SMPTE ER-3:2017

ENGINEERING REPORT

Study Group on Flow Management in Professional Media Networks

Contents

Approved 2017-11-14

1 Introduction

The media industry is rapidly moving towards transport of uncompressed and compressed media content streams (flows) over packet switched facility networks. Media transport protocols (such as SMPTE ST 2022-6, VSF TR-03, IEEE AVB etc.) are currently being adopted by equipment manufacturers. While these address the transport of live media content, they do not address how these flows will be managed in professional media networks, nor do the standards address a standard methodology for manufactures in this space to ensure interoperability.

Several concepts of flow management have been proposed, but the feasibility of these approaches in media networks need to be investigated. Many of these technologies are under development today and there is currently no timeline on when these technologies will be available for professional media networks or when open standards will be developed. In addition, it is unclear how control messages will be transmitted to switching devices in a standard way. Existing solutions (e.g. OpenFlow, NETCONF, SMPTE ST 2071) could be utilized but it is unclear today if they will fulfill the requirements for management of low latency flows in professional media networks. Furthermore the media industry needs to define metrics on how network congestion can be measured to enable efficient stream control.

There might also be a lack of understanding in the media industry of network management technologies, how they will be applied in professional media networks consistent with operational needs in the tele-production community.

This report seeks to address these issues.

2 Report Scope

Investigate current and future professional media network management technologies, determine user requirements, transmission methods for management commands and provide background information. Provide an overview of existing standards and specifications available to the professional media industry today. Detail gaps in standardization and make recommendations on needed standards development.

- Define terminology used in this work and create a glossary
- Determine tools to measure network congestion
- Collect a list of user requirements which are specific to flow management in professional media network and which need to be met in professional applications
- Identify and analyze existing standards or specifications which could be applicable
- Provide an overview of current network management technology approaches (possible surveys) based on user requirements
- Compare and consider the limitations of current technologies
- Provide recommendations on standardizations needs (extension of existing standards or the creation of new standards) and recommendations to the industry in general

3 Glossary

For the purposes of this document, the following terms and definitions apply. A portion of these terms and definitions were adopted from IABM, two earlier efforts, one in SMPTE and one in the Joint Taskforce on Networked Media (JT-NM, although some were modified to fit the purpose of this report. Notes to definitions were added if clarification from the SG was needed.

API
Application Programming Interface; a set of interface definitions (functions, subroutines, data structures or class descriptions) which provide a convenient interface to the functions of a system. They also simplify interfacing work by insulating application programmers from intricacies of the implementation.

ASIC
Application Specific Integrated Circuit. Custom-designed integrated circuit with functions specifically tailored to an application.

Broadcast Controller
A system to provide high-level management of networked broadcast operations through a “northbound” API to an SDN Network Controller.

Configuration
A static collection of values for parameters that are required when a Node or Device is provisioned (JT-NM, 2015)
Examples of configuration parameters include:

- IP address and hostname of a node
- Setting a camera to NTSC or PAL mode

Content
Essence that adds value to an organization (JT-NM, 2015)
Note: Whether an essence adds value is not relevant for the discussion in this report

Control Interface
An Interface that allows control of the operation and parameters of a Device (JT-NM, 2015)

Control Surface
A user interface on a client that accesses one or more Control Interfaces (JT-NM, 2015)

COTS
Commercial off-the-shelf, usually referring to the use of common components from the computer market to create a product, or application.

Destination Device
A Device that has at least one Receiver, i.e. it has input(s). A video monitor is an example of a Destination Device (JT-NM, 2015)

Device
A media processor that has receivers and/or senders. (TR-03)
Example of Devices could include:

- Camera
- SDI to IP adapter
- Chroma keyer
- Audio mixer

A Device may have a permanent presence on its Node (**Fixed Device**, e.g., a networked camera), or it may be created on demand by its Node (**Virtual Device**, e.g., a software-based transcoder). Nodes may dynamically create different types of Devices (**Dynamic Device**). (JT-NM, 2015)

Elementary stream
A stream that only contains only one kind of data, e.g. audio, video or ancillary data. The audio type may be multi-channel. (TR03)

Endpoint
The network entry point to a running instance of an Interface (JT-NM, 2015)

Essence
The material that television and radio programs are made of. In other words, the video, audio and any other material such as graphics and captions that are added to make up the final result.

Flow
A flow may refer to either a sequence of timed media elements (audio, video, or metadata) conveyed from a sender to a receiver or the transport of this data over a network. Flows often include regular timing marks for receivers to recover the element(s) and align to other related flows. A single flow may contain a distinct media element (e.g. audio or video) or be composed of multiple elements (e.g. audio and video).

Grouped Essence Flows (GEF)
Any combination of two or more MEFs and/or SEFs logically associated, grouped, at the output of a sender or within the network or at the input of a receiver. The intent is for synchronized presentation and logical carriage of related flows.

Interface
Provides a specified access to some functionality over a network. May be advertised for discovery purposes (JT-NM, 2015)

Latency
The time delay experienced in a system. In a packet switched network, latency may be the one-way delay from the source transmission of a packet until its destination, or it may be a round-trip delay of a packet from the source to a destination and then back to the source. (SMPTE, Report of the SMPTE Study Group on Media Production System Network Architecture, 2014)

Loss
In a packet switched network, packet loss is the dropping of a packet by a network element such that the packet does not arrive at its desired destination. (SMPTE, Report of the SMPTE Study Group on Media Production System Network Architecture, 2014)
Note: A packet delayed beyond the useful time frame for its delivery may be considered as packet loss.

Media
Audio, video, and other means to carry general communication, information, or entertainment (JT-NM, 2015)

Multiple Essence Flow (MEF)
A single real-time media stream that is composed of two or more aligned essence types. For example, SDI and SMPTE ST 2022-6 are MEFs since the transport payload concurrently carries time aligned audio, video and ancillary metadata.

Monitoring Interface
An Interface allowing a Monitoring Tool to obtain information about the status of a Device (JT-NM, 2015)

Monitoring Tool
A tool used to obtain information about the status of Devices (JT-NM, 2015)
Note: Flows may also be monitored by such a tool.

NETCONF (Network Configuration Protocol)
The Network Configuration protocol that provides mechanisms to install, manipulate, and delete the configuration of network devices, defined by IETF RFC 6241. It uses an Extensible Markup Language (XML)-based data encoding for the configuration data as well as the protocol messages. (From IETF RFC 6241, https://tools.ietf.org/html/rfc6241)

Network
Infrastructure for the interconnection of Nodes (JT-NM, 2015)

Network Controller
In an SDN network, the Network Controller manages flow control of the switches/routers (via southbound APIs) based on guidance from applications and business logic (via northbound APIs) to deploy intelligent networks.

Node
A logical host, acting as a container of one or more Devices. A Node may have a permanent physical presence, or may be created on demand, for example as a virtual machine in a cluster or cloud. (JT-NM, 2015)

Packet
A block of binary data that is grouped into a suitably size for transmission over a data network (SMPTE, Report of the SMPTE Study Group on Media Production System Network Architecture, 2014)

Packet switching
A digital network communications method that groups transmitted data into packets (SMPTE, Report of the SMPTE Study Group on Media Production System Network Architecture, 2014)

Packet Switched Network
A digital networking communications method that groups all transmitted data into blocks called packets, which are transmitted over a shared network that allocates transmission resources using statistical multiplexing or dynamic bandwidth allocation. (SMPTE, Report of the SMPTE Study Group on Media Production System Network Architecture, 2014)

Payload
The operational data that is being carried within a packet or other transmission unit. The payload does not include the "overhead" data required to get the packet to its destination. What constitutes the payload may depend on the point-of-view. To a communications layer that needs some of the overhead data to do its job, the payload is sometimes considered to include the part of the overhead data that this layer handles.

PIM
Protocol-Independent Multicast (PIM) is a family of multicast routing protocols for Internet Protocol (IP). It is termed protocol-independent because PIM does not include its own topology discovery mechanism, but instead uses routing information supplied by other routing protocols.

PMN
Acronym for Professional Media Network. (SMPTE, Report of the SMPTE Study Group on Media Production System Network Architecture, 2014)

Processing Device
A Device that has at least one Sender and at least one Receiver, i.e. it has input(s) and output(s). A transcoder is an example of a Processing Device. (JT-NM, 2015)

Professional Media Systems
Systems operated by professionals to acquire, manipulate, edit, process, and distribute media. (SMPTE, Report of the SMPTE Study Group on Media Production System Network Architecture, 2014)

Professional Media Network
A network infrastructure to support some or all of the activities of a Professional Media System. (SMPTE, Report of the SMPTE Study Group on Media Production System Network Architecture, 2014)

QoS
Acronym for Quality of Service. (SMPTE, Report of the SMPTE Study Group on Media Production System Network Architecture, 2014)

Quality of Service
The ability to provide different priority to different applications, Users, or data flows, or to guarantee a certain level of performance to a data flow. (EBU, 2014)

Receiver
A consumer of a single RTP elementary stream (TR-03)
Note: For the purpose of this document other transport protocols are included in this definition

Representational State Transfer (REST)
An architectural style for distributed hypermedia systems. Its six guiding architectural constraints are client-server, stateless, cacheable, layered system, uniform interface, and code on demand (optional).

RESTful
Web service APIs that adhere to the REST architectural constraints, typically using requests made to a resource's URI over HTTP using standard HTTP methods (such as GET, PUT, POST, DELETE).

RTP
Real-time Transport Protocol (RTP, IETF RFC 3550) is a network protocol that is typically used for streaming video and audio over an IP Network. RTP usually runs over UDP (User Datagram Protocol) offering point-to-multipoint and typically greater throughput than TCP.

RTP Session
An association among a set of participants communicating with RTP. (IETF RFC 3550)
Note: A participant may be involved in multiple RTP sessions at the same time. In a multimedia session, each medium is typically carried in a separate RTP session. A participant distinguishes multiple RTP sessions by reception of different sessions using different pairs of destination transport addresses, where a pair of transport addresses comprises one network address plus a pair of ports, one for RTP and (optionally) one for RTCP. (TR-03)

SDN
Abbreviation for Software Defined Networking. For a detailed description of SDN see section 5.2.

Sender
A producer of a single RTP elementary stream (TR-03)
Note: For the purpose of this document other transport protocols are included in this definition

Single Essence Flow (SEF)
A single real-time media stream that is composed of only one essence type; audio or video or ancillary metadata. For example, AES3 audio and VSF's TR-03 video/RTP are SEFs.

SLA
Acronym for Service Level Agreement (SMPTE, Report of the SMPTE Study Group on Media Production System Network Architecture, 2014)

SLA Metrics
Refer to Service Level Agreement Metrics (SMPTE, Report of the SMPTE Study Group on Media Production System Network Architecture, 2014)

Source
An abstract concept that represents the primary origin of a Flow or set of Flows. A Device may provide several Sources e.g. a video camera has separate video and audio sources (JT-NM, 2015)

Source Device
A Device that has at least one Sender, i.e. it has output(s). A camera is an example of a Source Device. (JT-NM, 2015)

Stream
A realization of a Flow (see definition above) using a transport protocol such as RTP or HTTP.

Synchronous
Occurring at the same time; coinciding in time; going on at the same rate, exactly together, and exactly in phase with one another. Synchronous refers to the relationship between two or more signals or things. (SMPTE, Report of the SMPTE Study Group on Media Production System Network Architecture, 2014)

Synchronization
The mechanism for achieving a synchronous relationship between flows.

Three Planes Model
See Annex C “The Three Planes Model for Media Systems”

Timecode
A representation of a time within a video with a precision of frames. The Epoch of Timecode is often implementation-specific. When used with fractional frame rates, some Timecode values may be skipped to maintain long-term alignment with the frame cadence (this is known as “*dropFrame*” Timecode). (JT-NM, 2015)
Note: Timecode is not limited to use in video

Time Stamp
An *absolute* time (that is, an offset from the start of an Epoch) describing the relationship of a Grain with a Clock (JT-NM, 2015)

Transmission Control Protocol (TCP)
The Transmission Control Protocol is an extension of the Internet Protocol intended to be a highly reliable host-to-host protocol for the transmission of data between hosts on a Packet Switched Network. The Transaction Control Protocol guarantees the delivery and prevents the duplications of messages. Please refer to IETF RFC 793 for greater detail. (SMPTE, Report of the SMPTE Study Group on Media Production System Network Architecture, 2014)

User Datagram Protocol (UDP)
The User Datagram Protocol is an extension of the Internet Protocol defined to allow for the applications to transmit of messages between network nodes with a minimum amount of overhead using the Internet Protocol. User Datagram Protocol is transaction oriented and does NOT guarantee delivery or provide duplicate protection. Please refer to IETF RFC 768 for greater detail. (SMPTE, Report of the SMPTE Study Group on Media Production System Network Architecture, 2014)

YANG
YANG is a data modeling language, defined in IETF RFC 6020, used to model configuration and state data manipulated by the Network Configuration Protocol (NETCONF), NETCONF remote procedure calls, and NETCONF notifications. (From IETF RFC 6020, https://tools.ietf.org/html/rfc6020)

4 What Is Flow Management?

Flow management is the mechanism for controlling how real-time media flows emitted by senders are guided through a fabric of Ethernet switches to receivers with appropriate functionality of media clean switching, reserved media bandwidth, and guaranteed required QoS.

PMN systems tend to use UDP rather than TCP. UDP is commonly used in the IP domain where low latency is an important criteria as is in the case of PMN. Since UDP does not guarantee delivery or ordered packets, a network has to be managed to provide guaranteed QoS and zero packet loss.

The size of PMN flows also tends to be much larger than the size of typical IP delivered flows. Instead of a datacenter where tens of millions of HTTP connections are handled, PMNs may have hundreds or thousands of media flows the size of 1 Gbps or more. Typical link aggregation (LAG) protocols that depend on hashing flow headers are unlikely to work well with PMN flows. Finally, PMN flows tend to inherently require point-to-multipoint reception, as many monitoring and test systems may need to view media flows at the same time. Thus PMN flows tend to be multicast IP.

Managing PMN flows through a fabric of Ethernet switches is a very different task than flow management of a typical datacenter.

This report focuses on flows transported using RTP but other transports, e.g. HTTP/DASH, are also found in PMN.

