WHITE PAPER

UNCOMPRESSED MEDIA OVER IP ON AMAZON WEB SERVICES (AWS)

The Transition from SDI to IP

NOTICES

CONTENTS

INTRODUCTION

As part of the ongoing transition to Internet Protocol (IP) networks for media workflows, the media and entertainment industry has developed new IP standards for transporting broadcast signals. This white paper offers guidance to help broadcasters and media organizations transition to IP, and demystifies common misconceptions when using the Society of Motion Picture and Television Engineers (SMPTE) 2110 standards and associated IP infrastructure. It provides details explaining why organizations transition from traditional Serial Digital Interface (SDI) to IP workflows using SMPTE 2110 standards and covers important topics related to workflow design, synchronization, and clocking, including relevant standards to support uncompressed contribution. It also provides details around Networked Media Open Specifications (NMOS) registration and discovery on an IP network. Finally, this paper gives readers an understanding of the best practices needed to achieve a smooth transition when using IP workflows and standards both for on-premises and cloud-based workflows.

REASONS TO MIGRATE FROM SDI TO SMPTE 2110

The broadcast industry has gone through several technology shifts in recent decades, from analog to digital, from SD to HD to UHD, and now from baseband to IP. IP technology is agile, enabling rapid deployment of new workflows and faster scaling using off-the-shelf information technology (IT) equipment, while bringing down costs through use of common networking infrastructure for both video and other IP traffic.

Previous technology shifts required large capital expenditures from broadcasters each time a new format or standard was proposed (for example, upgrading SDI routers from 1.5Gb to 3Gb for HD and then again to 12Gb for 4K), which limited the speed of technology adoption and created long capital investment cycles. In contrast, as IT technology followed the exponential growth of Moore's Law, network switches have eclipsed the capacity of SDI routers. It is now common to see 25GbE & 100GbE ports attached to 400 GbE switches. The maximum practical size of SDI routers was around 2,000 HD uncompressed video I/Os with an entire rack of equipment, but now 400 GbE switches can handle 20,000 HD uncompressed I/Os in just two rack units.

Using Commercial Off-the-Shelf (COTS) IT equipment along with software services for IP workflows allows for a more scalable media workflow. When requirements change, the architecture, IT infrastructure, and backbone network can be upgraded and scaled to add additional capacity, instead of replacing the entire workflow. With this transition to IP, broadcast IT departments no longer need separate cables, routers, and terminal equipment for video and the rest of their corporate network traffic. Most operators will continue to separate traffic by priority; however, newer switches have the intelligence to prioritize real-time media streams.

INDUSTRY TRENDS AND THE AFFECT OF IP ON MEDIA WORKFLOWS

Beyond replacing SDI with IP, which can be accomplished using SMPTE ST 2022-6, IP-based transport provides the flexibility to invent and use new applications based on, and leveraged from, IT protocols and infrastructure. IP also allows for the addition of new technology without the need to update hardware or change existing workflows.

The SMPTE 2110 standards for uncompressed media over IP represent a major industry shift. The transition is similar to the move by broadcasters from physical tape to virtual files for content storage, which opened up new workflows, functionality, and increased efficiency. IP and SMPTE 2110 bring multiple benefits and increase flexibility by using standards that are video-format agnostic.

OVERVIEW OF SMPTE 2110

WHAT IS SMPTE 2110?

SMPTE 2110 is a suite of industry standards designed to enable carriage of professional media over managed IP Networks. The diagram in Figure 1 depicts all of the relevant standards that form part of the SMPTE ST 2110 standard.

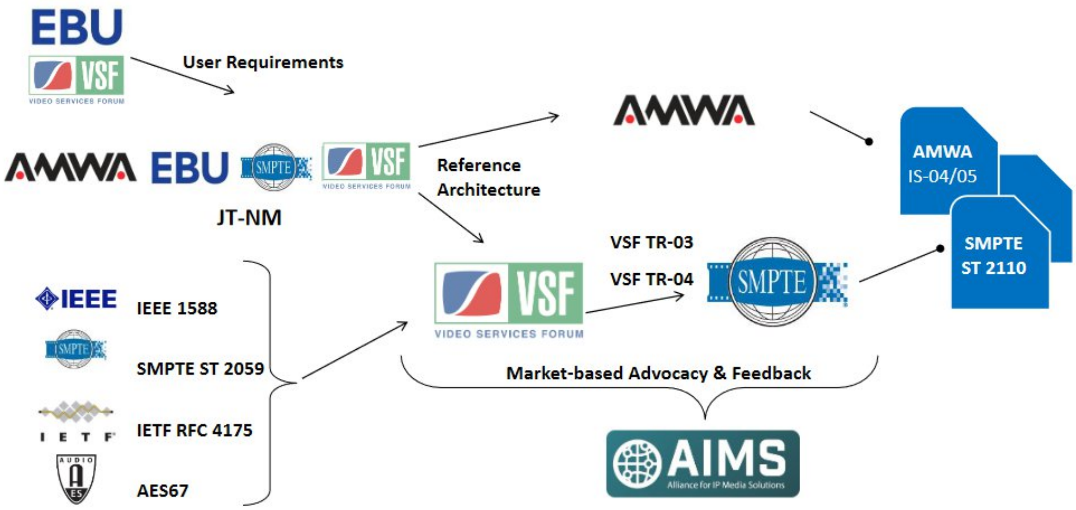

Figure 1: Development of SMPTE 2110 (courtesy of AIMS)

SMPTE 2110 ADVANTAGES AND EFFICIENCIES

SMPTE 2110 was designed to separate carriage of audio, video, and data. This offers the flexibility and gains in efficiency by carrying only the media content necessary to each stage of the workflow, which for production can enable highly efficient media workflows.

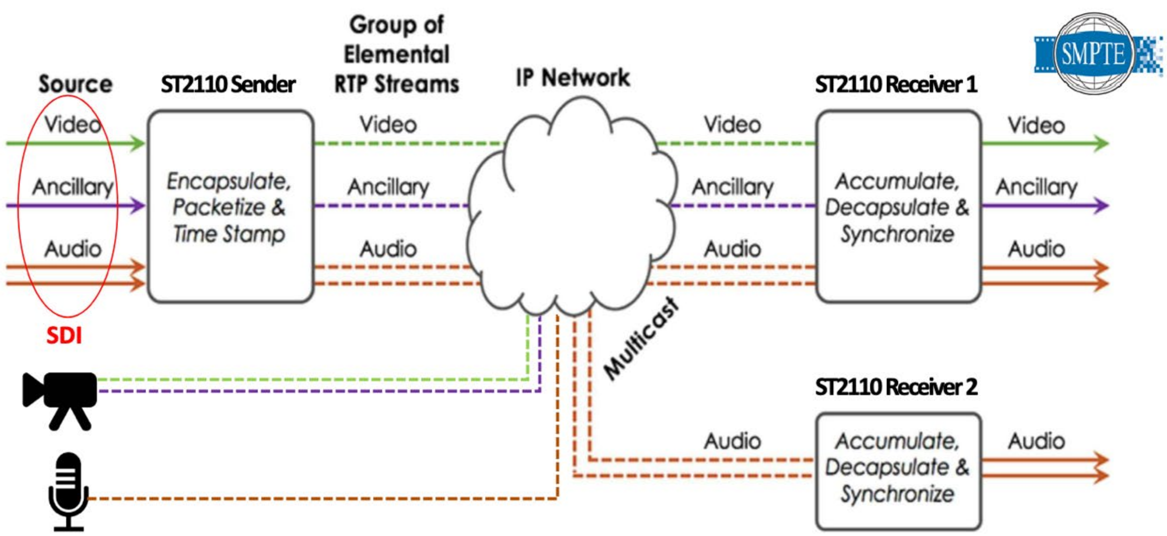

Figure 2: Composition of SMPTE 2110 (courtesy of AIMS)

STANDARDS

2110-10: System Timing and Definitions

2110-20: Uncompressed Active Video

2110-21: Traffic Shaping and Delivery Timing

2110-22: Constant Bit-rate Compressed Video

2110-30: PCM Digital Audio

2110-31: AES3 Transparent Transport

2110-40: Transport of SMPTE Ancillary Data

SMPTE ST 2110-10: SYSTEM TIMING AND DEFINITIONS

This standard defines the requirements for all essence types concerning system timing. RTP (Real-time Transport Protocol) is used for format-agnostic transport of real-time media. PTP (Precision Time Protocol) provides nanosecond granularity for system timing. SDP (Session Description Protocol) is used to describe connection, format, and timing information within the system.

