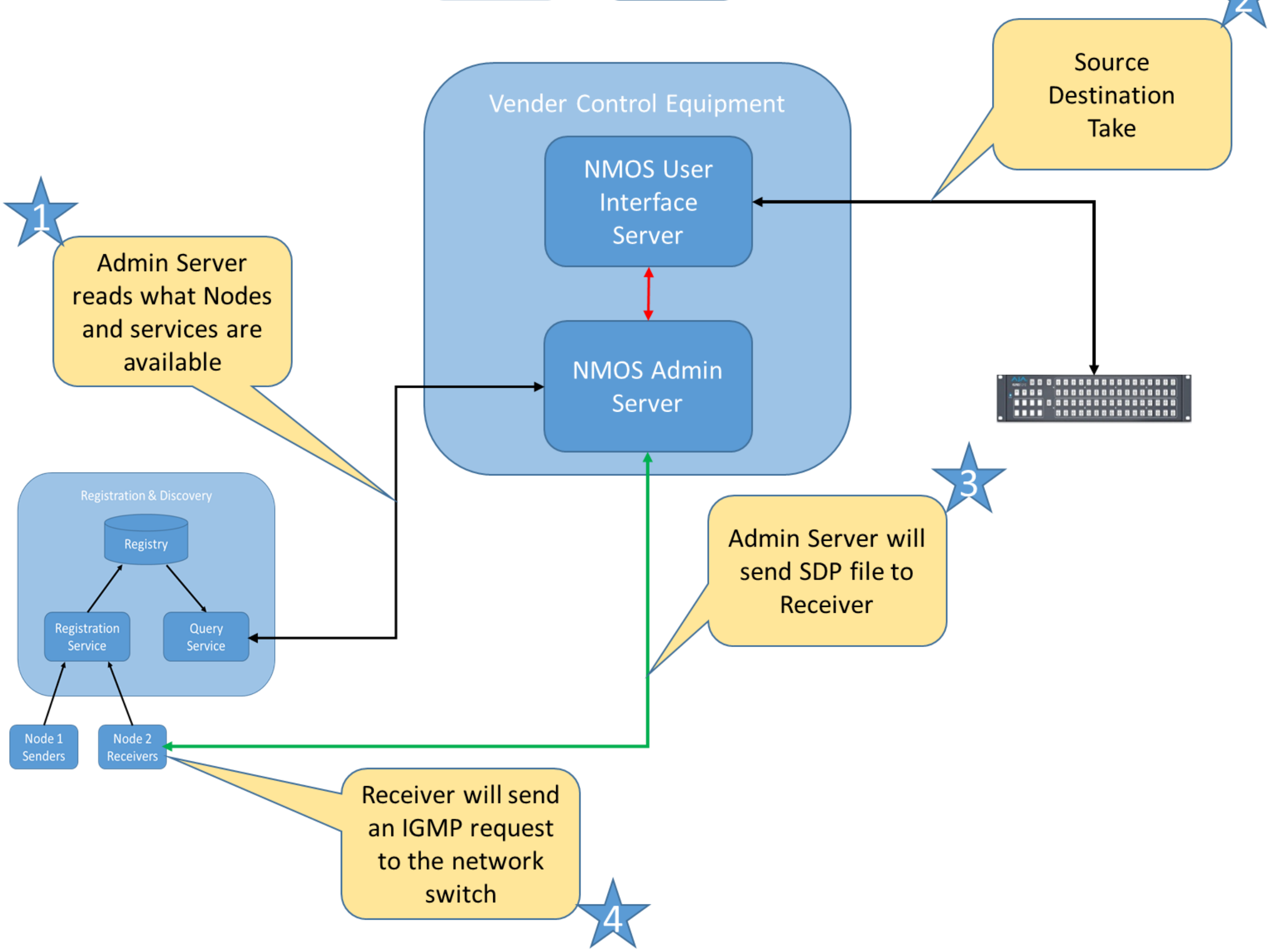
How to make a simple NMOS connection work



Node Finds RDS and posts the Node and its Senders and/or Receivers

RDS writes the Post data into the Registry

The Nodes, Senders and Receivers are made available to the Query Service



Admin Server reads what Nodes and services are available

Source Destination Take

Admin Server will send SDP file to Receiver

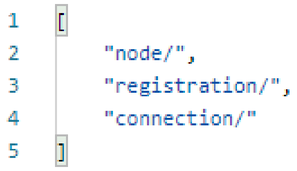
Receiver will send an IGMP request to the network switch

Looking at the NMOS Node server with your browser

You can browse your NMOS Node (source/destination) with a Browser or tools like Postman. On your Browser you may want to add the JSON Viewer Add-on. (The views below are done with Postman)

Type; http://(your IP address):(your port)/x-nmos/ The IP Address and port of your Node. (Device on the network I.E. 'Waveform Monitor'). http://192.168.50.203:3000/x-nmos/