5 Network Switch Architecture Overview

Network switching devices (such as an Ethernet switch) typically implement the “data plane”, and the “control plane” of the *Three Plane Model* (see Annex C) as seen in Figure 1. The “management plane” is generally external to network switching devices. Interfaces, IP subnets and routing protocols are configured through management plane protocols, which can range from manual CLI command entry to APIs such as NETCONF. The control plane utilizes these configurations to establish simple forwarding rules that are utilized by the data plane to send packet traffic to a selected destinations. A modern Ethernet switch must handle billions of packets per second, generally requiring the use of ASIC-based hardware to achieve this level of performance. The switching decisions can be based on many factors including the matching of packet headers, ingress interface that a packet arrives on, or even deep packet inspection for handling higher-level protocols. The switching silicon commonly used in COTS Ethernet switches uses different types of forwarding tables to handle different needs. These tables are used in a pipeline that processes each packet in order to make a switching decision. The most flexible forwarding tables involve the use of a complex, high speed type of computer memory known as TCAM (Ternary Content Addressable Memory) to enable high speed search of routing data. While TCAM is very flexible and offers very high performance, it is also very expensive in terms of silicon real estate, power use and generation of heat. Therefore, TCAM space is limited in most COTS switches. Other, larger tables are available within the switching silicon for more limited switching decisions involving “exact matches” against portions of the packet, such as MAC address matching (L2 switching) or destination IP address exact match. It is important to understand that different types of switching decisions utilize different types of tables within the switching ASIC, thus exhibiting different scaling characteristics.

The control plane runs more complex algorithms that set up the rules on the data plane. Often the control plane is built on top of standard computer hardware and operating systems since it operates at much slower speeds than the data plane. The control plane communicates with other network devices in order to generate its own view of the network. Control plane algorithms may include Spanning Tree Protocol (STP), Enhanced Interior Gateway Routing Protocol (EIGRP), Open Shortest Path First (OSPF), Intermediate System to Intermediate System (IS-IS), and many others in order to ensure a stable and optimally-routed network.

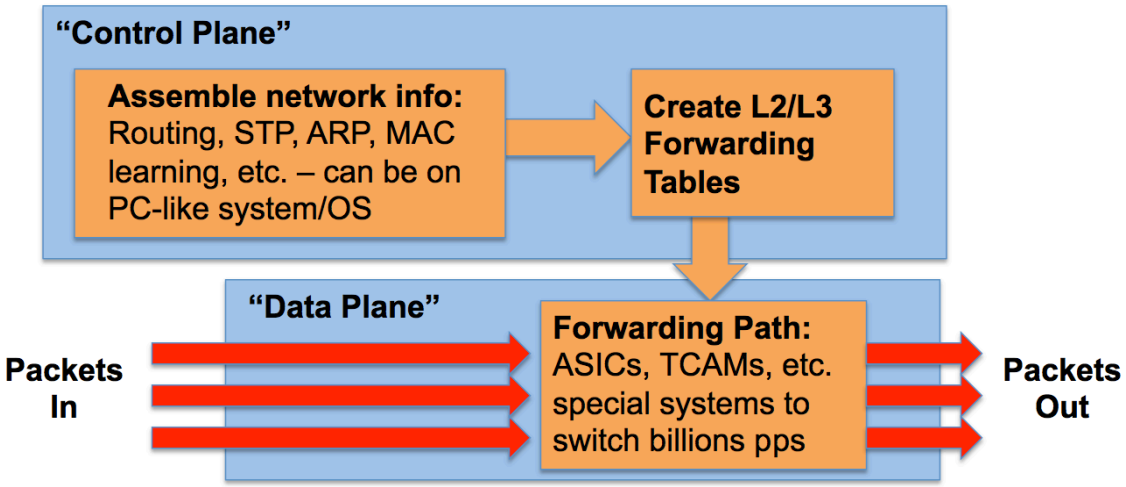

Figure 1: Control and Data Planes of Network Switching Devices

The data plane and control plane have typically been bundled together into a unified element, and the data plane forwarding path has been controlled only through built-in control plane algorithms. Alternatively, some portions of the control plane may also be offered by an external entity such as a network controller. The steering of traffic follows normal Ethernet and IP behavior. However, professional media networks require the ability to dynamically steer video flows based on some form of application control; for example, triggering video switching from a control panel. This requires programmatic control of network behavior. Such programmatic control can be achieved in multiple ways:

- Through the use of APIs to interface with the built-in control plane (which in turn, configures the data plane)
- Through the use of APIs to interface with the external control plane (which in turn, configures the data plane)
- Through the use of APIs to interface directly with the switch data plane

5.1 Multicast IP and IGMP

The delivery of video content in a professional media network usually encompasses a one-to-many relationship; a source needs to be received by multiple listeners. When utilizing IP as a transport protocol, there are well-established protocols and techniques to achieve a one-to-many delivery of data – namely IP Multicast. Thus many IP transport strategies permit the use of IP Multicast (e.g. SMPTE ST 2022-6).

Traditional IP Multicast networks utilize control protocols to allow listeners to subscribe to (receive) and unsubscribe (drop) multicast streams. These control protocols can be used to implement flow management. There are two protocols that are used with IP Multicast. IGMP (Internet Group Management Protocol) as defined in IETF RFC 1112, RFC 2236 (v2) and RFC 4604 (v3) is used in IPv4 between the receiver and the network switches. The corresponding protocol for IPv6 is Mulitcast Listener Discovery (MLD). PIM (Protocol

Independent Multicast) is used within the network fabric to route the flow from the switch facing the sender to the switch facing the receiver. The flow is routed in the network fabric by monitoring IGMP membership to ensure efficient use of bandwidth within the fabric.

The use of PIM as a protocol for the control of video distribution in a professional media network may not be desirable for multiple reasons:

- Complexity of establishing and maintaining a multicast control plane
- Relatively long timeframes for joining and dropping group membership (e.g. subscribing to and dropping streams)
- Lack of visibility as to how the multicast traffic is carried in a multi-hop network (without the use of special tools such as sFlow), and lack of control of multicast traffic flow with the potential for congestion and lossy transmission.
- No guarantee of QoS determinism

5.1.1 Recommendation on PIM

For effective management of flows in a PMN the Study Group members recommend that PIM should not be relied upon

5.2 Software Defined Networking (SDN)

Software Defined Networking is a promising technology for steering flows thru a PMN. In general, the term Software Defined Networking (SDN) refers to the ability to programmatically control network behavior and therefore how flows will traverse a network. Some forms of SDN provide a higher level of programmatic control of the data plane outside of typical built-in control plane features, as well as a higher level management plane driven by business applications. SDN may involve separating the data plane from the control plane; that is, the data plane might be housed in physical devices separate from a "controller," and a single controller can control a number of data plane switches as shown in Figure 2. This allows the controller to have a more unified view of a complex network. (*Note that this diagram is a simplification, there can be elements of management and control spread across actual network devices and control components*).

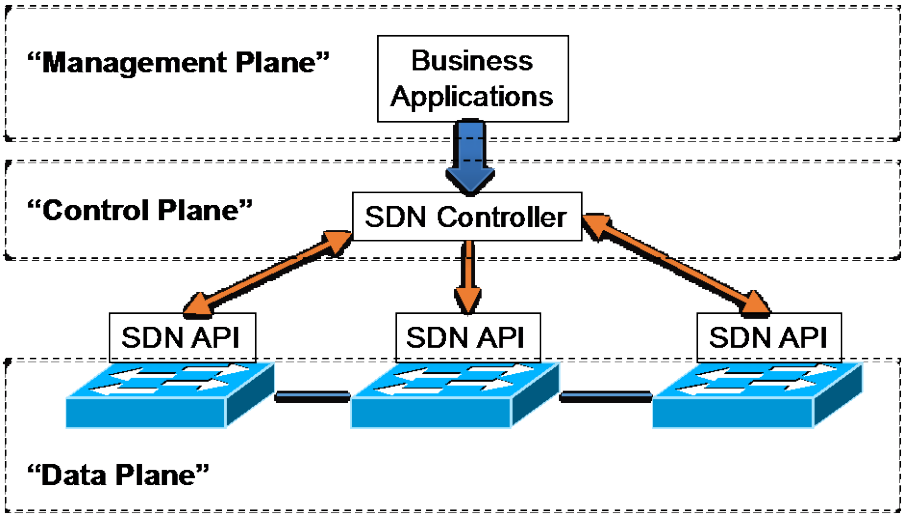

Figure 2: An example of Software Defined Networking

For flow management purposes the PMN application interfaces with the SDN controller using the "northbound" API. These APIs are usually REST-based. This API is being worked on in the AMWA-NMOS Network Control API working group. There is a need for creating a common specification or standard to permit a multi-vendor deployment. Different broadcast infrastructure control systems can interface with a common SDN controller, which, in-turn, can interface with a variety of network switches. The SDN controller speaks to SDN switches through a "southbound" API. OpenFlow is one such standardized southbound API, as seen in Figure 3. OpenFlow rules have "match fields" including ingress switch port, Ethernet source & destination, Ethernet type, IP protocol version, IP source & destination, and TCP/UDP source & destination ports. Some match fields can be implemented with full or partial wildcards. If a packet's headers match the match fields, then an action occurs on that packet. The actions can include forwarding a packet to a port, encapsulating the packet and forwarding it to the controller for analysis, dropping the packet, placing the packet in a queue, or allowing the packet to be processed by the "normal" Ethernet switching pipeline. OpenFlow also provides for packet and byte counters on the switch, which can be queried by the controller to aid in determining optimal network behavior. Another example of this southbound API is NETCONF (IETF RFC 6241).

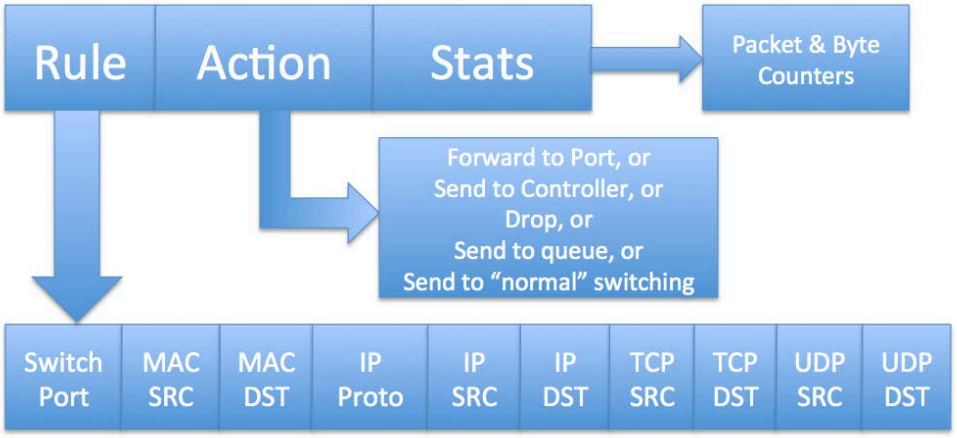

Figure 3: Elements of OpenFlow

There are a number of different SDN strategies from different vendors, but they all achieve the same goal: the ability of business applications to have fine-grained control over network flows.

A key element of SDN strategies is the "northbound" interface between business applications and the SDN controller. These are often application-specific.

The OpenDaylight Project is an example of an effort to create an open source SDN controller. It is a highly available, modular, extensible, scalable and multi-protocol controller infrastructure built for SDN deployments on modern heterogeneous multi-vendor networks. It provides a model-driven service abstraction platform that allows users to write applications to interface with northbound APIs like Openflow that easily work across a wide variety of hardware and southbound protocols.

A key consideration in designing a programmatically controlled network for steering video flows is the scaling limitations that might be exhibited by COTS based switches. Some professional media networks will only have to steer tens or hundreds of video flows. However, the replacement of a large scale SDI video router may need to steer thousands or tens-of-thousands of video flows.

An example of such scaling limitations is the use of Openflow as a southbound API in a software defined network. In many COTS switches, Openflow programs switching decisions into the TCAM table. As

discussed previously, TCAM size is limited and thus the number of “rules” that can be programmed may be smaller than desired (e.g. 1,000 – 2,000). Other types of programmatic control can overcome these limitations by using non-TCAM tables that are larger in size. For instance, the programming of static multicast port membership through the built-in control plane will often utilize exact-match tables in the switching silicon, thus overcoming TCAM limitations.

5.3 Programmable Data Plane

A recent development in networking devices is the ability to flexibly program behavior of the data plane. Non-programmable data plane SDN APIs (such as OpenFlow) may be limited by implementation decisions made in the hardware data plane, such as a fixed number and format of packet headers to match against, and a fixed set of actions that can result from match/action tables. Also communication between the data plane and an external programmable controller can result in a high level of latency between control decisions and re-configuration of the data plane.

Programmable data plane technology enables a more flexible method of controlling packet processing that can be performed at line-rate. The packet processor parser can be programmed to extract any header desired, and match/action tables based on those headers can be more complex than in non-programmable switching solutions.

P4 is an example language for expressing how packets are processed by the data plane of a programmable network device. The name P4 comes from the original paper that introduced the language, “Programming Protocol-independent Packet Processors” by Bosshart et al. P4 programs have been written, for example, to provide switching of RTP flows based on their timestamp.

5.4 Precision Time Protocol

Precision Time Protocol (PTP, standardized as IEEE 1588-2008) is a protocol used to synchronize clocks on a computer network. On a local area network, it achieves synchronization accuracy in the sub-microsecond range. PTP uses a master-slave architecture where master clocks transmit time synchronization information to slave clocks over the network. The root timing reference is called a grandmaster, and is often connected to a high-accuracy clock source such as GPS. SMPTE ST 2110-10 states that a Common Reference Clock should be provided and distributed on the network by means of PTP**.** SMPTE has defined a PTP Profile for professional broadcast applications in SMPTE ST 2059-2. AES has defined a PTP Profile (the "Media Profile") as part of AES67 and has issued a report AES-R16 on PTP parameters for interoperability between SMPTE ST 2059-2 and AES67.

Without going into the full details of how PTP works, it should be noted for flow management purposes:

- In its most usual deployment mode, the PTP messages are any-source multicast (ASM)
- Some switches provide special support for the accurate distribution of PTP across switches (Transparent Clocks) and between network segments (Boundary Clocks)
- Only one grandmaster is active on the network at one time, elected by a best master clock algorithm (BMCA), which gives a degree of automatic failover between grandmasters
- IETF RFC 7384 “Security Requirements of Time Protocols in Packet Switched Networks” defines a set of security requirements for time protocols (including PTP), discusses the security impacts of time protocol practices, the performance implications of external security practices on time protocols, and the dependencies between other security services and time synchronization.

6 Methods of Flow Switching

This section describes the three main methods of flow switching which can be utilized for flow management. They differ by the location in the network where the switching takes place:

- On the source device through source-timed video switching
- On the destination device through destination-timed video switching
- On the network switch as switch-timed video switching.

The 3 methods are describe in detail below. There is further differentiation between clean switching and non-clean switching which is describe below as well.

6.1 Methods of Clean Switching Packetized Media

The “clean” switching of live video is a key capability of a broadcast plant. Three potential strategies for clean switching of real-time packetized video streams are shown in Figure 4.