SMPTE ST 2110-20: UNCOMPRESSED ACTIVE VIDEO

This standard defines the format for uncompressed active video. Bandwidth is saved by removing the audio (HANC) and vertical ancillary (VANC) data from transport. SDP is used to describe the video format, connection data, and timing information required to reproduce the stream. ST 2110-20 is format agnostic, allowing for a wide range of video formats, along with multiple color bit depths, colorimetry schemes, and frame rates.

SMPTE ST 2110-21: VIDEO TRAFFIC SHAPING AND DELIVERY TIMING

SMPTE ST 2110-21 defines the egress packet spacing from the sender of the SMPTE ST 2110-20 and SMPTE ST 2110-22 essence types. Network components have varying levels of buffering capabilities. Due to buffer unknowns, three models are defined for traffic shaping:

- Narrow Linear (NL): Defines packet spacing with equal timing between packets.
- Narrow (N): Defines packet spacing very similar to SDI timing where no packets are sent during the VBI or VANC.
- Wide (W): Defines packet spacing with varying timing between packets.

SMPTE ST 2110-22: CONSTANT BIT-RATE COMPRESSED VIDEO

SMPTE ST 2110-22 defines a standard for the transport of compressed video data over IP networks. The standard defines the Real-Time Transport Protocol payload, as well as synchronization of compressed video via PTP.

SMPTE ST 2110-30: PCM DIGITAL AUDIO

Based on AES67, SMPTE ST 2110-30 defines the format and packet spacing for PCM audio streams. Audio streams can use 16 or 24 bit depth, 48 kHZ or 96 kHZ sample rate, and packet timing of 1 ms or 125 µs. Three levels are defined by this standard:

Level A: 1-8 Channels, 1 ms packet timing, and 48 kHz or 96 kHz sample rate.

Level B: 1-8 Channels, 125 µs packet timing, and 48 kHz or 96 kHz sample rate.

Level C: 1-64 Channels, 125 µs packet timing, and 48 kHz or 96 kHz sample rate.

SMPTE ST 2110-31: AES3 TRANSPARENT TRANSPORT

SMPTE ST 2110-31 specifies the real-time transport of AES3 data. Linear PCM data encapsulated in AES3 format can be carried in SMPTE ST 2110-31 transport as well as any non-PCM encoded audio or data like Dolby Digital (AC3) or Dolby E defined by SMPTE ST 337/338. This standard is built upon Ravenna's AM824, IEC 61883-6 definition for audio payload.

SMPTE ST 2110-40: TRANSPORT OF SMPTE ANCILLARY DATA

SMPTE ST 2110-40 defines the real-time transport of SMPTE ST 291-1 ancillary data via the Real-Time Transport Protocol as defined in RFC 8331. RFC 8331 describes the RTP packet payload for items such as line number and horizontal offset in addition to typical SMPTE ST 291-1 packet providing DID/SDID and Data Count.

REQUIREMENTS FOR MIGRATING TO SMPTE 2110

Migrating to SMPTE 2110 doesn't need to involve large changes all at the same time. A hybrid approach where SDI and IP co-exist for some period of time is more pragmatic. Adding IP workflows to existing solutions provides support for higher resolution video or new technologies such as High Dynamic Range and High Frame Rate while ensuring a scalable way to migrate to ST 2110.

SIMPLE MIGRATION TO SMPTE 2110

As equipment vendors and service providers add IP interfaces to on-premises products, a simple first step, when purchasing new equipment, is to make sure that the hardware supports SMPTE 2110 (via a 10GbE or greater capable Network Interface Card).

An easy way to migrate to SMPTE 2110 is to bridge from SDI to IP using encapsulators and de-encapsulators. Gateways that convert to or from SDI perform a number of functions. The SDI signal is converted into separate essence flows of audio, video, and ancillary data carried via SMPTE 2110 for equipment and elements of the workflow that support SMPTE 2110. These gateways also provide aggregation of essence streams into higher capacity bandwidth networks (e.g. 10 GbE, 25 GbE, or higher). Legacy equipment can also be used until a suitable time is available to migrate to SMPTE 2110 when newer models or equipment become available.

A SIMPLE HYBRID MIGRATION EXAMPLE OF SDI TO IP

A simple example of a common migration for a 4K/UHD service is depicted in Figure 3. This diagram has a UHD signal transported via 4 x 3G SDI signals from production encapsulated to SMPTE 2110 and then encoded to a delivery format using SMPTE 2110 inputs and outputs to a downstream IP delivery network. SMPTE 2110 carries the audio, video, and data as separate essences each with a unique IP address and/or port. Each individual essence is kept in sync using PTP timing.

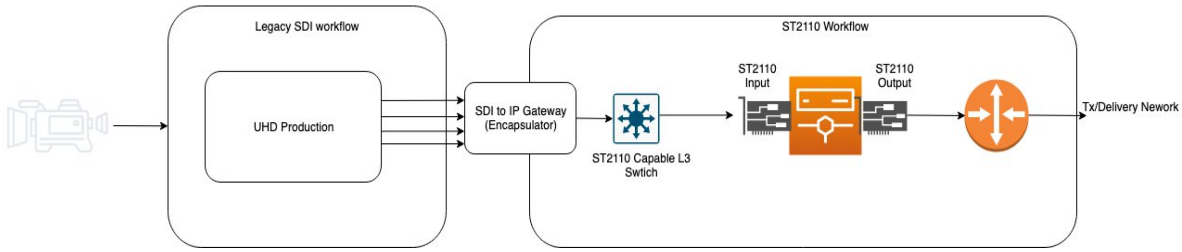

Figure 3: UHD hybrid example using IP/SMPTE 2110 to reduce SDI cabling using an AWS Elemental Live encoder and an SDI-to-SMPTE 2110 IP Gateway

A SMPTE 2110, layer 3 capable switch is used to route the SMPTE 2110 packets to an AWS Elemental Live encoder. AWS Elemental Live supports both SMPTE 2110 inputs and SMPTE 2110 outputs that can be routed to a downstream delivery network.

COTS hardware used for SMPTE 2110 workflows often require a 40GbE-100GbE backbone (or the capability to support this) with at least 10GbE connectivity to endpoints with PTP enabled.

For additional details on how to setup and use AWS Elemental Live as part of a SMPTE 2110 workflow such as the one shown in figure 3, refer to the following blog post:
https://aws.amazon.com/blogs/media/part-2-working-with-2110-on-the-aws-elemental-live-appliance/

SMPTE 2110 DESIGN CONSIDERATIONS

Network Architecture and Traffic Shaping

IP network design requires careful consideration and proper shaping of traffic to avoid overloading the network switch. Spine-leaf network architectures help avoid overloads. The SMPTE ST 2110-21 standard provides details on traffic shaping and delivery timing for video on managed networks.