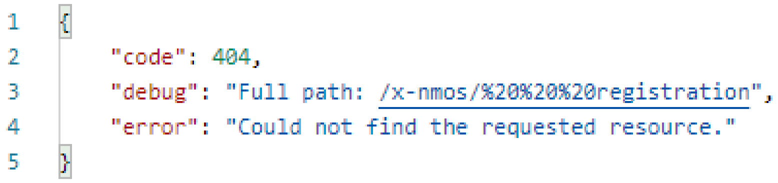
This is the return you will get;



```
[
    "node/",
    "registration/",
    "connection/"
]
```

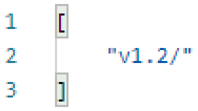
Note: Not all Links may have data in them: http://192.168.50.203:3000/x-nmos/registration/

You may get something like this on some links, this is normal. [ ] or an error code.



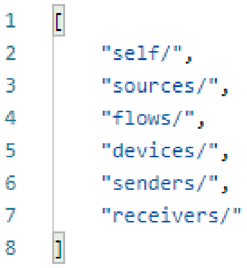
```
{
    "code": 404,
    "debug": "Full path: /x-nmos/%20%20%20registration",
    "error": "Could not find the requested resource."
}
```

If you select the NODE you will get a response, with the NMOS Version being used. http://192.168.50.236:8080/x-nmos/node/



```
[
    "v1.2/"
]
```

Add the Version to the URL bar, you will see; http://192.168.50.236:8080/x-nmos/node/v1.2

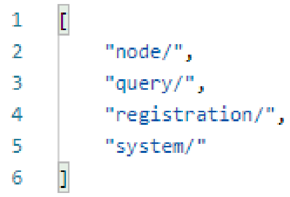


```
[
    "self/",
    "sources/",
    "flows/",
    "devices/",
    "senders/",
    "receivers/"
]
```

You can keep looking deeper and deeper in to the Nodes responses.

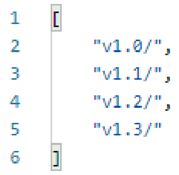
Looking at the NMOS Registry server with your browser

Type; http://(your IP address):(your port)/x-nmos/ Using the NMOS Registry IP address and port. http://192.168.50.236:8080/x-nmos/



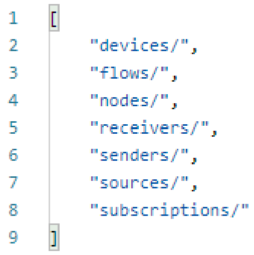
```
[
    "node/",
    "query/",
    "registration/",
    "system/"
]
```

The Query will have the data the NMOS Admin server will use. http://192.168.50.236:8080/x-nmos/query/



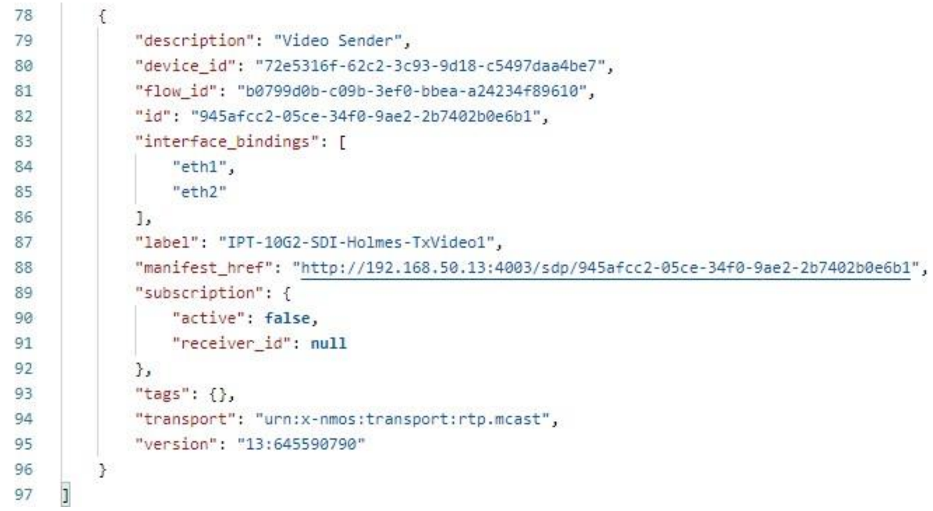
```
[
    "v1.0/",
    "v1.1/",
    "v1.2/",
    "v1.3/"
]
```

Now add Version, You will see all of the parts that make up the Query. (The Node was using Ver1.2) http://192.168.50.236:8080/x-nmos/query/v1.2/



```
[
    "devices/",
    "flows/",
    "nodes/",
    "receivers/",
    "senders/",
    "sources/",
    "subscriptions/"
]
```