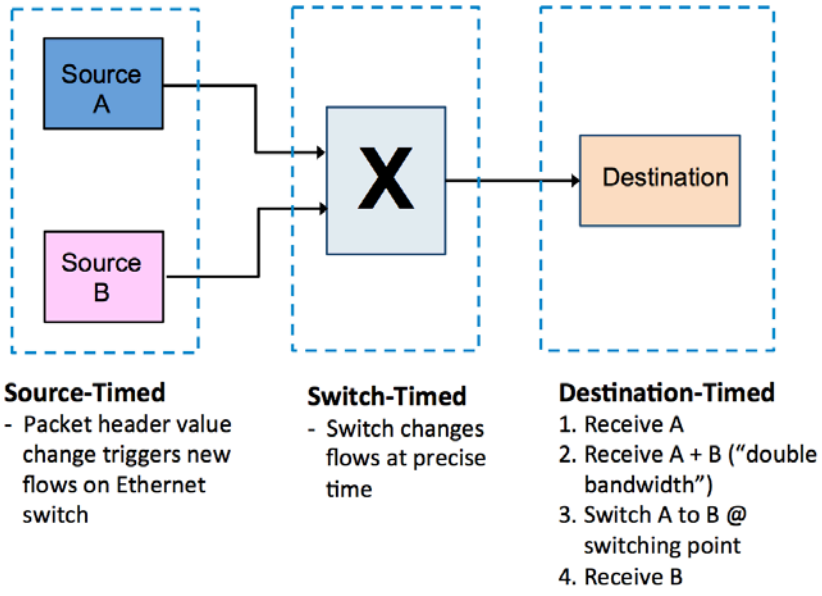

Figure 4: Three Ways to Clean Switch Packetized Video

6.1.1 Switch-Timed Video Switching

Switch-time video switching requires the network switch to change packet flow rules at precise times (such as the SMPTE RP 168 switch point for SDI carried in SMPTE ST 2022-6, or at the beginning of a video frame of active uncompressed video in SMPTE ST 2110-20). The switch-timed video switching technique suffers from the fact that today, common-off-the-shelf (COTS) Ethernet switches generally cannot provide the temporal accuracy of flow changes required for clean video switching. For example, to change flows within the SMPTE RP 168 area for 720p/59.94 video in SMPTE ST 2022-6, the required accuracy would need to be within the length of time represented by one SMPTE ST 2022-6 datagram, or about 7.4 µs. Unfortunately, reported OpenFlow rule update rates of COTS network devices from Bilbery et al. and Mogul et al. range from 100 to 1000 per second, implying an upward limit of temporal precision between 1 ms to 10 ms. Also, any lack of synchronization between the arrival of the video fames at the switch could cause a failure of the clean switch.

Programmable data plane techniques can allow a network device to perform matches on elements of the packet payload, such as the RTP timestamp. This has been shown in a proof-of-concept demonstration at NAB 2017. If senders' internal clocks are synchronized (via PTP, for example), control systems can pre-program desired future timestamps for RTP video to be switched.

6.1.2 Destination-Timed Video Switching

Destination-timed video switching requires the destination to receive the video flow to be switched into before ending the reception of the video flow to be switched out of ("make before break"). During that time, the destination buffers data from both video flows, and the destination device itself can determine the precise video switch point. This solution can be readily implemented using IGMP joins & leaves emitted from the destination device. However during the period between joining the new multicast group and leaving the old multicast group, one or more data paths through the network will have to carry twice the bandwidth of the media flow. In particular, the "last mile" Ethernet switch port that the receiving device is directly attached to is likely to be limited to handling half of the media flows that it could otherwise carry in a steady-state environment to avoid potential blocking.

It should be noted that IGMP could be implemented using typical networking, or it could be implemented through SDN. For example, an IGMP command could be forwarded by a switch to the SDN controller, and then acted upon by the controller installing new flow rules on the switch. Destination-timed switching could also be performed by a broadcast controller providing northbound commands into the SDN controller directly.

Inherently, destination-timed video switching requires buffering, and is thus capable of handling non-synchronized frame arrival.

6.1.3 Source-Timed Video Switching

The concept of source-timed video switching is to separate the temporally inaccurate process of updating Ethernet switch packet forwarding rules from the actual precise timing of a clean video switch. To do this, a specific element of the packet header is selected as the "timing signal" match field whose value will be used to trigger a precise flow change by matching rules previously configured using SDN on the network devices. Preferably this header will have little or no impact on other stream processing functions. In one proof-of-concept, the UDP source port value was used as the packet header to change. Like switch-timed video switching, source-timed video switching also requires synchronized arrival of video frames at the network switch.

6.2 "Break-before-Make" Video Switching

"Break-before-Make" packetized video switching is a sub-case of destination-timed switching that is not perfectly "clean," but may be good enough to many use cases. IGMP is used to leave a multicast group of video before joining the next multicast group of video, to avoid incurring the expense of extra stream bandwidth during the overlap of join before leave. The last frame of the left stream is repeated until the new stream has been acquired. With frame times of 16ms or longer, Break-before-Make switching with a single frame repeat can be accomplished with many COTS network switches. This mechanism is appropriate for a number of non-critical use cases, such as monitoring. Even so, many viewers will be unable to perceive a single frame repeat, especially at the end of a fade to black. A frame repeat might be visible in action video, and to some broadcast engineers this is problematic.

6.3 Recommendation for Methods of Flow Switching

The SG recognizes that destination-timed switching using IGMP and switch-timed switching using SDN methods are the most popular forms of flow switching. But it also recognizes that it is not advisable to standardize one specific flow switching method at this point in time. Technology for the different switching methods and the implementation of IP infrastructure are still evolving. The SG recommends that determining the best flow switching method should be left to users, equipment manufacturers or system integrators.

7 Control Protocols

This section describes a number of control protocols over IP, and how they may interact with flow control. In order to have coherent, multi-vendor solutions to professional media networking, it is important that there be standards for broadcast control systems to interoperate with a universe of network devices to ensure that flows are delivered where they are required, that flows do not over-subscribe network resources, and that unauthorized traffic cannot impede the performance of desired flows.

7.1 Advanced Media Workflow Association (AMWA) NMOS Specifications

7.1.1 Introduction to the NMOS Specifications

The Advanced Media Workflow Association (AMWA) is an open, community-driven forum, advancing business-driven solutions for Networked Media workflows. It is developing a family of Networked Media Open Specifications (NMOS). NMOS uses a logical data model based on the JT-NM Reference Architecture (JT-NM RA) to add identity, relationships and time-based information to content and broadcast equipment. The NMOS APIs are being developed in an “agile” fashion, providing useful deliverables to implement User Stories, with functionality actively tested in Networked Media Incubator interop events. Some of the APIs have been published as AMWA Interface Specifications (IS).

AMWA's architectural sprint identified different "layers" of connection in networked media systems and management of such connections. At a lower layer there is control of the raw streams that go over the network fabric. This is being addressed in the Network Control API, which will be designated IS-06. At a higher layer, there is the creation and identification of the "logical" connections over which elemental content flows between devices, independent of the network protocols used. Creation of connections is dealt with by the Connection Management API, which will be designated IS-05. Identification is dealt with by IS-04."

It should be noted that security and authentication methods for the AMWA NMOS APIs have not been formally specified yet.

7.1.2 AMWA IS-06: Network Control API

The objective of the AMWA Network Control API is to develop a multi-vendor, interoperable common interface for flow management in PMNs based on software defined network (SDN) technology. The API would exist between a broadcast controller (BC) and a network controller (NC).

The NC concept provides an abstraction of the network, and is typical of SDN systems. The interface between the NC and the network elements (the “southbound” interface) usually depends on the control capabilities of the network elements themselves, and could be NETCONF/YANG, OpenFlow, or a proprietary format like command-line emulation. OpenDaylight SDN controller is an example of an open source NC that

could be extended to implement this API as a “northbound” interface. It should be noted that development effort will be required to enable an SDN controller to support the Network Control API.

The BC is responsible for defining policy and workflows in the network. The BC authorizes endpoints and flows, and the NC enforces those on the network. The level of control can be such that no packets flow on the network that are not authorized by the BC. Another option is that the BC has complete control over a particular media priority QoS level on the network, allowing other lower-priority traffic to move through the network on an as-bandwidth-is-available basis. Figure 5 shows a system diagram of the proposed Network Control API.

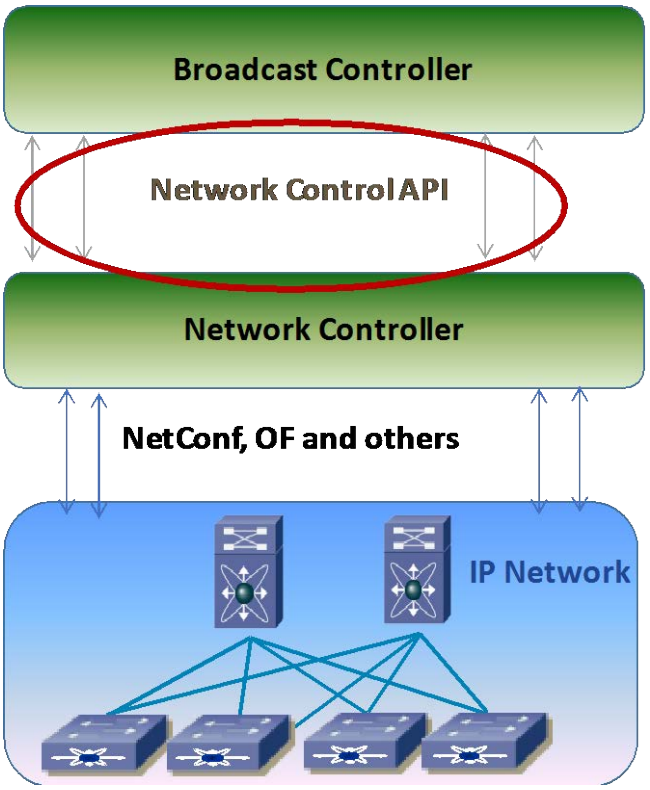

Figure 5: System Diagram of Proposed Network Control API

The API is proposed to use a Representational State Transfer (REST)-based set of operations that provide a secure and authenticated control channel between the BC and the NC. It would allow the BC to:

1) Discover network topology
2) Receive diagnostic messages about the network
3) Register network endpoints
4) Permit media flows between specific senders and listeners

The API can be supported in a model where listeners can issue an IGMP join to receive an authorized flow, or a model where network transmits the flow towards the listener and the listeners “promiscuously” receive (i.e. receive all IP packets transmitted to the device) their authorized flows.

Work is underway in the AMWA for a potential development into an interface specification IS-06.

7.1.3 AMWA IS-04: NMOS Discovery and Registration

IS-04 NMOS Discovery and Registration API is an AMWA Specification which provides a way for network-connected devices to become listed on a shared registry; it provides a uniform way to query the registry.It also describes a method for peer-to-peer discovery in order to permit operation on a link-local only or smaller network deployment. NMOS IS-04 multi-vendor interoperation has been shown at the IBC 2016 IP Interoperability Zone and the NAB 2017 and IBC 2017 IP Showcases.

The base data model of NMOS describes Nodes, Devices, Sources, Flows, and Grains. Nodes are logical hosts for processing and network operations. Nodes provide Devices, which are logical groupings of functionality. Devices with the capability to originate content must provide one or more Sources, which are abstract logical points of origin for one or more Flows, which themselves are concrete representations of content. Flows are composed of sequences of Grains. A Grain represents an element of essence or other data associated with a specific time, such as a frame, a group of consecutive audio samples, or captions. Grains also contain metadata information that specifies attributes such as the temporal duration of the payload, useful timestamps, originating Source ID, and the Flow ID the grain is associated with. Devices transmit and receive Flows over the network using Senders and Receivers.

Information regarding available Nodes, Devices, Sources, and Flows available through queries of the IS-04 API, and descriptions of this API are being made available on the public web-based repository hosting service, GitHub. More information can be found at:

http://www.amwa.tv/projects/IS-04.shtml

Flow control systems likely need to be aware of Nodes, Devices, Sources and Flows to understand the operational requirements of a professional media network. This information provides critical information in the areas of flow authorization, establishment, and bandwidth reservation.

7.1.4 AMWA IS-05: Connection Management

AMWA IS-05 will provide a general mechanism for managing connections between Senders and Receivers. It will be applicable to various transport protocols (not just RTP) and to unicast or multicast delivery. It will allow management of single connections or groups of connections and it will support seamless protection using multiple streams. The API has been tested at NMOS Incubator workshops and will be further developed before being published as AMWA IS-05.

7.2 Media Device Control over IP (SMPTE ST 2071)

The Media Device Control (MDC) suite of standards addresses the atomic, low-level features needed to control media devices and services over the Internet Protocol, both in a deterministic, low-latency manner, and in a *laissez-faire* manner.

Signals are one-way, asynchronous communications between one or more nodes, used to notify one another of changes to their internal state. Signals always have a point of origin, known as the sender, and may have zero or more recipients, known as listeners. Signals are used to broadcast packets of information to interested parties in an asynchronous fashion, commonly referred to as messages or events.

Capability Interfaces are provided to represent the minimal behavior required to implement a concise, atomic feature, but the Capability Identity may also be used to designate a traditional monolithic service, API, or interface, allowing for the features exposed by a monolithic service, API, or interface to be delineated and independently accessed in an atomic fashion, just like interfaces that were designed to represent a Capability, aka Feature. They can provide access to attributes and operations. A Service is a software service that

exposes one or more Capability Interfaces. A Device is a software service that represents logical or physical hardware, typically run in firmware, or as a microservice. A Mode is an operational state that defines the Capabilities and behavior of a Device or a Service while it is in that mode of operation. Modes change the available capabilities, while allowing a base set of capabilities to exist; for example, Play mode, Record Mode. Media is audiovisual material and its ancillary information.

The Media Device Control (MDC) Framework (MCDF) defined in SMPTE ST 2071-1 uses an identity scheme that is based on the IETF Uniform Resource Name (URN). Directories, aka registries, are software services that facilitate the searching and lookup of systemic resources. Directories may represent resources as a doubly linked hierarchical tree, known as a resource hierarchy. Directories expose the operations necessary to lookup, list, and search for the resources they contain. They may also aggregate multiple child directories into a single, consolidated view, acting as a centralized view into a distributed repository, aka registry. SMPTE ST 2071 directories may be queried with a basic Boolean expression language that can easily be transposed to a SQL-like syntax, using functions such as And, Or, Less Than, Greater Than, Equals, Matches, Contains, and Not.

The Media Device Control Protocol (MDCP) is defined in SMPTE ST 2071-2, and defines the exchange of messages between nodes for the purpose of executing operations, broadcasting notification events, and exchanging data. It is implemented using the OASIS Basic Profile 1.2 web services specification, using the SOAP 1.1 HTTP protocol binding. Service definitions are represented using Web Service Definition Language version 1.1 (WSDL 1.1).