Redundancy and Resilience

Two of the most important design considerations for workflows of any size are redundancy and resiliency. A key standard for IT network redundancy is SMPTE ST 2022-7, which provides guidance for seamless protection switching between RTP essence streams. ST 2022-7 defines a dual stream approach where two parallel sources are used to ensure any burst packet errors in the incoming streams are corrected, as each packet is sent multiple times in parallel over separate physical links.

Monitoring and Security

SDI signal monitoring and security are both well established, with tools and equipment available from many manufacturers to enable eyes-on-glass and remote monitoring. IP monitoring tools are required for SMPTE 2110 so engineers can select stream essences to probe and capture raw packets for additional analysis. These tools vary in cost and capability but are critical to make sure that any errors and trends are monitored to actively observe network traffic, errors, quality of service (QoS), and quality of experience (QoE).

Security of SMPTE 2110 is an important consideration to make sure that streams are protected and only accessible to devices that should have access. This is different from SDI, where point-to-point physical connectivity must be secure as there are no other entry points to access the data.

Device Registration and Discovery

Unlike SDI devices where physical connections are point-to-point, SMPTE 2110 IP workflows allow different devices to share the same physical connections. Media identity is not directly tied to a physical connection, but is associated with corresponding sender or receiver stream details, allowing the media to be transported over IP to various destinations simultaneously.

NMOS

At a high level, NMOS is a group of specifications, created by the Advanced Media Workflow Association (AMWA), that allows for the orchestration of media flows on an IP network. NMOS provides a simple, vendor-independent way to connect SMPTE 2110 senders and receivers, as this process is not defined in the SMPTE 2110 specification.

NMOS is used to discover, connect, and configure media devices to enable the transport of video, audio. and ancillary media on an IP network.

AWS Elemental Live appliances support the AMWA IS-04 and AMWA IS-05 specifications. AMWA IS-04 allows for device discovery and registration, while AMWA IS-05 focuses on device connection management.

For additional information on NMOS and how to configure AWS Elemental Live, where the appliance itself is referred to as a node, see the following blog post: https://aws.amazon.com/blogs/media/part-3-simplifying-with-nmos/

In summary, IP can be added to augment pre-existing SDI workflows and enable the use of new technologies with off-the-shelf hardware. Moving to IP requires an understanding of managed IP networks and PTP timing. There are important considerations outlined in this white paper that should be adhered to in order to ensure reliability of the network architecture, security implementation, as well as any management, orchestration, and monitoring services used.

SESSION DESCRIPTION PROTOCOL

Session Description Protocol (SDP) is a proposed standard published by IETF (Internet Engineering Task Force) in RFC 4566. SMPTE 2110 standards adopted the SDP protocol to describe connection, format, and timing information within the system. SMPTE 2110 capable devices transmit and receive media flows consisting of video, audio, and ancillary data in elementary essence streams. SDP describes essential information about the media streams, such as video frame rate, resolution, and colorimetry. In a SMPTE 2110 system, SDP information is made available on each device that contains logical inputs (receivers) and outputs (senders). The SMPTE 2110 control system defines the mechanisms to communicate and transport SDP information from one system to another. For example, NMOS IS-04 and IS-05 are used for discovery, registration, and connection management, and devices provide REST APIs to facilitate information exchange.

To establish a SMPTE 2110 stream from a sender to a receiver, the receiver needs to get video, audio, and ancillary data SDP information from the sender. The receiver validates the SDP information and connects to the sender’s media flow. SDP is the protocol that allows senders to communicate enough information for receivers to properly handle data and connect to its flow.

An SDP session description consists of a number of lines of simple text in the format: <type>=<value>, and is divided into two parts. The first part describes general information for the session while the second part describes the media information including video, audio, and/or ancillary data. The session description starts with a “v=” and ends at the first media-level section “m=”. It includes the version of the SDP, information to identify the session, and time of the session. Below is a table with session description and time description attributes.

Field	**Description**
Version	v=0
Origin	o=<username> <sess-id> <sess-version> <nettype> <addrtype> <unicast-address>
Session Name	s=<session name>
Timing	t=<start-time> <stop-time>
Attributes	a=<attribute> | a=<attribute>:<value>
Media Descriptions	m= m=<media> <port> <proto> <fmt> ...
Connection Data	c=<nettype> <addrtype> <connection-address>

The media description includes an overview of the media in the session and the information needed to receive the media. Video, audio, and ancillary data are described in separate SDP descriptions. To understand an SDP description, it is helpful to review this example of a video SDP file.

Field	**Description**
Version	v=0
Origin	o=- 0 1 IN IP4 192.168.10.125 *(-indicates that the sender does not support user ID)*
Session Name	s=AWS Elemental SMPTE ST 2110 Output: [LiveEvent: 8] [OutputGroup: smpte_2110] [EssenceType_ID: 2110-20_video_435]
Timing	t=0 0 *(start and stop times of 0 means the video is a live signal that does not have a schedule start or stop time)*
Attribute – group identification	a=group:DUP primary secondary
Media Descriptions	m=video 50020 RTP/AVP 96 *video RTP flow on port 50020, using dynamic format 96*
Connection Data	c=IN IP4 10.20.0.1/64

	This information can be specified at the media or session level. For redundant flows such as SMPTE 2022-7, it must be specified at the media level *10.20.0.1 is the destination IP address* *64 is the TTL*
Attribute – source filter	a=source-filter: incl IN IP4 239.20.0.1 10.11.12.50 *a = is an attribute of the media stream. There can be multiple attributes.* *Receiver applications are expected to use the SDP source-filter information to identify traffic from legitimate senders and discard traffic from illegitimate senders* *In this case, the first IP address is the destination and the second IP address is the source*
Attribute – dynamic payload	a=rtpmap:96 raw/90000 *rtpmap - maps an RTP payload type number to a media encoding name that identifies the payload format. raw indicates the video is not compressed. If the video was compressed using JPEG XS, "raw" would be replaced with "jxsv". 90000 is the clock rate*
Attribute – media format	a=fmtp:96 sampling=YCbCr-4:2:2; depth=10; width=1920; height=1080; exactframerate=60000/1001; colorimetry=BT709; TP=2110TPN; TCS=SDR; PM=2110GPM; SSN=ST2110-20:2017; PAR=1:1; *fmtp:96 indicates the start of format specific parameters and 96 is the RTP dynamic payload type* *sampling=YCbCr-4:2:2 The color space and sampling. In this case YUV 4:2:2. Many color and sample rates are supported* *depth=10; Pixel color bit depth* *width=1920; The width of the video frame in pixels* *height=1080; The height of the video frame in pixels* *exactframerate=60000/1001; The frame rate of the video. Integer values are acceptable* *colorimetry=BT709* *TP=2110TPN; Defines the SMPTE 2110-22 Sender Profile* *2110TPN = Narrow Sender Profile* *2110TPNL = Narrow Linear Profile* ***2110TPW = Wide Sender Profile*** *TCS=SDR; Transfer Characteristic System* *PM=2110GPM; Packing Mode* *SSN=ST2110-20:2017; SMPTE Standard Number. This is required and must be ST2110-20:2017* *PAR=1:1; Explicitly defining the Pixel Aspect Ratio is not required when it is 1:1*

Attribute – media clock	a=mediaclk:direct=0 *Defines the PTP reference clock mapping and always has the value direct=0*
Attribute – PTP reference clock	a=ts-refclk:localmac=00-0B-72-A7-C9-A1 *The Grand Leader clock identifier*
Attribute – group section	a=mid:primary ***Only present if using SMPTE 2022-7***

Audio and ancillary data SDP descriptions are similar to video SDPs. The session description format is the same, but the media description begins at "m=". Below is an example audio SDP and ancillary data SDP description, where the common fields similar to the previous video SDP have been removed.

