SCTE CABLE-TEC EXPO 22
SEPTEMBER 19-22 • PHILADELPHIA, PA

2022 Fall Technical Forum
SCTE • CABLELABS • NCTA

A Demuxed State of Mind: Transforming Content Ingestion and Distribution in an OTT World

A Technical Paper prepared for SCTE by

Yasser Syed, PhD
Distinguished Engineer
Comcast
Comcast Center 1701 JFK Blvd. Philadelphia, PA 19103
1-215-286-1700.
yasser_syed@comcast.com

Alex Giladi
Fellow
Comcast
Comcast Center 1701 JFK Blvd. Philadelphia, PA 19103
1-215-286-1700.
alex_giladi@comcase.com

Table of Contents

List of Figures

1. Introduction

Content is comprised of video, audio, and text or auxiliary information(e.g., closed captioning). An absence of any media component can lead to an incomplete media experience. A non-synchronized playout of the media components can lead to an incomprehensible media experience. Yet each media component has a separate creation, production, distribution encode and client decode path in its workflow chain with each step traditionally requiring all media components to be received together before processing such that complete and synchronized media playout is ensured.

The speed of each step is dependent on the processing of the slowest media component of the content. Processes can be both automated and manual with creative processes being usually slower than automated processes. Automated processes can still add latency to the workflow, dependent on the amount of compute power needed for operational processes or scene analysis and scaled to match expected quality output. And yet, with the introduction of cloud processing, this can be adjusted.

How can we create evolutionary changes to these workflows? It may be through adapting media workflows to be more like data workflows and taking advantage of better bandwidth and adjustable compute power. Through IP interfaces, increased network capacity, and volume-efficient cloud server processing, media workflows can be more efficient, and deliver better quality along with additional media experiences. But these improvements require us to separate (demuxed) media components to be handled with better optimizations in today's technology environment. Demuxing content allows for better handling of processing demands but reassembly is also important and should add additional information for resynchronization of media components in order to make the playout of the content feasible.

This paper will provide an overview of these processes in the context of traditional media workflows, what can be done today with present technologies, and what could be done in the future with newer technologies like over-the-top (OTT) delivery and cloud processing.

2. Production Media Workflows

Content origination starts with production and distribution workflows (see Fig 1) from capture to editing to distribution by service providers. This can be in the form of scripted content (e.g. movies and TV shows) or live events (e.g. a football game televised live). Some operations that happen are more automated such as chroma subsampling (4:4:4/4:2:2 to 4:2:0), bit depth reductions (16/12 to 10 bits), audio formatting, or lookup table (LUT) conversions for high dynamic range (HDR)/standard dynamic range (SDR) conversions. But there are a lot of creative operations happening here as well, such as editing, shading, translations, and subtitling, all of which can take longer but could also be parallelizable. Ultimately this is received by the service provider in the form of contribution linear feeds (e.g. see SCTE 277) or mezzanine assets. Multiple feeds or files may result from this process.