SMPTE ST 2071-3 Media Device Control Discovery (MDCD) describes the Zero Configuration (ZeroConf) and Service Discovery mechanisms defined for Media Device Control.

The service capability description framework of FIMS 2.1 incorporated the SMPTE ST 2071 service capabilities description interface in September, 2015. Since that release, the FIMS Repository, Transform, and Transfer media service capability interfaces have been tested and demonstrated using the SMPTE ST 2071 capability description.

The Media Device Control over IP suite of standards defines a comprehensive SASL compliant security layer and set of APIs.

7.3 AES70 Open Control Architecture

The Audio Engineering Society (AES) is the only professional society devoted exclusively to audio technology. In that role AES created the AES70 standard. AES70, the Open Control Architecture (OCA), defines a scalable, object-oriented control-protocol architecture for professional media networks. The standard is divided into three parts. Part 1 describes the models and mechanisms of the Open Control Architecture, which together form the AES70 Framework. Part 2 describes class structure, and Part 3 describes implementation with TCP/IP communications protocol.

AES70 supports high availability by offering:

- Device supervision of AES70 devices.
- Supervision of network connections to AES70 devices.
- Efficient network re-initialization following errors and configuration changes.

AES70 supports robustness by offering:

- A mechanism for operation confirmation.

- A mechanism for handling loss of control data.
- A mechanism for handling device failure of AES70 devices.
- Recommendations on network robustness mechanisms that network implementers may use.

The AES70 device model contains three categories of objects, as follows:

1) Managers. Manager objects shall be control objects that affect or report the basic attributes and overall states of the device.
2) Workers. Worker objects shall directly control the application functions of the device. Examples include audio mute switches, gain controls, equalizers, level sensors, overload sensors; video camera controls, signal properties, image processing parameters, and signal processing functions.
3) Agents. Agent objects shall provide indirect control of Workers within a device. An Agent shall not reflect a signal processing function, but instead may affect signal processing parameters in one or more associated Workers, or provide other application control functions. For example, an Agent named “OcaGrouper” implements complex grouping of control parameters in a manner that resembles VCA grouping in analog systems.

AES70 supports the following functions:

- Discovering the AES70 devices that are connected to the network.
- Defining and undefining media stream paths between devices.
- Controlling operating and configuration parameters of an AES70 device.
- Monitoring operating and configuration parameters of an AES70 device.
- For devices with reconfigurable signal processing and/or control capabilities, defining and managing configuration parameters.
- Upgrading software and firmware of controlled devices. Including features for fail-safe upgrades.

AES70 supports the following security measures for control and monitoring data:

- Entity authentication
- Prevention of eavesdropping
- Integrity protection
- Freshness, such that replayed messages in a replay attack on a protocol will be detected as such.

7.4 Networked Device Control Protocol (SMPTE RDD 38)

SMPTE RDD 38 Networked Device Control Protocol provides a lightweight and efficient control protocol specification to realize high speed control/response equivalent to using a conventional RS422/9-pin control device. It conforms to the specifications of MessagePack (an efficient binary serialization format) and MessagePack RPC (a cross-language remote procedure call library that uses MessagePack for object serialization), with some added definitions required for device control. These include an efficient method of message communication, and use of available transport protocols that take security into account.

SMPTE RDD 38 specifies messages for RPCs including “Request” and “Response”, as well as a “Notify” message. It does not define any method names for RPCs or Notifications.

7.5 Recommendation for Control Protocols

Several different standards and specifications for device control over IP are available today. There is some degree of overlap between them but also some areas where they complement each other. This diversity in standards and specifications can prove confusing to implementers and users. It can also cause unnecessary complexity which can be an obstacle for adoption. Harmonization across the different industry organizations is needed. The SG recognizes that some effort is underway in SMPTE to harmonize device control over IP standards and specifications. The SG recommends that SMPTE continues this effort and provides clarification on what the different standards and specifications apply to, what differences exist and where they complement each other. It also recommends that one or a minimal set of standards is being created for device control over IP.

7.6 Recommendation for SDN Control

The SG recommends that connection management is best achieved through SDN in consort with a controller application that can reserve bandwidth and establish reliable connections. While it is possible to manage flows without the use of SDN, care has to be taken over routing and bandwidth to ensure that network links are not overloaded; this can be difficult to achieve in larger scale networks.

For small system configurations (e.g. one switch/router), guaranteed transport QoS may be achieved without a network controller. However, the controller application interfacing with the network can help with security, monitoring and diagnostics.

8 Congestion Control

8.1 Background & Context

Early implementations of professional media networks are likely to be dedicated to real-time media flows, mostly uncompressed in nature. But the long-term vision is to utilize the same network for both real-time and non-real-time flows, as well as non-media flows, using some Quality of Service and/or traffic engineering mechanism to avoid any losses or delays on the real-time media flows.

As part of this effort, there is a desire to instrument the network to understand network load, congestion and latency. This document will discuss such instrumentation in the context of commercially available off-the-shelf Ethernet switching hardware..

8.1.1 Ethernet Switching Basics

Before discussing techniques to instrument network load and network congestion, it is important to first understand some basic architectural issues around Ethernet switching.

Ethernet is inherently a serial protocol. A packet is broken down into a stream of bits that are transmitted at wire rate. The packet consists of a packet header containing addressing information and context, and a payload. The time that it takes for an entire packet to be transmitted is called Serialization Delay. That delay depends on how big the packet is and the line rate of the particular Ethernet connection (e.g. 1Gb or 10Gb etc).

Ethernet switches generally operate in one of two modes:

- Store and Forward switching. A packet being received on a port is stored in a buffer until the entire packet has been received. Once the packet is fully received and verified (e.g. header, framing, length

and CRC is correct), a switching decision is then made and the packet is queued for transmission out of the correct egress port. Malformed or corrupted packets are usually dropped and not forwarded. The latency through the switch depends on a combination of the packet size (serialization delay) and the actual switch latency to make a switching decision and to start forwarding.

- Cut-through switching. A packet being received is stored in a FIFO buffer until just enough bits of the header are received in order to make a switching decision. Once a switching decision is made, transmission of the packet starts immediately to the correct egress port queue. Generally speaking only the section of the header containing the addressing bits are required to be received until the switching decision can be made and forwarding begins. However, if a packet is malformed or has an incorrect CRC, it will still be forwarded (since the malformed section might not have been received before forwarding started). Cut-through switching has much lower latency than store-and-forward because transmission starts without regard to overall serialization delay. It also has consistent latency, whereas store-and-forward switching latency varies because the serialization delay depends on the size of the packet.

Once a switching decision is made, the packet will start to queue for the output port. Such queuing may also involve QoS, security decisions and other data plane mechanisms. If the output port is free, transmission starts immediately. If another packet is already being transmitted out that port, this new packet is queued in an egress buffer. While there are variations in how queuing is handled depending on the silicon architecture of the switch, this high level description generally applies.

So depending upon the switching architecture, there are, generally speaking, ingress queues and egress queues, often segmented by QoS traffic class / priority. Such queues / buffers may be a fixed size per port / traffic class, or may be dynamic in nature whereby a larger buffer is dynamically carved and shared amongst many ports. The details of buffering architectures are complex but extremely important since they have a significant impact on how a switch handles congestion.

8.1.2 Network Congestion

Congestion in a switch is caused when multiple sources are trying to send traffic out of the same egress port. Congestion generally takes 3 forms:

1) Serialization congestion. Even if the aggregate sum bandwidth of the source streams are less than the bandwidth available on the egress port, it is possible that these sources are queuing for the egress port at the same time. So each packet must be transmitted before the next can be sent. This causes temporary buffering / congestion at the egress port, but buffer use will be minimal since the output can “keep up” with the overall aggregate input rate.
2) Microburst Congestion. When there are short lived and bursty traffic flows, it is possible to temporarily oversubscribe the bandwidth available at the egress port. This requires packets to be buffered. Assuming such oversubscription is short-lived, the packets are buffered and then transmitted in some logical order, and no traffic is lost. However, if buffers are exhausted, traffic may be dropped.
3) Sustained fan-in congestion. When there are real-time or long-lived flows that oversubscribe the egress bandwidth of a port, the switch will try to buffer such traffic, but eventually buffers will be exhausted and traffic will drop.

What is important to understand from the above description is that congestion may be very short lived. Thus the granularity and precision of instrumentation may or may not capture short-lived congestion.

8.2 Congestion Awareness and Instrumentation in a Packet-Based Professional Media Network

8.2.1 Lack of bandwidth awareness

The fact that existing multicast routing protocols like Protocol-Independent Multicast routing (PIM) or Link Aggregation Group (LAG) are not bandwidth aware creates several issues for deployments. The most obvious issue is the bit rate of uncompressed media flows. For a 1080i SMPTE ST 2022-6 stream, a single input generates a constant bit rate of 1.5Gb/s, 3Gb/s for 1080p. For 2160p “4K” UHDTV-1 formats the bit rate jumps to 12Gb/s. These rates create the potential for just a few streams to oversubscribe a single link. Likely there will be a maximum of 3 X 1080p streams per 10G link or a maximum of 3 X 4K streams per 40G link. This type of application necessitates leveraging multiple high bandwidth interconnects between network devices. Typically distribution of flows would be solved with Equal Cost Multi-Path routing (ECMP) or Link Aggregation Group (LAG) to scale to multiple interfaces across multiple devices. The lack of bandwidth awareness means that the existing hashing algorithms could place a high bandwidth flow on an already utilized link causing oversubscription based loss. PIM could be manipulated to use multiple paths by manipulating source routes but this is configured per hop, and would be manually or programmatically intensive yet still would not guarantee bandwidth aware stream placement across an L3 network.

8.2.2 Stream programming latency and scalability

In a scenario where a device would like to receive transmissions from a number of senders quickly (say a multiviewer or switcher) there could be the need for hundreds of streams that need to be programmed in parallel in milliseconds. PIM and IGMP typically take 10’s to 100’s of milliseconds to program new forwarding entries, and the performance is typically worse in parallel. Also, in many solutions a source address or source port switch is required which necessitates a fast “toggle” of inputs sending to the same destination group. Reprogramming a multiple hop network path for this type of switch cannot be done reliably with the PIM protocol in the 10-20 ms target required for these types of switchovers.

8.2.3 Lack of end to end topology awareness

In a multi-device L3 IP network, the current PIM protocol programs each source/multicast group (S,G) flow on a per hop basis, without knowledge of the end to end topology. If a path for a specific S,G through the network needs to be engineered outside of the unicast forwarding path, each PIM router must have its forwarding table manipulated manually or via an alternate routing protocol. Due to the requirement of bandwidth awareness, the lack of end to end topology awareness means an external controller or application must evaluate each device in the system to influence the PIM or unicast protocol.

8.2.4 Assumption of a *,G forwarding model

Both PIM and IGMP default to an Any-Source Multicast (ASM) forwarding model in which forwarding of packets to a particular destination is based only on the multicast group (G) to which the packets are sent (*,G). This matches very poorly to the media application profile that specifically defines each source. Also the networking industry has an aversion to any type of flooding, intentionally programmed or not. Due to the bit rates, sources must be implicitly enforced for both PIM and IGMP to ensure that traffic sent to the same destination group is only received for the source address requested and any software based registration to the rendezvous point (RP) is unacceptable. In the Source-specific Multicast (SSM) model, forwarding of packets to a particular destination is based both on the IP address of the source (S) and the multicast group (G) to which the packets are sent (S,G). For PIM Sparse Mode (PIM-SM) the use of SSM can reduce the risks of the *,G forwarding model and there is no requirement for RPs. For IGMP snooping programming there is an unresolved issue as most IGMP snooping tables forward based on [group,VLAN, port] and don't

take into account the source even when explicitly defined in an IGMPv3 join. Hence, with IGMPv3 and SSM a receiver on the same VLAN as multiple senders using the same group may receive all traffic for the group regardless of SSM join state.

It should be noted that Multicast Listener Discovery (MLD) is a component of the Internet Protocol Version 6 (IPv6) and used by IPv6 routers for discovering multicast listeners on a directly attached link, similar to IGMP.

8.2.5 Data Path Redundancy

SMPTE ST 2022-7:2013 "Seamless Protection Switching of SMPTE ST 2022 IP Datagrams" defines a redundant RTP stream transmission method that provides complete content redundancy. It does not define the Ethernet or IP layer requirements so this could be implemented as 2 identical S,G streams forwarded over diverse paths. There is also the potential for a single transmitter interface to have network "assisted" redundancy with the first network device replicating the stream to two diverse paths. With traditional PIM forwarding only one path for any given S,G stream can be active so any network level redundancy for a duplicate stream would require an alternate programming model.

8.2.6 Deployments with non-blocking network elements

In small deployments with a few endpoints connected to a single non-blocking network element, many of these issues can be mitigated. Traditional IGMP, Access Control Lists (ACLs), and network protocols can be manipulated to overcome some of the above limitations. The non-blocking "any to any" bandwidth of most modern silicon mimics the traditional crosspoint SDI behavior and hence removes the requirement for bandwidth awareness across the network fabric. Even larger scale non-blocking elements that do not use Ethernet frame based hashing algorithms in their internal fabric may also provide alternate multicast programming models that remove the limitations that PIM and IGMP may have. Unfortunately deploying a "bigger box" is typically not the acceptable way to scale deployments and may not be feasible or desirable.

8.2.7 Congestion and PMN

Within the context of a Professional Media Network, the desire to instrument network load and congestion is to address several needs:

1) In the case of an SDN network with a bandwidth-aware controller that is dedicated to constant bit rate media flows, sustained fan-in congestion should never take place. However, it is useful to have instrumentation to verify that all is operating as expected and detect any potential traffic loss caused by design flaws in the SDN controller logic. In addition, the allowance of bursty flows can cause packets to be buffered and, in some cases, dropped (this is the motivation for the Network Compatibility Model of SMPTE ST 2110-21).

2) In the case of networks handling compressed media flows, actual flow bandwidth may vary due to compression algorithms. In such cases it is necessary to monitor congestion since any SDN type of control will be unable to accurately predict precise flow bandwidth requirements.

3) If the PMN relies (at least partially) on traditional Ethernet techniques to handle dynamic traffic steering such as PIM for Multicast, it is impossible to know the actual route that multicast traffic will take over a multi-switch network and congestion management may be necessary to avoid oversubscription and traffic loss.

4) If the broadcast network is also shared with non-real-time and non-broadcast related traffic (using QoS mechanisms), it is necessary to monitor congestion and traffic loss to understand network design and perform network evolution / traffic engineering to adequately address business needs.