Field	**Description**
Media Descriptions	m=audio 6001 RTP/AVP 97 *This is an example of an audio RTP flow on port 6001, using dynamic format 97*
Attribute – dynamic payload	a=rtpmap:97 L24/48000/2 *This is an example that defines dynamic format 97 as a 24-bit linear PCM stereo audio sampled at 48 kHz*
Attribute – audio payload channel layout	a=fmtp:97 channel-order=SMPTE2110.(ST) *Defines the audio payload channel layout* *M = Mono* *ST = Stereo* *SRGP = 4 channel audio* *5.1 = Typical 5.1 audio setup* *7.1 = Typical 7.1 audio setup*
Attribute – packet time	a=ptime:1 *Specifies that each RTP packet contains 1 millisecond of audio data*

Field	**Description**
Media Descriptions	m=video 20000 RTP/AVP 100 *This is an example of an ancillary RTP flow on port 20000, using dynamic format 100*
Attribute – dynamic payload	a=rtpmap:100 smpte291/90000 *This is an example that defines dynamic format 100 as SMPTE ST 291 ancillary data packet with 90 kHz clock rate*

SYNCHRONIZATION, CLOCKING, AND PTP

SMPTE 2110 enables the transition from a traditional SDI-based production workflow to an IP-based workflow, with the flexibility to deliver video, audio, and ancillary data in separate flows. Because of the nature of an IP-based, bi-directional, asymmetric packet switching network, the inherent delay, jitter, and risk of dropping packets requires a mechanism for stream synchronization, especially in a live production environment. Maintaining alignment is critical for live video and audio “lip sync”, and enables frame accurate switching. Precision Time Protocol (PTP), also referred to as IEEE 1588-2008 (PTP v2), is the standard timing protocol for SMPTE 2110.

PTP is a packetized protocol that runs on an IP network to synchronize time across its domain, to a single grand leader clock, which is usually synchronized to GPS, GLONASS, or both. Network devices that synchronize timing from a PTP domain are PTP followers.

PTP defines a leader and follower architecture model for time distribution across IP networks, and different types of clocks are used throughout the distribution system. An ordinary clock is a device that acts as either the source (leader or grand leader) or the destination (follower) for time synchronization. Both leader and follower clocks connect to the network via a single network segment. Devices such as routers and switches that interconnect the IP network can operate either as boundary clocks or transparent clocks. A boundary clock has multiple network segment connections, and it can synchronize one network segment to another. PTP frames are not forwarded through a boundary clock; instead, each port connecting to a network segment has its own state as a leader or follower. On the other hand, a transparent clock is a device that has multiple network segments, but PTP frames are forwarded through the device with modified time to compensate for the time the frames spend on the device.

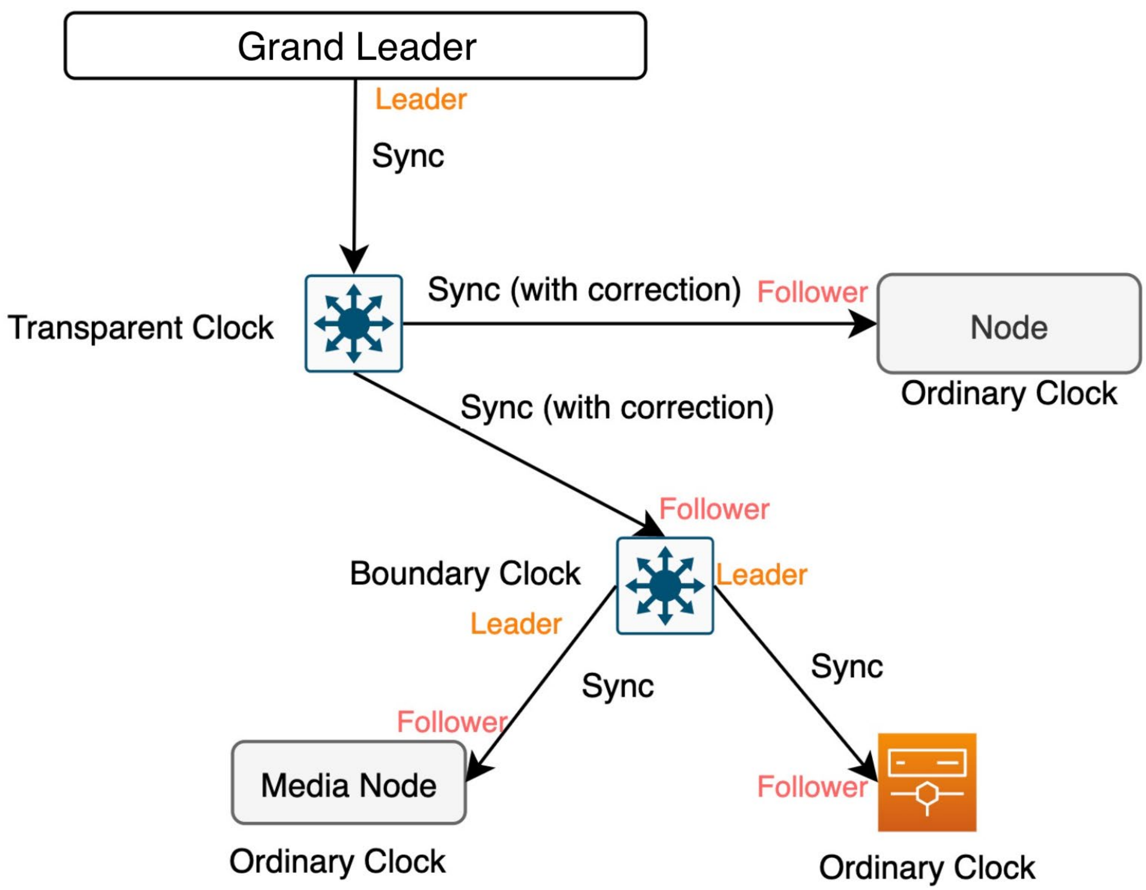

Figure 4: Diagram of PTP clocks communicating

Within a PTP domain, the protocol uses a number of message types to manage the PTP domain system and establish time synchronization. Sync and Follow_Up messages are transmitted by grand leader and boundary clocks, and used by followers to drive the time. Delay_Req and Delay_Resp are messages initiated by the follower to determine the reverse path propagation delay and the response is sent by ordinary or boundary clock with the “time of receipt” timestamp information. These four messages carry timestamp information that can be used to synchronize time across a PTP domain. Announce messages are used by the Best Main Clock Algorithm (BMCA, also referred to as Best Master Clock Algorithm) to select the grand leader within a clock hierarchy.

PTP uses BMCA to improve resiliency. BMCA allows automatic election of a grand leader if the previous grand leader fails for any reason, such as loss of GPS lock or a disconnect from the network. The following attributes are used in the respective order in the BMCA selection algorithm.