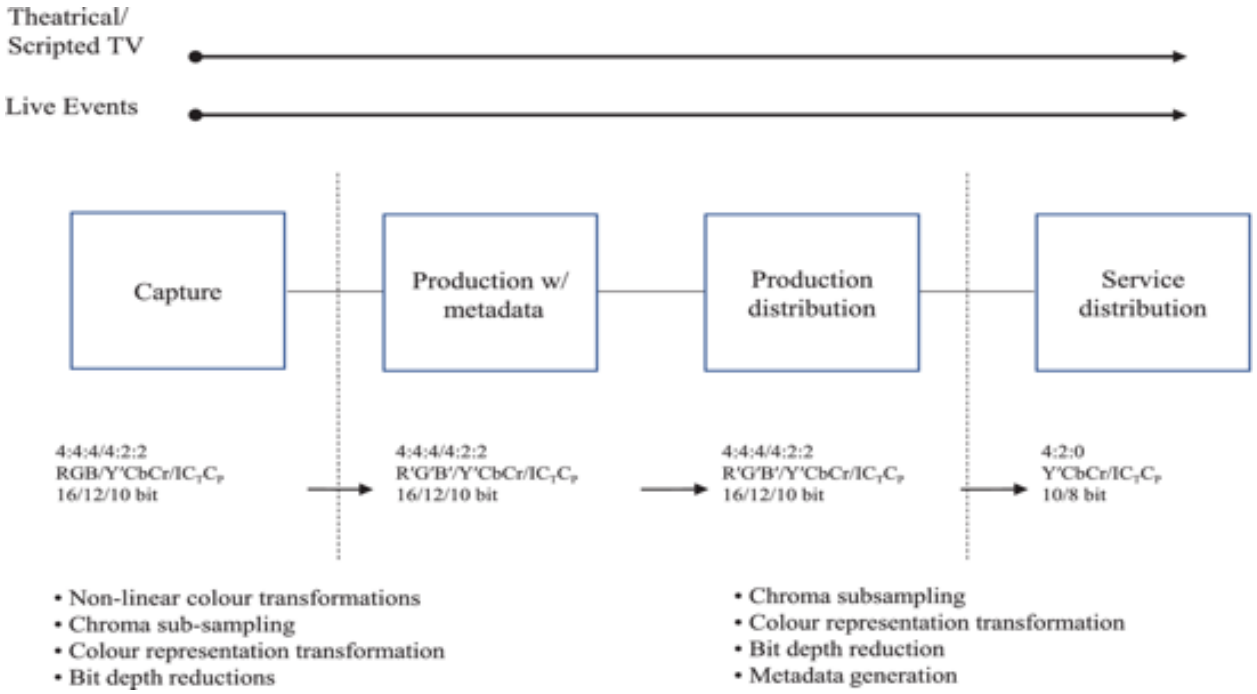

Figure 1 – Production Workflow[1]

In terms of moving assets or feeds around in this domain, SDI (Serial Digital Interface) (see Fig 2) is widely used (e.g. SDI, HD-SDI, 3G-SDI, 12G-SDI) providing a baseband interface to allow for frame editable content to be distributed within a local area. An SDI frame allows for a timing frame to encapsulate video, audio (16 channels), closed captioning, and timecode within a serial stream.

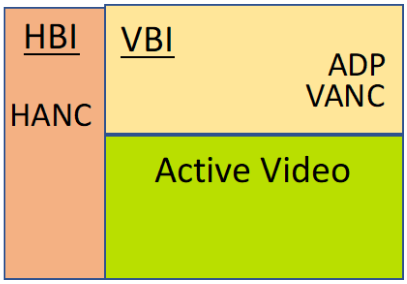

Figure 2 – SDI Frame

The limitations in this type of setups are that content is only locally moveable and that the cabling infrastructure is designed specifically for moving SDI signals around to production and post-production sites. With the developments from ST 2022-6, which encapsulates SDI over IP, it facilitated production workflows to begin to move beyond the local plant.

3. Carriage of Media in Distribution Service Workflows

Once the studio passes content to the service delivery workflow, the content is mostly formed and it is more of a question of delivering the content in the right format for the client player for playout. The

[1] From ITU-T H. Supp 19|ISO 23091-4

workflow still follows the live and file-based workflows that are then encoded for distribution to create a decodable bitstream in which compression of content to utilize bandwidth efficiently is essential to operate within the capacity and scalability limitations of the distribution network. Most of the media operations are done on a single device on which the content is unwrapped, the components are processed, and finally the components are reassembled back into an integrated content format again to be carried on the network for device distribution.