8.2.8 Common Instrumentation

In the world of off-the-shelf Ethernet switching, there are several common methods available to instrument network loading and network congestion:

- Interface counters. There are hardware-based statistical counters to detect many aspects of the data plane including ingress and egress packet drops. Packet drops would occur if buffers are exhausted, depending upon the switching architecture.
- Interface traffic rates. Most silicon can measure an overall port loading. However, such measurements are averages taken over a window of time and will not detect short-term bursts that exceed available bandwidth.
- Sflow / Netflow. Some form of sampling and mirroring can be used to measure actual link usage. However, statistical sampling may not capture short-lived bursting.
- Correlation Tools. Some switches have the ability to feed network information to a monitoring and correlation platform such as Splunk. This is useful to get a big-picture view of congestion and loading, but since the granularity of information will depend on the particular switch implementation, it may not highlight short-lived oversubscription.
- SNMP. Simple Network Management Protocol (SNMP) can be used to monitor link usage, packet drops etc. However, this is a polled environment and is most useful to detect drops, not dynamically understand congestion.
- Port mirroring. All traffic can be mirrored to performance monitoring tools that can measure congestion. However, it is not cost effective to capture and analyze a large number of high-speed links.
- Some switching architectures (less common) have special real-time instrumentation to understand buffer usage and detect congestion. By definition, congestion will cause buffers to fill. Thus real-time detection and analysis of buffer use is highly valuable to truly understand network congestion. However, it is important to understand the granularity of such buffer instrumentation. Some vendors measure buffer use over a relatively long window of time that can miss microburst events.

8.2.9 Considerations

When instrumenting a PMN for understand loading and congestion, the following aspects should be carefully considered:

- Nature of congestion. Does the network only have real-time uncompressed flows, or are there bursty and short-lived flows as well?
- Does the network utilize QoS mechanisms?
- What are the goals of congestion instrumentation? To detect drops? To understand loading so that network engineering can be undertaken to avoid any drops? To verify QoS mechanisms? To troubleshoot application problems? To police an SDN controller architecture?
- What granularity is required for the intended purpose?
- Will available instrumentation deliver the required granularity of measurements or are “averages” utilized that will not capture the desired data? It is critical to understand the low-level implementation details of the specific instrumentation being utilized.
- What special switch capabilities might exist to provide finer-grained measurements?
- Do you need an overall “heat map” of network loading and congestion, or are you only interested in point measurements for individual links / switches?

- Do you use dynamic traffic steering or only SDN approaches? Will you be fully aware or in control of where flows are being steered in a multi-switch network? This is especially relevant for multicast traffic.
- How can mirroring / Sflow / Netflow be used to measure network loading? What budget / cost implications have been factored into the overall project plan?

8.3 Network Redundancy

SMPTE ST 2022-7 defines requirements for redundant streams of RTP packets (including SMPTE ST 2022-6 SDI over IP) to allow for creation of a single reconstructed output stream through seamless protection switching at the RTP datagram level. The transmitter sends at least two streams, each containing copies of each RTP datagram. The RTP header and the RTP payload are identical for each datagram copy. The seamless reconstruction method makes no assumptions about the Ethernet or IP headers of the source streams.

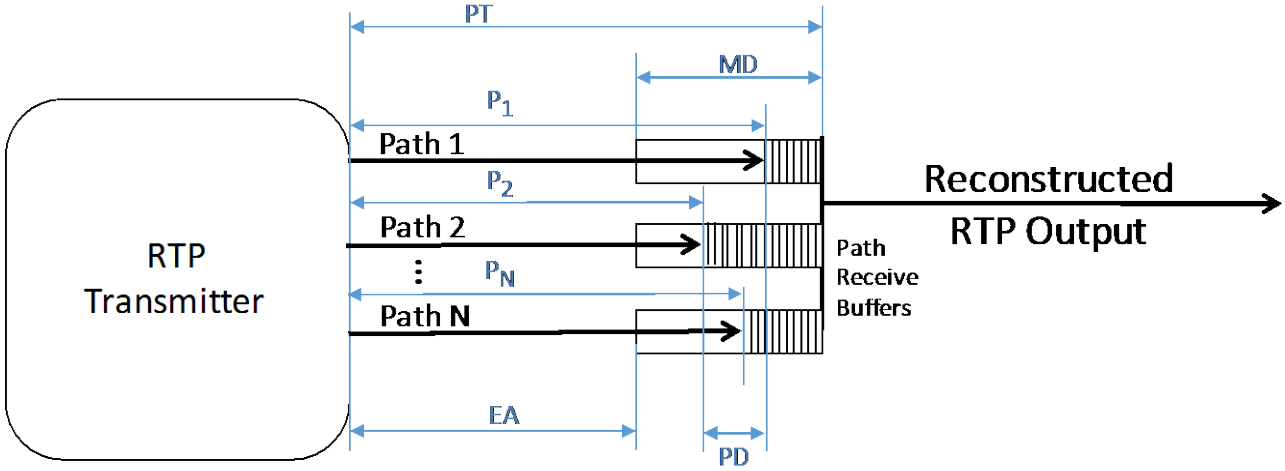

Figure 6 Buffering Details of SMPTE ST 2022-7

The Maximum Differential (MD) is the latency from transmission to reconstructed output (PT) minus the earliest time that a packet can arrive at the receiver to be part of the reconstructed output (EA).

P_1 is the instantaneous latency from transmission to reception of datagrams on path 1, and P_N is the similar latency on path 2. PD is the instantaneous path differential, and is always equal to the maximum of the absolute value of the differences between instantaneous latencies. After startup, the instantaneous latencies of the paths may change due to changes in network routing and latency, but only to the extent that PD remains within the bounds. SMPTE ST 2022-7 describes PD maximums for low, moderate, and high skew classes.

As long as path latencies are greater than EA and less than PT, then seamless reconstruction is able to recover packet losses and create a successful output stream.

SMPTE ST 2022-7 could be used to provide seamless “dual chain” redundancy in a broadcast plant. It could solve problems with both random packet loss and well as burst packet loss or failure of a complete chain. Note that SMPTE ST 2022-7 can only solve problems of packet loss, it cannot provide redundancy for failures due to media processing that do not cause packet loss. Imagine if a logo inserter continues to emit packets, but the video is all black due to a software bug. This kind of problem will not be solved by SMPTE ST 2022-7 redundancy, but would require deeper inspection by a video QA system.

8.4 QoS Requirements in PMN

8.4.1 Introduction

As fixed facilities and mobile production facilities transition to IP-based infrastructure, these Professional Media Networks (PMN) will need to support a mix of different packetized traffic types including real-time A/V flows (e.g. SMPTE ST 2022-6, SMPTE ST 2110-x, AES67), intercom audio, IEEE 1588v2 PTP messaging, near-real-time file transfers for playout systems, and other media flow and traffic types. These flow types will coexist inside the network, and each needs a defined quality of service (QoS) level based on the specific "technical and business needs" of the flow. For example, real-time A/V QoS should have a specified data rate, be lossless and with low latency.

Figure 1 is a drill-down of what is needed to define QoS for different flows. To ensure that an authorized flow gets the necessary QoS, there are three key requirements:

1) The network should be well designed and provisioned with sufficient resources so the flow requests will be accepted.
2) There is an admission control process that admits the flows and authorizes the use of network resources for the flow.
3) Finally, when a flow arrives, it is permitted at the admitted rate and with the right QoS markings for the required service class.

These three are described in **Figure 7**.

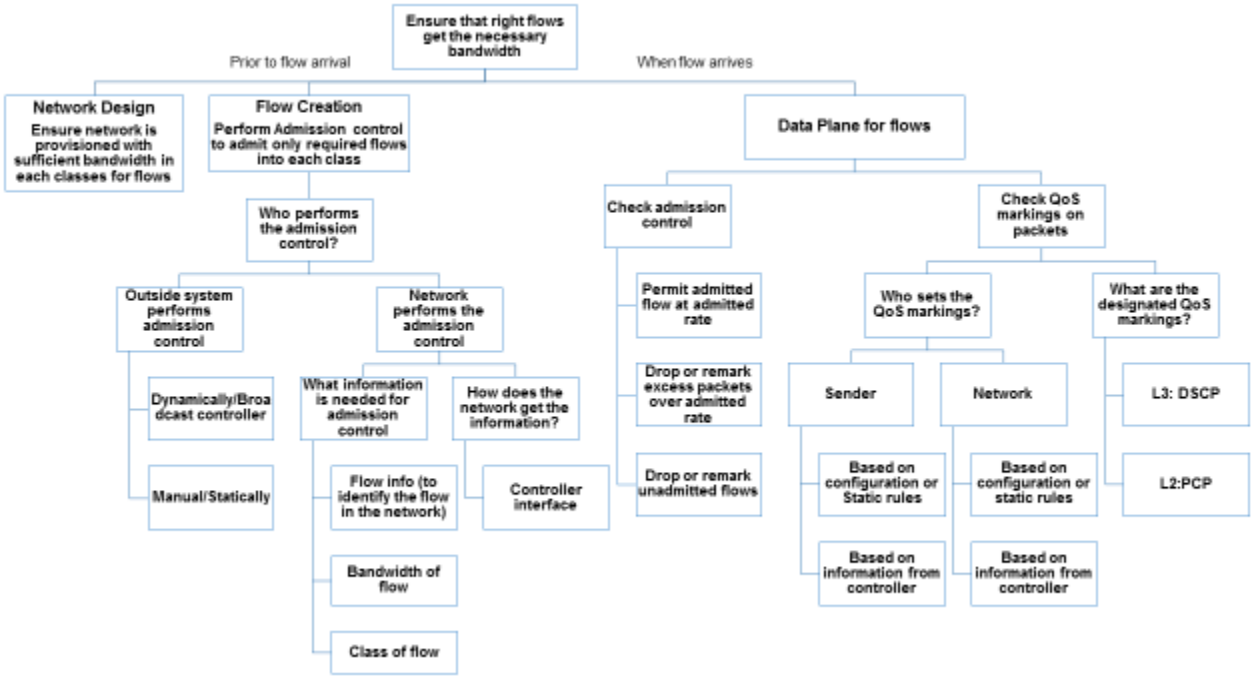

Figure 7: The Elements of Network QoS

8.4.2 An Example QoS Profiles

The first and foremost requirement is a well-designed network. Only with sufficient capacity, can the loss, latency and jitter requirements be delivered. This is different from over-provisioning the network which is a brute force tactic and wasteful of resources.

A well-designed network requires each flow to be identified with the appropriate QoS markings. The network uses the packet's QoS markings to determine the service class being requested by the sender or traffic forwarding mechanism. As per IETF RFC 4594, a service class represents a set of traffic that requires specific delay, loss, and jitter characteristics from the network. The network traffic is usually partitioned into multiple service classes. Each service class provides a different forwarding priority within the switching fabric, which is used to engineer the right behaviors for a successful system. The forwarding behavior allows attributes such as bandwidth, latency, jitter and packet loss to be managed through proper network design and traffic engineering.

In order for per-flow QoS markings to be effective, they must be supported in every switch and router along the path. Note that there are QoS flow markings in layer 3 (L3) (the Differentiated Services Code Point (DSCP) field (IETF RFC 2474 and RFC 4594)) and also in layer 2 (L2) (Ethernet header Priority Code Point (PCP) field (IEEE 802.1Q class of service)).

Table 1 shows an example QoS profile. The DSCP and PCP values are indicative of values that might be used but should not be taken as definitive recommendations

Generally, network devices have multiple (often 8) hardware queues, and therefore each of these classes of traffic are internally mapped into these different hardware queues. The actual dynamics of the hardware queues differ from platform-to-platform and also differ based on specific configuration in the switch. Often there are one or two "strict priority" queues, which will always get scheduled for transmission ahead of others in the egress interface. The classes of traffic that require this type of strict scheduling are marked as *Priority* in the *Queue* column in the example QoS profile of Table 1. The remaining queues are often scheduled on some kind of weighted-round-robin (WRR) basis, in order to avoid starvation or excessive delays of the lowest queues. This type of scheduling is referred to as rate-based scheduling and is marked as *rate* in the table below.

Service Class (Flow/Content)	L3 DSCP	L2 PCP	Constant Planned Static Bandwidth	Queue	Notes
PTP time and sync	EF	5	No	Priority	It is not recommended to use CS7 (56). Please see IETF RFC 4594 section 3.1 for additional details.
Intercom & Intra-mix audio	EF	5	yes	Priority	latency-critical interactive
AES67, TR-03 Audio	AF41	4	yes	Please See note below	AES67 specifies AF41
SMPTE ST 2022-6, TR-03 Video	AF41	4	yes	Please See note below	
TR-03 Ancillary Data	CS5	5	No	rate	Suggest not carrying audio as ancillary data
Real-Time control traffic	AF21	2	No	rate	TCP socket, operator-interactive or machine-to-machine
Storage Traffic for Recording/Playout (quasi-real-time)	AF31	3	Yes	rate	Ingest and Playout servers attached to network storage
Storage Traffic for File transfers and offline transcodes (non-real-time)	AF11	1	No	rate	High bandwidth, bursty due to TCP model, may be delivery-time critical
Operations, Administration, and Management (OAM) traffic	CS2	2	No	rate	
Email and other low priority traffic	DF	0	No	rate	All best effort traffic

Table 1: Example Flow Types and QoS Markings

If there are fewer than 8 queues, then multiple service classes may have to be mapped into a single queue which can negatively impact the forwarding of the packets in the queue. However, in some cases, less than 8 queues may suffice depending upon the types of traffic in the network and the QoS required for them.

The network can verify the markings (or even over-write them) based on information received from either upstream control systems or static configuration. This is done to prevent the misuse of these markings, either inadvertently or deliberately, that can cause network resources to be oversubscribed. The network can also

mark the traffic flow on behalf of the sender if the sender is incapable of marking the packets with the QoS values.

Scheduling and Bandwidth Management are the primary purpose of QoS classifications of flows. In current practice, many applications manage bandwidth through static design of the system and its topology - the maximum bandwidth requirements of senders and receivers are analyzed and the network equipment and links are designed and provisioned to enable all of the planned traffic interactions to simultaneously occur. This not only results in the over-provisioning or over-engineering of the network resources, but also cannot scale to large and dynamic systems.

Software defined networking (SDN) and other higher-level network control methodologies offer the capability to manage bandwidth dynamically -- providing admission control (to guarantee that only known flows enter the network) and also providing ingress rate policing to ensure that flows do not exceed the planned rates. In addition, dynamic bandwidth planning can be accomplished within these higher-level control methodologies, in order to more fully utilize the network resources and/or prevent the need to over-engineer resources and capacity. While over-engineering is not required, a well-provisioned network design is still a requirement to deliver the loss levels, latency and jitter metrics required by the application.

8.4.3 Recommendation on QoS Profile

SMPTE should develop a Standard or Recommended Practice regarding a PMN QoS profile.