1. Priority 1: a configurable clock priority (lowest value <= 128 wins)
2. Class: a clock’s overall quality associated with its own priority
3. Accuracy: a clock’s accuracy
4. Variance: a clock’s variability
5. Priority 2: a configurable second order clock priority (lowest value <= 128 wins)
6. Unique Identifier: clock source ID, usually the MAC address of the node

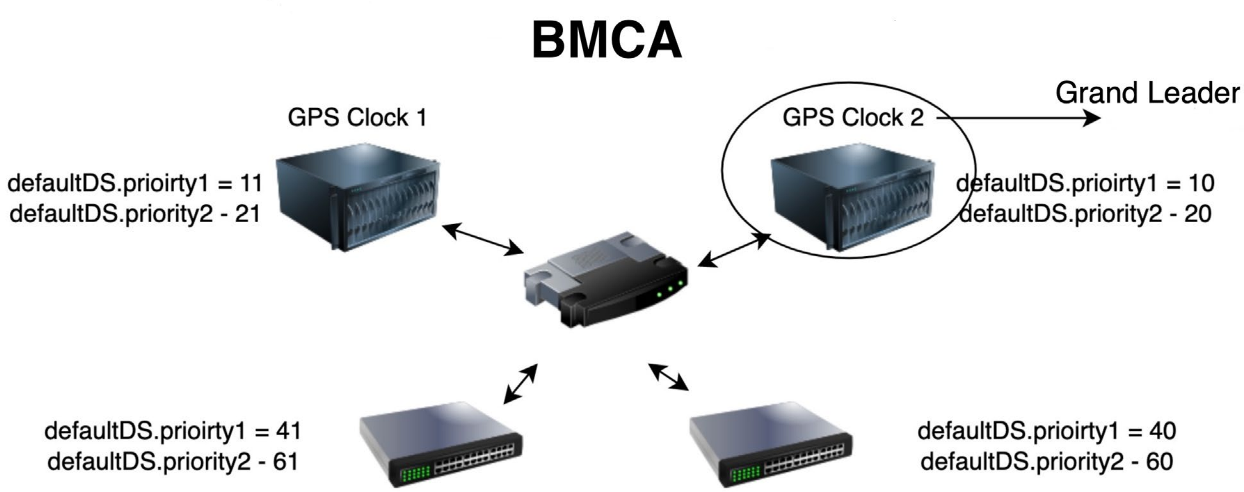

Figure 6: Diagram showing BMCA

A clock in a PTP domain starts to send Announce messages with clock properties and listens to Announce messages from connected network segments. When a clock receives messages from a better clock based on the selection algorithm, it will stop transmitting Announce messages and become a follower. A clock that receives no better clock Announce messages for a defined interval will become leader and continue to send Announce messages. Eventually there will be a single clock elected to be grand leader in the PTP domain. The algorithm runs all the time and a follower will start to transmit Announce messages if it fails to receive Announce messages for a defined interval.

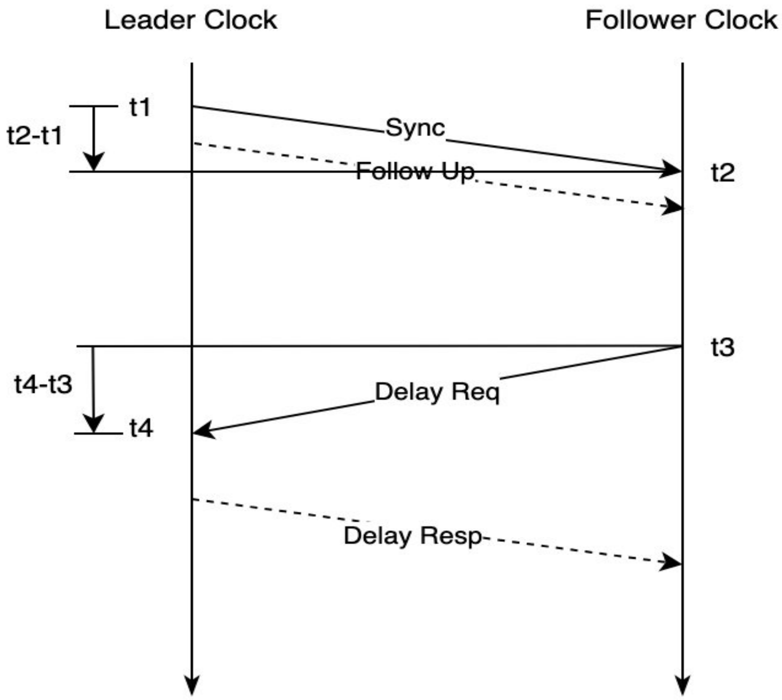

Figure 6: Diagram showing the Leader and Follower synchronization message exchange

PTP is a mandatory requirement for all media IP deployments based on SMPTE 2110. Currently, PTP is only used for SMPTE 2110 outputs on AWS Elemental Live appliances, and is not required for SMPTE 2110 inputs. However, sources without PTP sync can cause synchronization issues when ingested by Elemental Live. The best practice is to have all SMPTE 2110 devices synchronized to the same PTP source.

As mentioned previously, PTP provides the timing for packet spacing on Elemental Live's SMPTE 2110 output. There are 3 profiles in the 2110-21 spec:

- Type NL - Narrow Linear: Stream where all the packets for a video signal are evenly spaced across the duration of each video frame.

- Type N - Narrow: Similar to Type NL, except the sender doesn't send packets during the time that would correspond to the VBI (Vertical Blanking Interval) or VANC (Vertical Ancillary data space) of the corresponding SDI video signal. **This is the format used by Elemental Live.** You will see TP=2110TPN in the output video SDP file.

- Type W - Wide: Greater burstiness. For Type W flows, senders can have at least quadruple the amount of burstiness as a Type N or a Type NL.

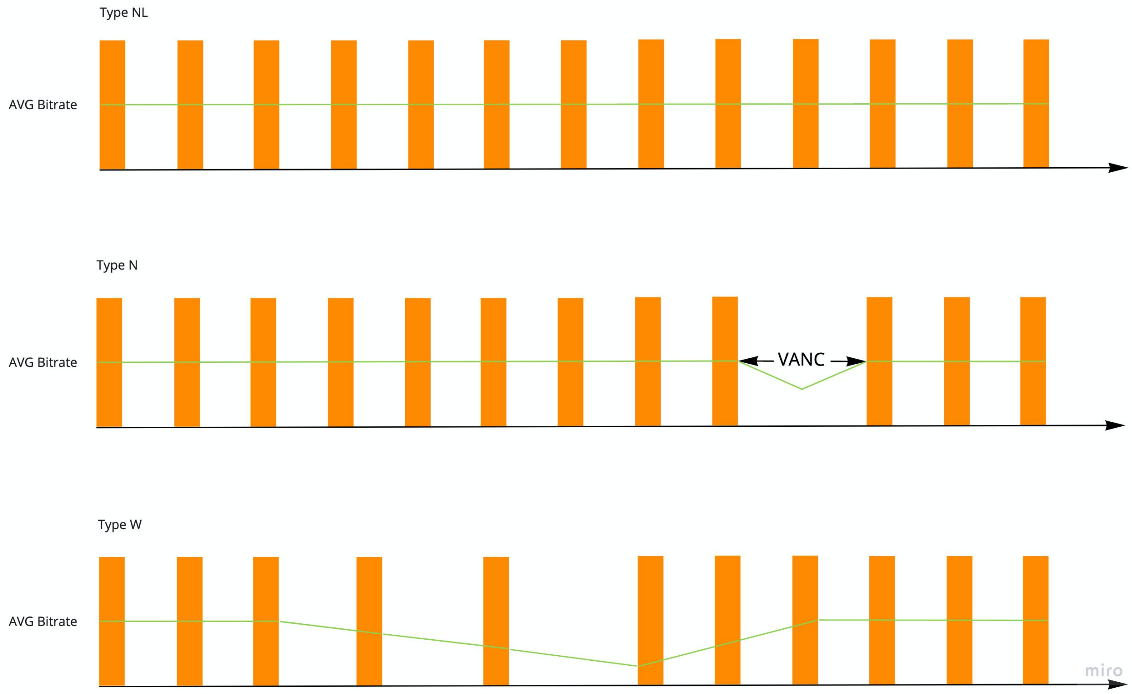

Figure 7: Diagram showing the three types of PTP packet spacing [i]

When deploying PTP within a media distribution network, especially one tuned for live broadcast production workflows, it is recommended to use hardware-based, time-stamped grand leader devices that are locked to GPS. A dual grand leader implementation is used to improve PTP reliability and resiliency. With a dedicated connection between dual grand leaders, BMCA will auto-select the grand leader and maintain high availability. The network switching layer should support PTP or be PTP aware. The core switches are configured as boundary clocks to improve network scalability and traffic isolation. The distribution layer switches can help isolate and prevent a media node from accidentally becoming a leader clock by port level configurations.