Now look at Senders you will see the location for the SDP file for each sender, this may return several. http://192.168.50.236:8080/x-nmos/query/v1.2/senders



```
    {
        "description": "Video Sender",
        "device_id": "72e5316f-62c2-3c93-9d18-c5497daa4be7",
        "flow_id": "b0799d0b-c09b-3ef0-bbea-a24234f89610",
        "id": "945afcc2-05ce-34f0-9ae2-2b7402b0e6b1",
        "interface_bindings": [
            "eth1",
            "eth2"
        ],
        "label": "IPT-10G2-SDI-Holmes-TxVideo1",
        "manifest_href": "http://192.168.50.13:4003/sdp/945afcc2-05ce-34f0-9ae2-2b7402b0e6b1",
        "subscription": {
            "active": false,
            "receiver_id": null
        },
        "tags": {},
        "transport": "urn:x-nmos:transport:rtp.mcast",
        "version": "13:645590790"
    }
]
```

The following pages provide an overview of IS-06 through IS-12. For more detailed, up to date information use the links shown earlier in the document.

IS-06 Network Control

IS-06 is an API for Network bandwidth control in a managed network.

The API is “Northbound” of the Network Controller. The purpose of this API is to:

- Discover network topology
- Create and modify media flows in the network from a sender to one or more receiver devices
- Control how flows move on the network
- Assure bandwidth for these media flows
- Ensure network security by only allowing authorized senders, receivers and flows.

Conventions used in the Network Control API that are common with the specification of AMWA IS-04 and IS-05 are covered in the next section, APIs.

This specification has no dependency on AMWA IS-04 or IS-05. However the overall system is more fully automated with IS-04 and IS-05.

IS-07 Event & Tally

The purpose of the Event & Tally specification is to provide a mechanism by which to send and receive states and state changes issued by sources (switchers, sensors, actuators etc.). This is not a control API but is used for sending things like Camera Tally from a switcher.

IS-08 Audio Channel Mapping

This is used to remap Audi channels in a Sender or in a Receiver using an IS-08 API command. An NMOS compatible system can re-map audio channels. This can take place on a sending device, where the resulting re-mapped audio may be sent out over the network, or a receiving device, where audio received from the network may be re-mapped prior to consumption.

Inputs and Outputs both have a number of Channels available for use. An Input represents each of the audio tracks assigned to it as a Channel. The number of channels an Input or Output has can be restricted by the Device, and cannot be altered by the client.

The relationship between Input channels and Output channels is defined by the “Map”. Where a given Input is routable to a given Output, it MUST be possible to route any of the Input’s channels to any channel of the Output. An Output channel that does not have an Input channel routed to it is said to be “unrouted”

IS-09 System Parameters

The System API provides a single global configuration resource. This resource contains global configuration parameters which apply to the system in which Nodes are deployed. The System ID is exposed via the id attribute of the global resource. This is the ID of the system and is expected to be constant for the “life” of the system. This ID MUST be assigned uniquely in each facility which deploys a The System API is provided in order that Nodes can obtain configuration parameters at start-up. The

IS-10 NMOS Authorization

This specification adds a mechanism for the authorization of clients which need to access or modify resources held by an AMWA NMOS API. The specification requires the implementation of OAuth 2.0. OAuth 2.0 focuses on client developer simplicity while providing specific authorization flows for web applications, desktop applications as defined by various IETF RFCs, with additions made covering the topics of discovery, preferred OAuth grant types and the use of claims within JSON Web Tokens (JWTs).

IS-11 Stream Compatibility Management

This Specification is a work in progress and would be used to configure Sources, Flows and Senders using information from Receivers. With this specifications a Receiver would provide its capabilities, in the form of an EDID (an EDID is a list of a receivers capabilities). A Sender would read the EDID so that it knows what constraints the receiver has and can restrict media formats allowed for sending, and Inputs. Senders and Receivers can also present their EDID to upstream devices so they know the capabilities of each Sender and Receiver.

IS-12 Control Protocol

This Specification is a work in progress and aims to give a general description of the NMOS Device Control protocol. This protocol is compatible and adheres to the rules and requirements described in the NMOS Modeling architecture document and is the way in which NMOS Control models can be exposed and consumed in a standardized way.

About structure of the documents

"Vol.4 IP Conversion of Broadcasting" has four parts in total. This documents is “Overview edition”. Please take a look at the followings as well.

- IP Conversion of Broadcasting 1 Overview Background, Technologies that support IP transmission, Basic knowledge about communication
- IP Conversion of Broadcasting 2 Content About SMPTE ST 2110
- IP Conversion of Broadcasting 3 Synchronization About PTP
- IP Conversion of Broadcasting 4 Control About NMOS

Leader Electronics Corporation

Address	2-6-33 Tsunashima-Higashi, Kohoku-ku, Yokohama-shi, Kanagawa Prefecture 223-8505, Japan
Phone	+81-45-541-2123
Email	sales@leader.co.jp
URL	www.leader.co.jp/en
Published	June 27, 2022