8.4.4 QoS and Network Reliability

When considering systemic reliability, network traffic engineering is only one part of the solution. In addition to flow QoS markings, additional design strategies such as network connection redundancy and high availability (HA) design are required to support the overall availability of the service. For example, in an SDN model, active-active network controller redundancy should be considered.

8.5 Tools for Bandwidth, Loss, Latency and Jitter Measurement

There are two common mechanisms for measurement of bandwidth, loss, latency, and jitter on networks. The first is an active probing of the network by introducing test traffic and measuring the desired traffic statistics. Iperf (https://github.com/esnet/iperf) is an example of such a tool. Artificial test traffic may not perfectly reflect the characteristics of production traffic, and also may interfere with desired traffic in a network that is in production.

The second method depends on collecting statistics and performing analysis on them. Network devices can be interrogated with Simple Network Management Protocol (SNMP) queries to look at bandwidth, packet loss known to the network device, and in some cases network device queue lengths. The precise SNMP queries required may differ by device, and also there are often performance limitations on how fast network devices can be polled by SNMP. The Multi Router Traffic Grapher (MRTG, http://oss.oetiker.ch/mrtg/) is a common free software application for collecting and organizing SNMP data.

sFlow (http://www.sflow.org), short for "sampled flow", is an industry standard for exporting truncated packets at Layer 2, together with interface counters. sFlow uses sampling to achieve scalability, which makes it more applicable to high speed networks. Sampled and truncated packets as well as sampled interface counters are sent from sFlow “agent” devices to an sFlow “collector” for analysis. There are many different sFlow collectors available with different capabilities that have been developed for different use cases. SFlow can be used for detecting flows across the network and measuring their bandwidth, as well as measuring counters of

packets known to be dropped by network devices. However it is not as useful for detecting random packet loss and performing latency and jitter measurement.

In-band Network Telemetry (INT) is a new framework to allow the collection and reporting of network state by the data plane, without requiring intervention or work by the control plane. In the INT architectural model, packets contain header fields that are interpreted as “telemetry instructions” by network devices. These instructions tell an INT-capable device what state to collect and write into the packet as it transits the network. INT traffic sources can embed the instructions either in normal data packets or in special probe packets. INT traffic sinks retrieve the collected results of these instructions, allowing them to monitor the exact data plane state that the packets “observed” while being forwarded. Network state recorded in the packets can include switch and port IDs, link utilization, and latency of the paths the packet takes through each hop it takes through the network. To date, INT has generally been associated with programmable data plane network devices.

Of further interest is Anomaly Detection, a data mining technique that detects patterns which do not conform to the expected normal operation from analyzing the logs. This technique can be used to monitor the health of a complex system, help predict failures before they happen and identify successful or attempted security breaches.

8.6 Recommendation for Congestion Measurement

Many tools are available which can be used for congestion measurement. Some of these tools are proprietary, some are open specifications and some are standardized. The IT industry has used these for a long time to detect or avoid congestion in IT networks. That said, there are currently no standards available specifically designed for congestion measurement in PMN. The SG recommends that SMPTE undertakes an effort to create one standard for congestion measurement in PMN to reduce complexity and increase interoperability between flow management and network devices.

8.7 Recommendation for Anomaly Detection

Anomaly detection technique for the professional media infrastructure is new and developing. Further discussion on this topic is recommended.

9 Security in Flow Management

There are several aspects to security in flow management, such as controlling access to particular flows, protection against inadvertent disruption and protection against malicious disruption.

Due to several recent high-profile cyber attacks on media companies cyber security became a very important topic. There are a multitude of reason for cyber attacks and they can vary from politically motivated, financial gains to just doing it for “fun”. They can come from organized crime, terrorists, foreign states, individual hackers or insiders. Employees can wittingly or unwittingly aid those attackers by exposing user credentials. In fact, most of the cyber attacks (90%)[1] were successful because of stolen user credentials unwittingly given

[1] http://www.prnewswire.com/news-releases/employee-errors-cause-most-data-breach-incidents-in-cyber-attacks-300342879.html

away by users through e.g. phishing attacks. The thread of cyber attacks have been further increased by a trend to use consumer technology in professional media infrastructure.

Most importantly these cyber attacks can create substantial damages if security is neglected and no countermeasures are applied. Malicious parties could take control over or destroy equipment, covertly divert content streams, publicly disclose private information or generally disrupting production.

Discussion of most of the countermeasures, i.e. intrusion protection, are not in scope for this group as they do not directly apply to flow management but to the overall network architecture or how network devices and software are constructed. Furthermore, the main reason for successful cyber attacks, the unwittingly disclosure of access credentials, needs to be address through proper staff training which again is not in scope of this group.

There are countermeasures which are applicable to the flow management discussion. The Joint Taskforce on Networked Media (JT-NM) conducted an exhaustive gathering of requirements regarding network media in phase 1 of their effort. It identified several user stories concerning security in networked professional media and the Study Group on Flow Management in PMN reviewed these user stories and determined that some of them apply to flow management (see Annex D). Additionally, a survey among users of network equipment was conducted during drafting of this report and revealed a need for some form of security measures in the flow management process.

The need for authentication and the possible encryption of flow management commands was specifically raised in survey responses. Authentication can prevent malicious third parties from accessing vital systems in the production of content as long as login credentials are not compromised. Authentication is common in consumer applications and open as well as proprietary solutions are available. Single-factor and increasingly two-factor authentication are part any consumer online experiences. Multi-factor and strong authentication for very sensitive systems is available as well. There is a need to investigate if these solutions are applicable to professional media network and if a consensus can be reached in the professional media community to utilize just one of these methods as common standard. This would ensure interoperability among different products from different vendors.

Encryption can prevent “Man in the Middle” attacks aimed at capturing user credentials, identifying and possibly altering flow management commands or distributing production with malicious data. Encryption needs to be on-the-fly and with minimal or no delay to not negatively affect the behavior of a control system. Similar to authentication, encryption methods are available but not necessarily included in or tailored to flow management protocols.

9.1 Recommendation for Authentication in PMN

No standard for user authentication in professional media networks exist today. SMPTE should undertake an effort to study user authentication in professional media networks, investigate if available solutions are applicable for professional media solutions and determine of a standard or recommended practices for such should be created.

It should be noted that security mechanisms are not an immediate concern for most users. Most IP based infrastructure will be implemented as island solutions utilizing a walled garden approach separating the IP network completely from the outside world.

Also besides external attacks, there are other factors that could compromise security, such as user error.

10 Recommendations

10.1 User Survey Recommendations

The SMPTE Study Group on Flow Management user survey asked users to provide feedback on additional standardization work to be addressed in SMPTE or other organizations:

“Please highlight any opportunities you feel SMPTE should consider for standardization work - what aspects of the IP ecosystem do you feel should be standardized by SMPTE, which should be standardized by other relevant organizations, and what should remain vendor-specific?”

The submitted recommendations are listed below. They were discussed in the Study Group and notes from this discussion were added to each recommendation.

Recommendations for SMPTE Work

1. Link quality and open error correction: SMPTE should look at more efficient FEC for compressed and uncompressed workflows.

   Note: The SG agreed that this is out of scope for the flow management discussion as it concerns the transport mechanism. This effort would need an industry leader for standardization.

2. The overall education of vendors building IP devices needs to improve dramatically. There are some basic fundamentals about how multicast and IP packet timing work that are ‘off the radar’ for many vendors who are unaware.

   Note: The SG agreed that this is out of scope for the flow management discussion.

3. Definition of how to include signal components required by law, such as captioning language and descriptive service identifications.

   Note: The SG agreed that this is out of scope for the flow management discussion.

4. Timecode or frame stamping since frame rates are still archaic with 30/1001.

   Note: The SG agreed that this is out of scope for the flow management discussion. This is being addressed in SMPTE Time Labeling effort. It does have some relevance in switching of streams.

5. Possibly UID bank for company registration to help identify signal sources and destinations.

   Note: The SG agreed that this is out of scope for the flow management discussion. There is work going on in other organizations. It is an important effort for transport

6. There should be a standardized SDN protocol for control of switching systems. We have lived for far too long with numerous router control protocols that all essentially do the same thing.

   Note: The SG agreed that this is being addressed.

7. There presently seems to be an opportunity to standardize on endpoint device control across vendors for flow subscription. It is unclear whether that standardization would be best served by SMPTE, another organization, or between vendor-partners.

   Note: The SG agreed that this is being addressed.

8. Encapsulation should not be vendor-specific. There should be a standard API to send commands to a broadcast device.

Note: The SG agreed that this is being addressed.

9. SMPTE should develop a Standard or Recommended Practice regarding a PMN QoS profile.

What SMPTE should not do:

- Accurate switching mechanism, among others, should remain vendor-specific.

  Note: The SG agreed that a standard or at least recommended way of frame accurate switching for interoperability should be discussed.

- Error concealment should also be vendor-specific.

  Note: The SG agreed that this is out of scope for the flow management discussion.

Additional Industry Recommendations

Validation of devices as being SMPTE ST 2110 compliant would be helpful as the interop of IP devices has much greater packet delivery timing variability compared to deterministic SDI signals.

Note: The SG agreed that this is out of scope for the flow management discussion. While the SG agrees that there is a need to address this, other organizations are working on it.

10.2 Recommendations for SMPTE work

Below are recommendations from the SG on Flow Management in PMN in regards to new or ongoing work in SMPTE.

10.2.1 Recommendation for Control Protocols

Several different standards and specifications for device control over IP are available today. There is some degree of overlap between them but also some areas where they complement each other. This diversity in standards and specifications can prove confusing to implementers and users. It can also cause unnecessary complexity which can be an obstacle for adoption. Harmonization across the different industry organizations is needed. The SG recognizes that some effort is underway in SMPTE to harmonize device control over IP standards and specifications. The SG recommends that SMPTE continues this effort and provides clarification on what the different standards and specifications apply to, what differences exist and where they complement each other.

10.2.2 Recommendation on QoS Profile

SMPTE should develop a Standard or Recommended Practice regarding a PMN QoS profile.

10.2.3 Recommendation for Congestion Measurement

Many tools are available which can be used for congestion measurement. Some of these tools are proprietary, some are open specifications and some are standardized. The IT industry has used these for a long time to detect or avoid congestion in IT networks. That said, there are currently no standards available specifically designed for congestion measurement in PMN. The SG recommends that SMPTE undertakes an effort to create one standard for congestion measurement in PMN to reduce complexity and increase interoperability between flow management and network devices.

10.2.4 Recommendation for Authentication in PMN

No standard for user authentication in professional media networks exist today. SMPTE should undertake an effort to study user authentication in professional media networks, investigate if available solutions are

applicable for professional media solutions and determine of a standard or recommended practices for such should be created.

10.3 Other Recommendations

Below are several recommendations from the SG on Flow Management in PMN which do not directly concern new or ongoing work in SMPTE but rather the media the media industry in general.

10.3.1 Recommendation for Methods of Flow Switching

The SG recognizes that destination-timed switching using IGMP and switch-timed switching using SDN methods are the most popular forms of flow switching. But it also recognizes that it is not advisable to standardize one specific flow switching method at this point in time. Technology for the different switching methods and the implementation of IP infrastructure are still evolving. The SG recommends that determining the best flow switching method should be left to users, equipment manufacturers or system integrators.

10.3.2 Recommendation for SDN Control

The SG recommends that connection management is best achieved through SDN in consort with a controller application that can reserve bandwidth and establish reliable connections. While it is possible to manage flows without the use of SDN, care has to be taken over routing and bandwidth to ensure that network links are not overloaded; this can be difficult to achieve in larger scale networks.

For small system configurations (e.g. one switch/router), guaranteed transport QoS may be achieved without a network controller. However, the controller application interfacing with the network can help with security, monitoring and diagnostics.

10.3.3 Recommendation on PIM

For effective management of flows in a PMN the Study Group members recommend that PIM should not be relied upon.

10.3.4 Recommendation for Anomaly Detection

Anomaly detection technique for the professional media infrastructure is new and developing. Further discussion on this topic is recommended.

11 Bibliography

AMWA IS-04 NMOS Discovery and Registration
Available at: https://www.amwa.tv/projects/IS-04.shtml

AMWA IS-05: NMOS Connection Management
Available at: https://www.amwa.tv/projects/IS-05.shtml

JT-NM. (2015). *Reference Architecture v1.0.*
Available at: http://www.jt-nm.org/RA-1.0/JT-NMReferenceArchitecturev1.0 150904 FINAL.pdf

SMPTE. (2014). *Report of the SMPTE Study Group on Media Production System Network Architecture.*
Available at: https://www.smpte.org/standards/reports

J. Bilberry, M. Palmer, and R. Sullivan, “Network Service Security Through Software-Defined Networking,” 2013, http://institute.lanl.gov/isti/summer-school/cluster-student- projects/network-service-security-through-software-defined-networking.

J. C. Mogul, J. Tourrilhes, P. Yalagandula, P. Sharma, A. R. Curtis, and S. Banerjee, “Devoflow: Cost-Effective Flow Management for High Performance Enterprise Networks,” Proc. 9th ACM SIGCOMM Workshop on Hot Topics in Networks, Hotnets-IX, Monterey, CA, 1:1–1:6, 2010.

P. Bosshart, D. Daly, G. Gibb, M. Izzard, N. McKeown, J. Rexford, C. Schlesinger, D. Talayco, A. Vahdat, G. Varghese, and D. Walker. P4: Programming Protocol-independent Packet Processors. SIGCOMM Comput. Commun. Rev., 44(3):87–95, July 2014.

Annex A User Questionnaire

The SG conducted a survey to collect information from users on requirements for flow control in professional media networks. All interested parties were invited to respond to this questionnaire, including those who are not members of SMPTE. The information collected in this survey is presented in a summarized fashion below.

1) What are your requirements for switching of media flows in terms of the following?

Please consider both general routing (e.g. sending a particular source to a monitor) and production switching (cutting between sources to make broadcast output).

a) Seamless switch (no glitch on output)

The need for seamless switching depends on the application. General routing does not require a seamless switch but it is required for production switching. Most accept one black or repeated frame in production switching. Audio should have a simple cut only or a fade (V or X) when switched. Multiple black or repeated frames or multiple repeated audio samples are not acceptable. Latency should be between 1 to 2 frames. The capability to switch off seamless switching is desired if it lowers latency.

b) Switch at defined position in flow (e.g. at a particular time code or at same frame operator was viewing when he/she hit switch button)

Most responses stated that switching at a defined position in a flow is highly desirable for some applications (i.e. automation system, production video mixer). For these applications frame accurate switching is required (< 1 frame). In scheduled playout scenario ±1 frame is a problem. Human controlled switching for general routing can be somewhat more relaxed (< 750ms). Latency to achieve frame accurate switching should be minimal (within a frame) and consistent. Some expect the end-point device to handle timing issues based on PTP time stamps.