In summary, PTP is required in any SMPTE 2110 deployment and plays an essential role in keeping timing and synchronization between essences. In a SMPTE 2110 enabled network, use PTP aware switches and boundary clocks to scale the network deployment. BMCA is used to support redundancy and improve PTP domain resiliency in case of a grand leader clock failure.

DISCOVERY, REGISTRATION, AND CONNECTION MANAGEMENT WITH NMOS IS-04 AND IS-05

NMOS defines how networked media devices interoperate at the control and application layer over IP infrastructure. NMOS uses the web-friendly protocol HTTP, a REST API, and technology-independent data modeling to facilitate interoperability.

NMOS uses a shared common data model such as Nodes, Devices, Sources, Flows, Senders, and Receivers to represent resources on networked media devices. A **Device** is a logic block of functionality, and a **Node** is the host for one or more Devices. Devices have logical inputs (**Receivers**) and outputs (**Senders**). A **Flow** is a sequence of video, audio, or data that moves from a Sender to a Receiver or multiple Receivers. Each flow is associated with a **Source**.[ii] [iii]

IS-04: Discovery and Registration allows media nodes and their capabilities to be discovered. It essentially provides a database collection function or a **Registry** for networked media nodes. IS-04 exposes an HTTP **Registration API** that Nodes use to register their resources, and an HTTP **Query API** that applications use to find available resources. In addition, IS-04 exposes an HTTP **Node API** that applications use to find further resources on the Node.

When an NMOS node is first attached to a network, it will use the DNS Service Discovery (DNS-SD) protocol to locate the Registration and Discovery System (RDS), and then register itself and any associated resources via the registration API.

IS-05: Connection Management allows media nodes to be configured to send and receive IP streams. The specification provides an HTTP API for establishing and removing flows between Senders and Receivers. With IS-05 Device Connection Management, applications can manage connections without having to know the underlying transport protocol. It allows 'bulk' configuration management and for connections to be prepared and 'activated' at a particular time.

AWS Elemental Live appliances support NMOS IS-04 and IS-05. NMOS can be used to discover and control SMPTE 2110 uncompressed video connections, both as inputs to and outputs from events running on an Elemental Live appliance.

AWS Elemental Live does not auto-discover the NMOS registry using DNS-SD, so to configure NMOS on Elemental Live, need to edit the configuration file:

/opt/elemental-nmos/config/node-config.json

This is the file where the registry server information is configured. With this, Elemental Live will use the Registration API to register SMPTE 2110 resources.

Each AWS Elemental Live appliance registers itself as a Node in the NMOS Registry. An Elemental Live event supports NMOS-controlled inputs, registering each new input as an NMOS Device and each of its elementary streams as NMOS Receivers. Elemental Live also deletes and de-registers all associated NMOS Devices and Receivers when an input is deleted or its parent event stops. Similarly, an Elemental Live event supports NMOS-controlled outputs, registers and manages each new output as an NMOS Device, and registers associated elementary streams as NMOS Senders.

AWS Elemental Live will not expose any NMOS objects until an event is started, in the preprocessing state, or in the running state with an NMOS Input or Output. As soon as an event is started, Elemental Live will create an NMOS Node, corresponding to the Elemental Live appliance, an NMOS Device corresponding to the Elemental Live input or output, and a receiver or sender, each corresponding to one video, audio, or ancillary component of an AWS Elemental Live input or output. The following table lists the implementation of NMOS-controlled live SMPTE 2110 events.

	AWS Elemental Live	NMOS	NMOS Naming Convention
1	Appliance*	Node	nmos-live-<hosname>
2	Input	Device	<hostname>_<event number>_<input number>
3	Output	Device	<hostname>_<event number>_<input number>
4	Video, Audio, or Ancillary component of an Input	Receiver	<hostname>_<event number>_<input number>_<stream ID>
5	Video, Audio, or Ancillary component of an Output	Sender	[hostname] AWS Elemental [LiveEvent: <event number>] [OutputGroup: <output group name] [EssenseType_ID: <essense type id>]

6	Active 2110 connection	Flow	<hostname> - <Event number> - <Output Number> - <Audio/Video ID> - Flow

* An AWS Elemental Live Appliance with at least one event running with an NMOS enabled input or output creates the NMOS node.

Following is an example of an event configured in AWS Elemental Live with one NMOS-controlled SMPTE 2110 input, which includes one video elementary stream and one audio elementary stream by default. This example shows one NMOS-controlled SMPTE 2110 output with one video and one audio interface.