Note: The SG think's that the above value of <750ms for switching in general routing is generous. A more appropriate number would be in the range of <100ms. Latency below <100ms can be achieved in some equipment today. The above number of <750ms may be acceptable for setup of routes.

c) Latency of switched output with respect to

i) original signals (transit latency through the system)

Expectation regarding transit latency are application dependent. Camera to Studio Output latency can should be as little as 1 to 2 frames or as much as 4 to 5 frames. Audio latency expectations for this application should be around less than 5ms. Production switching requires latency of 2 to 3 lines or up to 10 lines. Non-live applications can have up to 500ms latency. Any latency should be constant.

Note: The SG believes that participants in the survey may have misunderstood the questions as latency from camera to studio output while the intent of the questions was regarding packet level latency within a switch fabric.

ii) Timing of the command (control latency)

Survey participants stated there should be a rapid feedback in human operated control systems, about 50ms or less. Any latency for control commands greater than 100ms for operator actions is

problematic. There can be application where latency for control commands of up to 750ms is acceptable.

Note: The SG considers that the above value of <750ms for control command feedback is generous while 50ms may be challenging for some applications. <750ms may be acceptable for setup routes.

d) Switching related flows (audio, video, data) separately or co-timed with each other

Some participants stated that co-timed switching is not required today. Some stated that both separate and co-timed switching will be required depending on the application. This might require the switching of more than 100 flows at the same time. Some require multiple streams of multiple essence types to be switched together. In playout and master control switching should be frame accurate. Some require flows to be switched within 50ms of each other. Some require 50µs. Audio and video sync must be maintained. Time stamping flows to establish sync is required.

Note: The SG believes that 50µs may be unrealistic.

2) What scale of network do you require to route media flows over?

a) logical scale (e.g. number of flows, number of interfaces)

Users want scalable systems to virtually unlimited number of media flows. There are requests for 1x1 video (as SDI, point to point), 10x10 and 4Kx4K. One user asked for 10K non-blocking flows. Another asked for 2K intercom connections. Up to 4K audio flows was requested. Point-to-multipoint was requested as with SDI.

b) physical scale (e.g. local, intra-facility, inter-facility, internet)

Intra-facility was most commonly cited but all were important to some extent. For inter-facility, private line and dark fiber preferred. Some cited internet transport as a use case while others found no use for it.

3) What compressed real-time media flows (e.g. Dolby E, DTS audio, JPEG 2000, MPEG-2, H.264) do you use in your facility today?

Regarding video codecs, every respondent mentioned the use of MPEG-2 and H.264, and some specifically called out AVC-I. Nearly half of respondents mentioned JPEG-2000, and a third of respondents mentioned HEVC. One respondent mentioned LLVC, and one mentioned VC-2.

Regarding audio codecs, a third of respondents mentioned the use of Dolby E and AC-3. One respondent mentioned DTS, and another mentioned FLAC.

One respondent said that except for CATV, "Compressed flows are not in consideration at the moment, not even for higher resolutions (4K or beyond)." However another respondent suggested "Splicing Insertion of Captions into MPEG2 TS is another example of compressed applications. We believe in retaining the compression of a source unless it absolutely merits decoding. With unwrapping/rewrapping of transport codecs to file based ones, the requirement to always decode first isn't there."

Regarding requirements for PMN systems, one respondent said "The important thing is that networked media systems are codec-agnostic."

4) What formats do you envisage using for real-time media flows over IP? (e.g. RTP, HTTP, DASH; multiple/single essence streams per flow; SMPTE ST 2022 family, AES67)?

Nearly half of respondents mentioned SMPTE ST 2022 formats, although only one specifically called out SMPTE ST 2022-6, and another called out SMPTE ST 2022-7 (not actually a media format) use. Also half of respondents mentioned AES67.

Several respondents discussed the use of single essence streams per flow versus multiple essence streams per flow, however it was unclear if the precise definition was the same between all users. For example, different users identified both SMPTE ST 2022-6 and SMPTE ST 2110 as "single essence streams per flow."

Note: The study group believes that there is some confusion among users about single and multiple essence streams per flow.

SMPTE ST 2022-6 appeared to come in for some criticism. One respondent said "SMPTE ST 2022-6 is of little use particularly in WAN workflows due to its failure to control the spacing of packets to avoid bursts and no FEC." Another suggested that they required "more [audio flows] than SDI and SMPTE ST 2022-6 can handle". One respondent was more balanced and suggested "Single essence streams per flow (SMPTE ST 2022 family) would suffice, but multiple essence streams per flow (Aspen, TR-03/04 + AES67 family) is much more desirable."

About 20% of respondents mentioned SMPTE ST 2110 and a similar amount mentioned VSF TR-03.

One respondent also mentioned RTMP, DASH, HLS, and F4M (Adobe HTTP dynamic streaming).

5) What requirements do you have for discovering sources and destinations for media flows?

One respondent stated that Inter-operability between manufacturers is very important. Another stated that Registration and Discovery of source and destination devices should be a standard, including Authentication. Discovery requires strong security to have any value. Unauthorized use must be prevented.

One respondent suggested that discovery of logical sources, logical flows, senders and receivers must be dynamic, automatic and scalable. The discovery must be agnostic to underlying protocols.

Other respondents had variations. Where “auto-discoveries” of devices are utilized, they should be manually vetted and "quarantined" until activated. Manual configuration of databases should be maintained, which can be quickly and easily updated. An interface should be provided to manage them. Discovery of a device should include make/model, capabilities, any provisioning of physical interfaces, modes (e.g. active, inactive or redundancy/failover), etc. Naming of a source or destination would depend mainly upon the physical location and intended use of a device, rather than the device itself.

One respondent stated that the schema should not require using IP numbers. Another one described that fixed IP addresses are usually used, and they are not connected to the internet.

6) What requirements do you have for security (e.g. restricting access to media flows and protection against malicious traffic)?

The responses indicate that High security is essential. Authentication and security should be based on best practices for IP networks (existing tried and trusted methods).

One respondent summed it up as a need to be able to prevent spoofing of flows by unauthorized users, devices, and sources. In addition, to be able to prevent discovery, interception, or manipulation of flows by unauthorized users, devices, and destinations.

Provide authentication controls. Allow accesses only from known, fixed IP addresses of source and destination devices. One respondent suggested that newly discovered flows should be initially "quarantined". (See "auto-discoveries" in previous answers.)

Be able to restrict connections (access flows) based on extensible criteria (e.g. Production ID, facility location, User ID). Provide accountability of media flows not "leaving" a facility (e.g. for security audit certification).

Multiple respondents expressed concern about denial-of-service by way of link exhaustion, whether unintentional or otherwise. The system should be resilient to aberrant traffic, whether malicious or unintentional (e.g. an accidental DHCP server on the network).

Secure flows from onlookers. Public network flows (inter-facility) should be able to be encrypted, which may introduce latency. Private network flows (inter- or intra-facility) can take advantage of an isolated network. Networks are usually isolated by strict firewall rules, or a physical "air-gap". Even on an isolated network, protocols such as SSH and SFTP could be used to make eavesdropping more difficult. One respondent suggested using something like Zixi encryption.

7) What requirements do you have for flow identity (e.g. preserving it during switching or creating a new identity for a switched flow)?

Summary of responses from seven respondents:

- Identifiers should be based on standardized identifiers such as UID to uniquely identity flows and stored files.
- While the formatting might change as the streams flow and are switched it is important that the source identity is maintained. Intermediate identities might be nice to have but are not required.
- There should be the ability to identify the Logical Source and Flow (in the JT-NM/NMOS/VSF meaning), as well as the parents of Sources and Flows. “A suitable document has been provided for NMOS, which provides use cases for how these should be used in many production scenarios.”
- The authentication of Identifiers is an important aspect of security.
- Using a flow identity mechanism to relate multiple essence streams within flow is of high importance. Incorporating concepts from previously designed mechanisms for flow identity are desirable (e.g. SDP). There is a desire for a hierarchical organization of flows that travel intra , as well as, inter-facility. The nomenclature system should allow single-essence and multi-essence flows to continue to maintain notional relationships with their related peer flows even after leaving one facility for another.

8) What requirements do you have for management and monitoring?

Summary of responses from eight respondents:

- Management and monitoring tools from different vendors should use common tools and procedures to promote inter-operability. These tools should use established protocols such as REST, JSONRPC, IGMP, and SDN.
- Monitoring is required for both the data streams and the nodes in devices through which these streams flow. To avoid failures tools should have diagnostic capabilities to probe points in the data flow and trace signal flows and paths through the network. Communication with users requires status reporting and visualization. Reporting should include passive (good/bad, on/off) and active (parameter level) to detect pending failures or performance deteriorations. Status changes should be reportable either automatically or on demand.
- A standardized or common protocol should be available for configuration and control of devices on the network. Devices should advertise their capabilities, including what can be controlled or configured.

Devices should be able to discover other devices and their capabilities. Control and configuration should support both fixed and dynamic flow management with scheduled and rule-based monitoring and reporting.

- Access and control must include an authorization procedures and mechanisms.

9) Please highlight any opportunities you feel SMPTE should consider for standardization work - what aspects of the IP ecosystem do you feel should be standardized by SMPTE, which should be standardized by other relevant organizations, and what should remain vendor-specific?

Seven out of the ten respondents to the User Survey provided answers to Question 9. Various suggestions were made as to what SMPTE should do, and these are listed below. In a few cases, respondees indicated activities that SMPTE should not undertake.

It was suggested that SMPTE should take the lead on standardizing where possible. Also, that there might be opportunities for SMPTE to leverage protocols from other standards organizations where applicable, thereby minimizing efforts that involve reinventing the wheel, and leveraging SMPTE's specific focus on media & entertainment. SMPTE ST 2059 / IEEE 1588 was given as an example. There was a request that proprietary and patent encumbered formats should be avoided.

Recommendations for SMPTE Work

10. Link quality and open error correction: SMPTE should look at more efficient FEC for compressed and uncompressed workflows.
11. The overall education of vendors building IP devices needs to improve dramatically. There are some basic fundamentals about how multicast and IP packet timing work that are ‘off the radar’ for many vendors who are unaware.
12. Definition of how to include signal components required by law, such as captioning language and descriptive service identifications.
13. Timecode or frame stamping since frame rates are still archaic with 30/1001.
14. Possibly UID bank for company registration to help identify signal sources and destinations.
15. There should be a standardized SDN protocol for control of switching systems. We have lived for far too long with numerous router control protocols that all essentially do exactly the same thing.
16. There presently seems to be an opportunity to standardize on endpoint device control across vendors for flow subscription. It is unclear whether that standardization would be best served by SMPTE, another organization, or between vendor-partners.
17. Encapsulation should not be vendor-specific. There should be a standard API to send commands to a broadcast device.

What SMPTE should not do:

- Accurate switching mechanism, among others, should remain vendor-specific.
- Error concealment should also be vendor-specific.

Additional Industry Recommendations

Validation of devices as being SMPTE ST 2110 compliant would be helpful as the interop of IP devices has much greater packet delivery timing variability compared to deterministic SDI signals.

Annex B Technology Provider Survey

The SG conducted an industry survey among a large group of technology providers for Media over IP solutions. Based on the limited number of responses the SG was not able to draw comprehensive conclusions.

Annex C The Three Planes Model for Media Systems

In order to model the protocol structures of ATM, ISDN and early telephony systems the *Three Planes Model* was developed more than 50 years ago. In 1998 the EBU and SMPTE referenced this model extensively in the published paper “Task Force Report for Harmonized Standards for the Exchange of Programme Material as Bitstreams”. This 191 page report was the first effort by media industry technologists to explain the technology, methods and systems aspects related to the move to IP from analog and SDI. See https://tech.ebu.ch/docs/techreview/ebu-smpte-tf-bitstreams.pdf. This model is useful for understanding how the data layers and protocol layers of packetized media networks and their components interoperate. The components in this model include switches/routers, senders, receivers and “media nodes” of all types.

The basic idea behind the 3-planes model is shown in Figure 1. Each plane is a stack that includes the lowest essence layer data structures with other protocols layered above. Of course, this model applies to any networked system but for this Appendix only media-related examples are referenced.

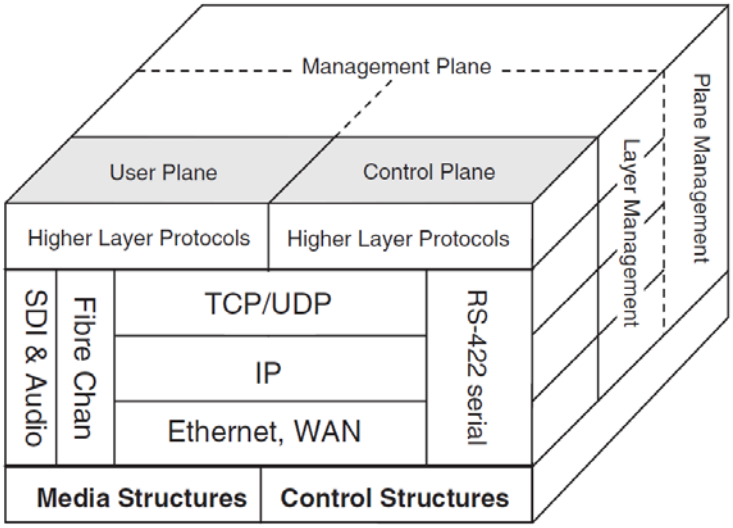

Figure 8: Three Planes Model – Example for a Media System

Using this model, the 3 planes are defined as follows;

Data or User Plane: The user-data (and associated layers) that is transported across a network as flows or files or is input/output to a device. Examples of this plane include all manner of A/V media essence and metadata; SDI essence/transport, AV/RTP/IP/UDP as outlined in SMPTE ST 2022-6 or files moved via FTP.

Control Plane: Signaling (and associated layers) that controls or instructs devices in some manner. Examples are the API commands for device control (e.g. record, play), SDN switch/router commands such as those defined by OpenFlow, triggers to initiate an action, and other control-

related signaling. SMPTE's ST 2071-x "Media Device Control Protocol" and the SIP connection protocol (IETF RFC 3261) are examples of Control Plane usage.

Management Plane: Device management signals (and associated layers) for faults/warnings, configuration, performance, security and other management features. Examples of this Plane include SNMP (IETF RFC 1157 and others) and NETCONF (IETF RFC 6241 and others).

Not all devices/modules will support all three planes. Some will have one, two or all three. Naturally, a device supporting three planes would be more useful in a managed network than a device with only say a data/user plane.

Model Usage

Professional media networks may be described and analyzed with the help of the 3-plane model. Whenever protocols, signaling or data-types are discussed their associated plane may be referenced to aid in the understanding of the operational characteristics of a system. Figure 2 illustrates one way to view the 3-plane model using a train system. The data/user plane are the tracks and train, the control plane routes the trains along its journey and the management plane keeps tabs on the train's type, passengers, location and so on. For packetized networks, protocol examples are listed in the figure.