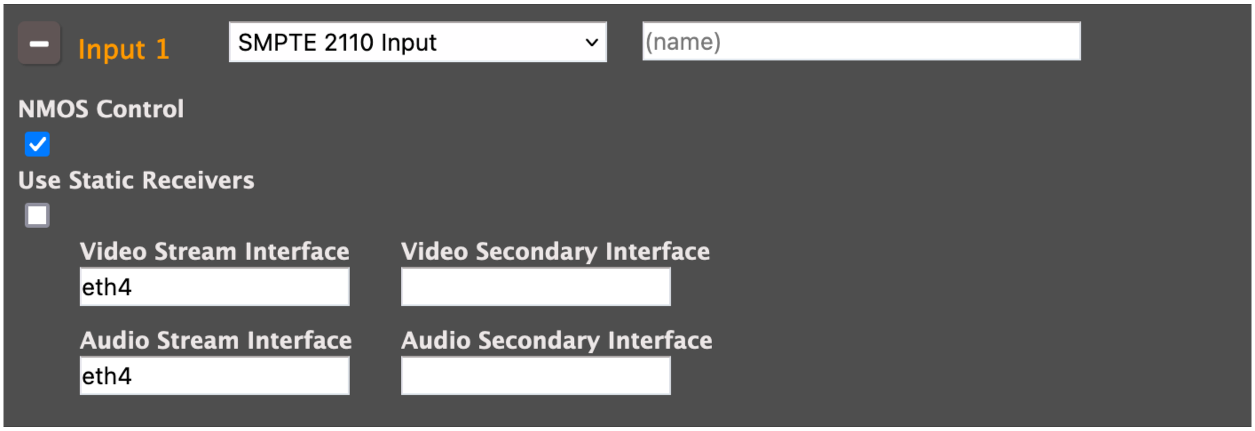

Figure 8: Screenshot showing input configuration, with NMOS Control enabled

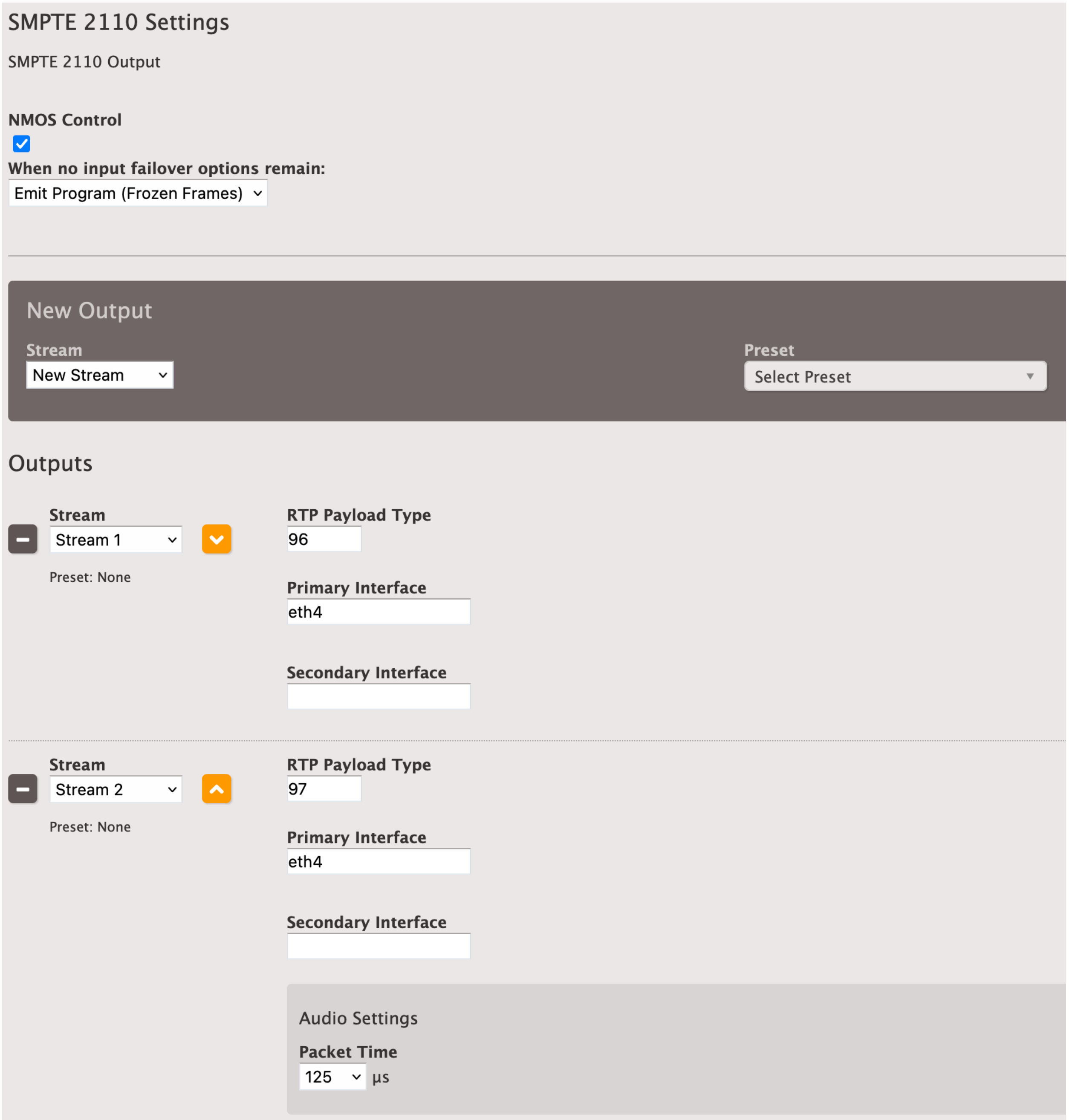

Figure 9: Screenshot showing the output SMPTE 2110 configuration with NMOS Control enabled

After the event starts, it will register the following resources to the NMOS registry. The following table assumes the AWS Elemental Live appliance hostname is 'chopper'. Other resources are not listed in this table, such as sources and flows.

1	Node	nmos-live-chopper	This is the node name for the Live appliance
2	Device	chopper_event1_input1	Each input is a device object
3	Device	chopper_event1_output1	Each output is a device object
4	Sender	chopper_event1_output1_video1	Each output elementary stream is a sender object
5	Sender	chopper_event1_output1_audio1	Each output elementary stream is a sender object
6	Receiver	chopper_event1_input1_video1	Each input elementary stream is a receiver object
7	Receiver	chopper_event1_input1_audio1	Each input elementary stream is a receiver object

JPEG XS

WHY DO WE NEED JPEG XS?

A 'forklift' approach from traditional SDI infrastructure to IP is not feasible, meaning SDI and IP will co-exist for some time. As content providers push towards 4K, and in some cases 8K, more stress is put on current infrastructure because of increased bandwidth requirements. The same limitations apply to contribution of video to the cloud. This drives the need for lightweight compression that preserves video quality while minimizing encode and decode latency. JPEG XS (ISO/IEC 21122), is a visually lossless, low-complexity, and low-latency video codec. JPEG XS targets visually lossless quality using intra-frame encoding and fixed-compression ratios.

JPEG XS can be used anywhere that uncompressed video is used. It can be used for low-latency applications like live remote production, VR/AR, gaming, autonomous vehicles, vehicle displays, and medical procedures. JPEG XS works well for contribution in ground-to-cloud video workflows, and is also an efficient way to transport video inside the cloud.

JPEG XS BENEFITS

JPEG XS is a visually lossless codec. Using fixed-compression ratios, the codec allows tuning of video quality and bits/second based on the target requirements for a workflow. In testing, up to ten encode/decode cycles have been performed before seeing any generational loss.

JPEG XS offers low-complexity encoding that allows for a smaller hardware footprint within a data center without the need for additional memory. The codec can run on both CPU-based and GPU-based systems while taking advantage of parallelism to achieve real-time or faster than real-time encoding. This low-complexity encoding cuts down on power consumption, reducing overall energy costs and promoting more sustainability in the broadcast industry. The codec also allows for both hardware and software solutions.

Finally, JPEG XS can provide sub-frame latency from encode to decode, allowing it to be used in latency-sensitive production workflows.

JPEG XS STANDARDS

JPEG XS Part 1 - ISO/IEC 21122-1:2019: Information Technology — JPEG XS Low-Latency Lightweight Image Coding System — Part 1: Core Coding System

- Specifies a decoding process for converting compressed image data to reconstructed image data.
- Specifies a codestream syntax containing information for interpreting compressed image data.
- Provides guidance on encoding processes for converting source image data to compressed image data.

JPEG XS Part 2 - ISO/IEC 21122-2:2019: Information Technology — JPEG XS Low-Latency Lightweight Image Coding System — Part 2: Profiles and Buffer Models

- Defines a limited number of subsets of the syntax specified in ISO/IEC 21122-1 and a buffer model to ensure interoperability between implementations in the presence of a latency constraint.

JPEG XS Part 3 - ISO/IEC 21122-3:2019: Information technology — JPEG XS Low-Latency Lightweight Image Coding System — Part 3: Transport and Container Formats

- Defines transport and container formats for JPEG XS codestreams as specified in ISO/IEC 21122-1. It defines file formats for working with still images and motion image sequence files on computer platforms and gives guidance on how to embed the codestream into transport streams, allowing internet-based communication.
    - JXS - JPEG XS File Format - Single Images
    - MP4 - ISOBMFF - Video File
    - HEIF - High-Efficiency Image File Format - Image and Video
    - MPEG2 TS - Transport Stream - Video/Audio/Metadata
    - RTP - Transport over the IP using Real-time Transport Protocol
    - 2110-22 - Transport using SMPTE ST 2110-22:2019 protocol

BITRATE COMPARISON

	Resolution/Frame Rate	SDI Format	JPEG XS Compression
1	HD 720p60 HD 1080i60/p30	HD-SDI - 1.485 Gbps	70 Mbps - 195 Mbps
2	FHD 1080p60	3G-SDI - 2.970 Gbps	150 Mbps - 390 Mbps
3	UHD 2160p30	6G-SDI - 5.940 Gbps	250 Mbps - 750 Mbps
4	UHD 2160p60	12G-SDI -12 Gbps	500 Mbps - 1.4 Gbps

Looking at the SDI format and the bitrates that uncompressed video requires, it's easy to understand the need for a low-latency, visually lossless codec. The bitrates are quite large with uncompressed data and stress the IP infrastructure. Using JPEG XS eases the stress of bandwidth constraints within a broadcast center. In the same vein, JPEG XS offers the same value for cloud contribution workflows. For cloud-only workflows, JPEG XS provides the means to move bits around the cloud network while maintaining low-latency and lossless-quality that uncompressed transport provides. One way to use JPEG XS in the cloud is with AWS Elemental MediaConnect, a managed service for transporting live video.