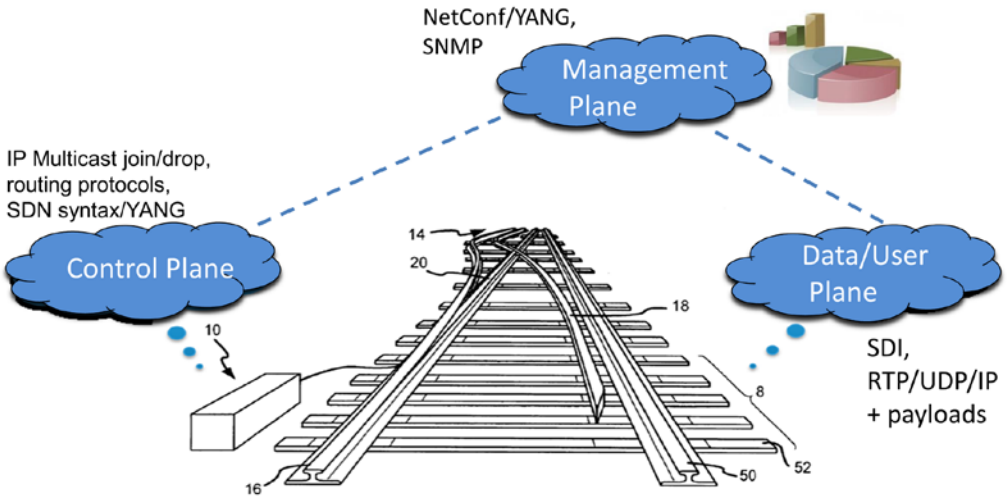

Figure 9: The Three Plane Model Allegory

The 3-plane model may be used to document how a hardware device or software module functions. Some example cases are;

- An equipment vendor documents how their products support different aspects of the three planes. For example, a vendor of a video processing service (software module) could provide the specs for each plane that the software supports.
- An equipment vendor of a hardware media receiver of SMPTE ST 2022-6 flows could provide the specs for each plane that the device supports.
- End users and designers document their systems based on guidance from the 3-planes

- End users may request that vendors provide documentation (Request for Information) based on the 3-planes structure.
- Vendors of test equipment may specialize on one, two or all 3-planes.

The bottom line is the 3-plane model provides a structured way to analyze, design, document and discuss the functional characteristics of media products and systems. Using the “common language” of the 3-plane model enhances the media industry’s ability to manage the transition to a packetized infrastructure across vendors, systems integrators and end-users.

Annex D JT-NM User Stories Regarding Flow Management

The JT-NM collected 136 unique user stories from media organizations, manufacturers and consultants that identified a number of User Requirements for networked professional media. This annex grouped these stories into sixteen Use Cases which are composites attempting to capture the overall spirit of the original stories.

Below are the user stories that this Study Group believe are relevant to Flow Management in PMN.

NOTE: Additionally, Respondents may reference any of the original User Story listed in the Task Force publication Report on User Requirements, which is available at http://tech.ebu.ch/jt-nm, if necessary.

11.1 Configuration (CONFIG)

As a facility operator, I want to have flexible error-free configuration to:

(CONFIG-1) be able to quickly add and configure new equipment and elements;
(CONFIG-2) be able to auto-discover devices attached to the network;
(CONFIG-3) be able to have the configuration of devices be intelligent and highly automated;
(CONFIG-4) be able to have an excellent management/monitoring view of the system;
(CONFIG-5) be able to deal with the variety of formats, stream-types, and file types.

So that I can be on-air quickly, avoid the human mistakes and errors associated with high complexity repetitive engineering tasks, to understand faults in a timely manner.

11.2 Commercial Off-The-Shelf (COTS)

As a systems designer I would like to deploy commercial IT technology for use in professional media applications to:

(COTS-1) take advantage of the marketplace economics of IT technology including packet-based networking, servers and storage;
(COTS 2) make use of the extensive and well trained base of design, operations, and maintenance personnel available in this field;
(COTS-3) deploy enterprise-class capabilities and redundancy options;
(COTS-4) use any one of a number of monitoring, diagnostic and troubleshooting tools that currently exist for enterprise deployments of IT infrastructure.

So that I can reduce the total cost of ownership of my professional media operations.

11.3 File-based (FILE)

As a video editor, I want to:

(FILE-6) be able to change dynamically between streaming and high-quality transfers.

So that I can get the best signal and content quality while editing on low-bandwidth connections.

11.4 Interoperability (INTEROP)

As a system architect, product designer, manufacturer or content provider, I want to:

(INTEROP-1) be able to use readily available and accepted packet-based standards, technology (e.g., IEEE and IETF standards for networking), interfaces (e.g., APIs), components and products in a multivendor environment;

(INTEROP-2) be able to ensure that all network-attached devices are designed and tested to operate in likely real-world scenarios;

(INTEROP-4) be able to have control surfaces that are conceptually decoupled from the software control APIs of the underlying infrastructure and equipment;

(INTEROP-5) be able to design and manufacture systems and test compliance to an industry-standard interoperability specification;

(INTEROP-7) be able to use IPv4 or IPv6 (for an IP-based solution);

(INTEROP-9) be able to use "self-contained" / "self-defining" streams with software-defined connections and/or physical-only connections;

(INTEROP-10) be able to include communications (e.g., "intercom") along with content streams;

So that my operations are optimized, I can have maximum vendor sourcing flexibility through "plug-and-play", "future proof" my system designs, I can choose the appropriate human interfaces for the evolving workflows independently of core infrastructure, maintain quality and compliance with broadcast regulations (e.g., US FCC CALM), I can manage the large (and growing) number of network-attached device addresses, and I can meet the media format needs of my downstream customers.

11.5 Monetization and Revenues (MONETIZE)

As a professional media content producer, I want to:

(MONETIZE-4) monitor media resources (network/processing/storage) usage.

So that I can gain more revenue from each of my content sources, through larger numbers of subscribers, maximize benefits for us getting better advertiser's satisfaction and personalized user experience and I can bill to service usage.

11.6 Provisioning (PROV)

As the systems engineer of a professional media facility I want to:

(PROV-1) be able to use state-of-the-art tools to deploy professional media connectivity whenever and wherever I need it;

(PROV-2) be able to send professional content over the Internet, meeting our quality needs, but taking advantage of the self-routing and self-provisioning capabilities of the Internet;

(PROV-3) be able to rapidly (and in some cases, automatically) set up streams from new devices;

(PROV-4) be able to have my infrastructure scale automatically with load balancing capabilities that take advantage of various links available;

(PROV-5) be able to have my workflow automatically adjust to incorporate the correct transcoding so that when I provision a stream, the format type at the destination node is correct;

(PROV-6) be able to quickly set up efficient distribution networks that deliver the same content to multiple places;

(PROV-7) be able to provision a link at a low quality initially, if that is all that is available, but then allow the quality to improve as resources become available.

So that I can rapidly meet the business-driven operational needs of my company and make economical decisions about the links I use for transport of professional media.

11.7 Quality of Service for File Transport (QOS-FT)

As a system designer or facility operator I want to transport media files between endpoints in non-real-time using a packet-based network with:

(QOS-FT-1) adjustable and deterministic transfer time, including faster-than-real-time if desired;

(QOS-FT-2) upper-end bounded data loss; (define a max transport loss %)

(QOS-FT-3) rate-sufficient to meet the needs of current and future format payloads;

(QOS-FT-4) transport over local, campus networks and Internet;

(QOS-FT-5) multiple defined QoS levels for file transfer based on job, workflow, source or destination;

(QOS-FT-6) the ability to monitor QoS deliver-to-commit and to make adjustments by priority criteria;

(QOS-FT-7) profiles of service to support a variety of workflows. One goal is to provide deterministic file transfers with a known transfer time. For example,

a. Class A: superior QoS similar to what a lossless, high bandwidth, low latency LAN can provide today.
b. Class B: relaxed Class A profile. One or more parameters are relaxed to create a "good enough" profile for many real world use cases.
c. Other classes if needed.

So that I can configure agile file-based media workflows and transport media files using the packet-based network in my facility, be able to select between QoS profiles and trade off costs and performance depending on business needs, and to ensure that files are consistently delivered when they are needed.

11.8 Quality of Service for Streams (QOS-S)

As a system designer or facility operator I want to transport synchronized, end-to-end, real-time, muxed or individual, audio/video/metadata streams over the packet-based network with:

(QOS-S-1) video-frame/audio-sample time accuracy (see Timing case);

(QOS-S-2) very low latency;

(QOS-S-3) lossless transport;

(QOS-S-4) a rate sufficient to meet the needs of current and future format payloads;

(QOS-S-5) transport over local and campus networks;

(QOS-S-6) each stream or group of streams having selectable QoS profile that is defined by the system configuration;

(QOS-S-7) profiles of service to support a variety of workflows. For example,

a. Class A: superior QoS similar to what the SDI ecosystem provides today. This is a "near SDI" profile but not equivalent in every aspect. This also applies to

Media-Associated Data Payloads and their links, not just SDI.

b. Class B: relaxed Class A profile. One or more parameters are relaxed to create a "good enough" profile for many real world use cases that do not require the full feature set of SDI, for example.

c. Other classes if needed.

So that I can configure agile media workflows and transport real-time AV streams using the packet-based network in my facility and be able to select QoS profiles and tradeoff costs and performance depending on business needs.

11.9 Reach (REACH)

I want to exploit the near-ubiquitous reach and rapidly increasing bandwidth of the globally connected packet-based networks (including private leased links and also the public internet) in order to:

(REACH-2) be able to quickly create ad-hoc live interconnections that are able to utilize the available network;

So that I can improve time-to-air and improve staff, equipment, and budget utilization.

11.10 Reliability (REL)

As a professional media organization, I want to:

(REL-1) implement redundant paths in my network to ensure that the facility does not contain single points of failure;

(REL-2) identify primary and backup paths of the same stream; redundancy switching among those paths should be seamless;

(REL-3) ensure that a failure of one system in a studio is contained within that system and cannot affect other systems in that studio, or other studios in that facility;

(REL-4) eliminate making on-air mistakes;

(REL-5) include an equivalent function of the broadcast "tally" system in the packet-based network so that devices downstream or, in a routing infrastructure, can understand a bidirectional (upstream/downstream and vice-versa) status of "on-air" so that inadvertent system changes could be locked-out (or prioritized to administrational / override) status;

(REL-6) know the key system reliability specifications that constitute "enterprise-class" network equipment that will be able to transport high-bitrate video signals in a live television production environment.

So that broadcasting can continue without interruption even in the event of failures (including configuration errors) of shared systems, so that I can recover from a link failure without having time gaps in the media, and so that I can effectively communicate with suppliers to explain my requirements and appropriately evaluate products for use in my facility.

11.11 Security (SEC)

As a broadcast media organization, I want to:

(SEC-1) protect against unauthorized access from within the organization or from outside the organization to data, systems control, or media;

(SEC-2) protect against attacks that disrupt the proper function of the organization;

(SEC-3) have appropriate administrative control systems to support dynamic access control to organization systems;

(SEC-4) have appropriate security monitoring and alarming.

So that restricted or sensitive material does not leak to unauthorized users, I can prevent my operation from being disturbed by malicious actions and no one can conduct unauthorized activities under the name of my organization.

11.12 Streams (STREAM)

As a system designer or facility operator I want facility-wide media/data real-time streaming so I can stream:

(STREAM-3) virtual bundles: separate streams and data paths logically grouped as one;

(STREAM-5) across an infrastructure enabled to carry future payloads (such as UHDTV);

(STREAM-6) in a point-to-point or point-to-multipoint fashion as desired;

(STREAM-7) such that media is switchable on video or audio frame boundary (see Timing case);

(STREAM-8) across an infrastructure that scales from small to large installations;

(STREAM-9) between any nodes connected to the packet-based network;

So that I can build agile, real time, lossless, low latency, workflows with the ability to trade off QoS, formats, and reach.
As a video editor, I want to:

(STREAM-12) be able to change dynamically between streaming and high-quality transfers;

So that I can get the best signal and content quality while editing on low-bandwidth connections.

11.13 Sustainability (SUST)

As a professional media organization, I want to:

(SUST-1) be able to separate the physical locations of control surfaces, displays, video and network processing gear to the most appropriate locations for energy usage, efficient cooling, and noise;

(SUST-3) monitor resources (network/processing/storage) usage;

(SUST-4) minimize the energy consumption of storing, streaming and moving media around the network, particularly when idle;

(SUST-5) be able to easily repair, upgrade, maintain and disassemble the equipment when decommissioned;

(SUST-6) ensure the longevity of my design by using future proof technologies;

So that I have the freedom to deploy people and technology in the most cost and process efficient way, save on transport cost, installation time and travelling of operating staff, pay only for the resources that I use, I can also meet “carbon consumption” regulations, reduce OpEx on energy spend and carbon tax, and protect myself against possible future resource shortages.

11.14 Test & Monitoring (TESTMON)

As a facility owner, a media system reseller, a maintenance person, a network operator or an administrator I want to:

(TESTMON-2) be able to monitor full-quality stream audio, video, and metadata at any point in the facility by multiple simultaneous users;

(TESTMON-4) be able to view exception-based monitoring alerts of any stream (such as presence of video/audio/captions) and set off audible alarms based on these;

(TESTMON-5) be able to quality test streams including pass/fail non-destructively in a straightforward manner;

(TESTMON-8) be able to test streams for standard broadcast-style quality measures and standards and for packet-based quality measures and standards;

(TESTMON-9) be able to verify compliance of the end-to-end packet-based network infrastructure to specifications for installation, function, performance, reliability and interoperability;

(TESTMON-10) be able to monitor media network traffic;

(TESTMON-11) be able to monitor systems for compliance with QoS/SLA agreements or for system commissioning and acceptance;

(TESTMON-12) be able to observe packet-based network statistics and trends;

(TESTMON-13) be able to decouple monitoring from mechanism used for media stream transport content for reliability;

(TESTMON-14) be able to see a 'dashboard-view' roll-up of important routes and flows in my facility;

(TESTMON-15) be able to remotely monitor all system parameters in real time;

(TESTMON-16) have a consistent amount of delay between the time a signal is present at the source and the time it appears at a monitoring point;

So that I can ensure that these complex systems are operating as required, diagnose, support and manage to QoS agreements, minimize overall costs and downtime, provide the Quality of Experience (QoE) that my consumers expect, quickly determine the location of errors or outages and take appropriate remedial action, and so that I can quickly and simply verify the presence or absence of critical systems to be able to troubleshoot and restore media services.