AWS CLOUD DIGITAL INTERFACE (CDI)

WHAT IS AWS CDI?

[AWS CDI](#) is a network technology that allows users to transport uncompressed video, audio, and ancillary data inside AWS with latency as low as 8 milliseconds. AWS CDI is provided as an open-source software development kit (SDK) that can be embedded inside Amazon Elastic Compute Cloud (Amazon EC2) services on both Linux and Windows. Data is transferred between EC2 instances in the same Availability Zone (AZ) and cluster placement group, and can support video up to 4K (UHD) resolution at 60 frames per second. AWS CDI was created to support Independent Software Vendors (ISVs), AWS partners, and AWS Media Services to create applications and workflows such as TV channel playout, live video production switching, motion graphic insertion, multi-viewer applications, video frame rate and color space conversion, forensic watermarking, and video encoding and decoding.

AWS CDI is built on two key components - EFA and SRD. At the heart of the technology is the use of the Elastic Fabric Adapter (EFA), a custom-built network interface created to allow high-bandwidth data transfer, with ultra-low latency and high-reliability by bypassing the kernel of the operating system. The second key component is the Scalable Reliable Datagram (SRD) protocol. This protocol transmits packets over a large variety of pathways all at once, swarming the receiving device with opportunities to receive any instance of the packet, while providing reliable out-of-order packet delivery keeping queuing to a minimum, in contrast to traditional network fabrics which use in-order packet delivery.

HOW DOES AWS CDI WORK WITH JPEG XS?

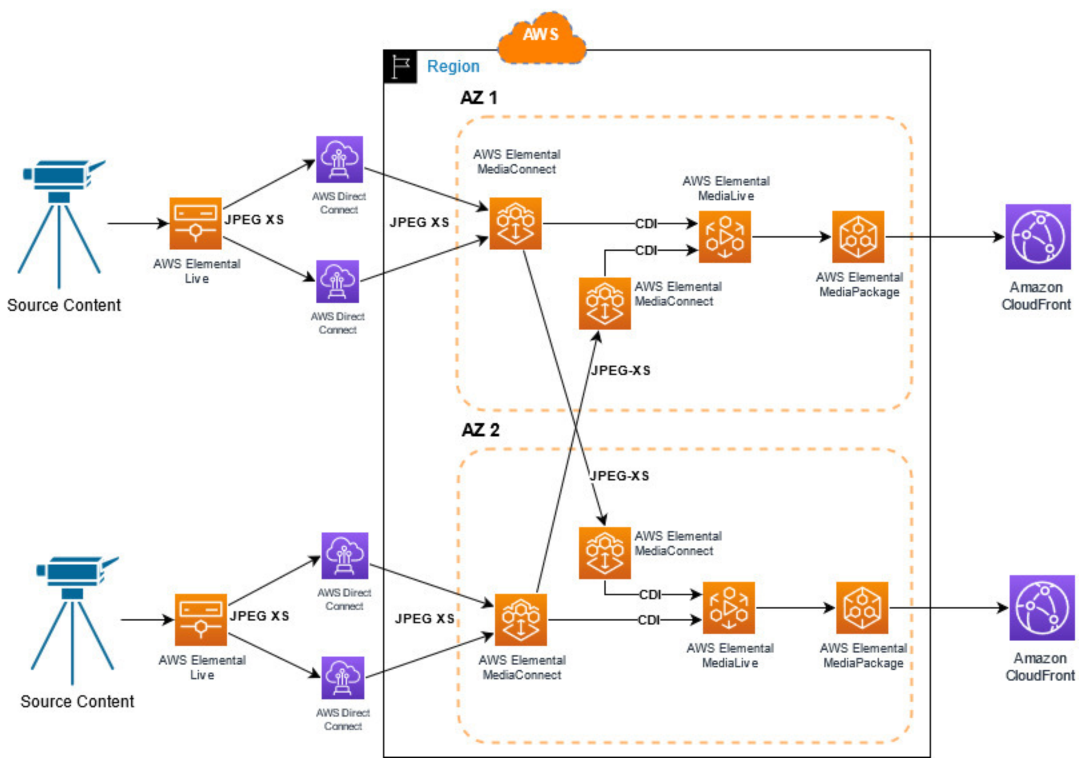

Figure 10: Diagram showing JPEG XS contribution to MediaConnect and AWS CDI into MediaLive

AWS Elemental MediaConnect provides a method to convert JPEG XS to and from AWS CDI. This allows customers to contribute SMPTE ST 2110-wrapped JPEG XS content up to MediaConnect using AWS Direct Connect, then hand off the AWS CDI output to AWS Elemental MediaLive or a partner's cloud-based service for further processing. Since MediaConnect supports multiple outputs per flow, parallel processing of the media can be achieved for live workflows. For hybrid workflows where an outbound distribution to on-premises encoders or receivers is required, MediaConnect can convert AWS CDI to a JPEG XS output and distribute it over a Direct Connect link.

When moving content across AZs is required, SMPTE 2110 JPEG XS is used for the transport between AZs. VPC peering can also be used to hand off content between two AWS accounts by creating a peering connection, then updating the main route table on the VPC with a route that includes the destination IP CIDR range that targets the peering connection id.

WHERE IS AWS CDI SUPPORTED?

AWS CDI is supported on EC2 appliances that are enabled with EFA. Below is a list provided from the EC2 documentation showing the instance types that can be used with AWS CDI:

- General purpose: m5dn.24xlarge | m5dn.metal | m5n.24xlarge | m5zn.12xlarge | m5zn.metal | m6a.48xlarge | m6i.32xlarge | m6i.metal
- Compute optimized: c5n.18xlarge | c5n.9xlarge | c5n.metal | c6gn.16xlarge | c6i.32xlarge | c6i.metal
- Memory optimized: r5dn.24xlarge | r5dn.metal | r5n.24xlarge | r5n.metal | r6i.32xlarge | r6i.metal
- Storage optimized: i3en.24xlarge | i3en.12xlarge | i3en.metal | im4gn.16xlarge
- Accelerated computing: dl1.24xlarge | g4dn.8xlarge | g4dn.12xlarge | g4dn.metal | g5.48xlarge | inf1.24xlarge | p3dn.24xlarge | p4d.24xlarge

CONCLUSION

This white paper identified and explained the reasons to transition from SDI to SMPTE 2110. It outlined the necessary components needed to successfully deploy SMPTE 2110 in a data center or other facility. It reviewed the technologies behind SMPTE 2110, NMOS, PTP, JPEG XS, and AWS CDI, and discussed how they can be deployed together to create hybrid workflows as well as ground-to-cloud and cloud-to-ground workflows. The transition to these modern technologies allows for greater scalability, versatility, and reliability, all while helping to future-proof a data center and move away from a cap-ex model to a cloud-based op-ex model, enhancing media workflows. The future of media is ultra-low latency, high-quality, reliable, and IP-based. SMPTE 2110, JPEG XS, and AWS CDI are at the forefront of this vision. Now let's go build on AWS.

FOOTNOTES

[i] Comparison of the three different sender types diagram, https://www.tvtechnology.com/opinions/smpte-st-211021-taming-the-torrents

[ii] Configuring PTP using ptp4l, https://access.redhat.com/documentation/en-us/red_hat_enterprise_linux/7/html/system_administrators_guide/ch-configuring_ptp_using_ptp4l

[iii] NMOS technical overview, https://specs.amwa.tv/nmos/branches/main/docs/Technical_Overview.html

CONTRIBUTORS

Brian Bedard, Sr. Edge Solutions Architect

Chris Zhang, Sr. Edge Solutions Architect

Matt Buk, Sr. Manager, Account SA

Dirk Young, Sr. Solutions Architect

Tim Addy, former Sr. Solutions Architect

Jason Ackman, Sr. Technical Account Manager