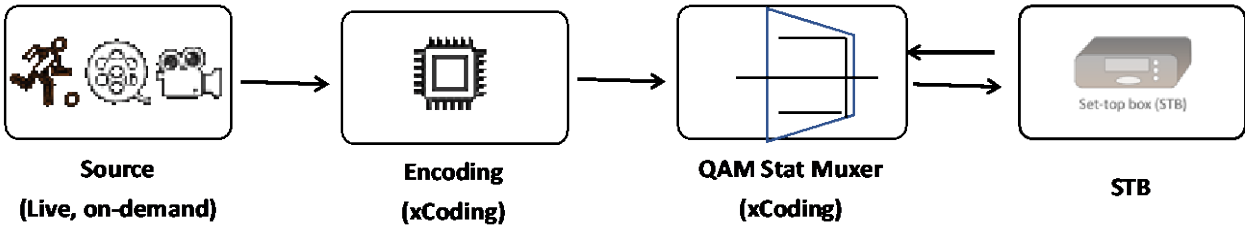

Figure 3 – MPEG-2 TS Service Distribution

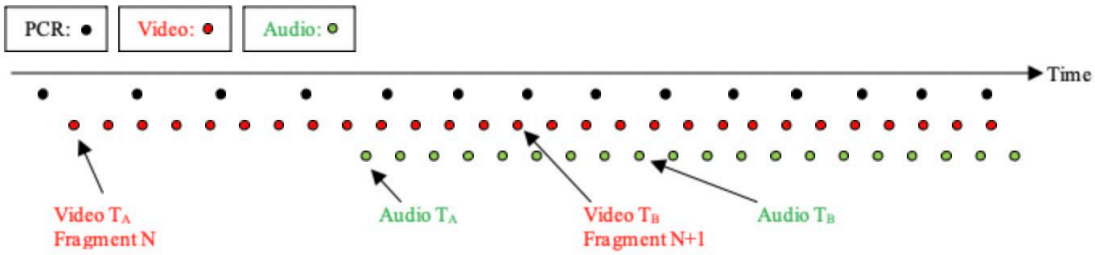

Figure 4 – MPEG-2 TS Muxiplex Composition[2]

For traditional one to many broadcast QAM delivery over MVPD networks, an MPEG-2 transport stream is used to carry content (see Fig 3). The transport stream packetizes each media component into elementary streams, which are grouped together as a program stream. The MPEG-2 packetization allows for the media component packets of the same timeframe to be interspersed in the data stream (see Fig 4), such that a decoder with coded picture buffer can receive the packets and output a time-continuous stream while taking advantage of temporal compression strategies to reduce the bandwidth demands of delivering the content.

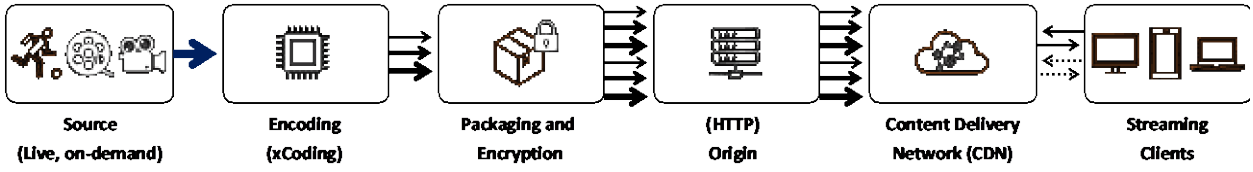

Figure 5 – Adaptive Streaming HTTP Service Distribution[3]

[2] From SCTE 223 – Adaptive Transport Stream

[3] From IEEE ICIP Tutorial Ali Begen/Yuri Reznick Sept 19th, 2021

With recent adaptive streaming technologies that are used in OTT delivery, service functionality moved from a single encoding device to distributed functions (see Fig 5). Because OTT services deliver content over the internet using unicast internet protocols, the operations of encoding, packaging, placement, and delivery were spread across the network; additionally, media components did not have to be packaged together for delivery, but ultimately delivery still was constrained by the same decoder buffer restrictions on outputting a continuous stream of frames.

4. Real-Time Considerations in Media Workflows

Video is a series of frames displayed to the eye at a certain rate (see Fig 6), allowing a simulation of the what the brain would perceive in the real world. The rate to achieve this is known as frame rate (frames-per-second [fps]) and can vary from approximately 24fps for feature film content to 60fps for live production and most video sources. There are also some sources at 30 fps and new high frame rate video formats at 120 fps that are better at reproducing fast action.[4] For real-time processes, frames need to be created, edited, produced, delivered, encoded, packaged, distributed, buffered, and then decoded before the next frame is outputted. Causes that will add time to the workflow would be compute time for media processing at different points in the chain, or network bandwidth needed to deliver bits across to different points in the workflow, or simply any creative handling of the content.

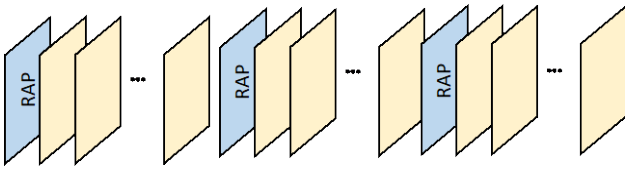

Figure 6 – Stream of Frames

At a constant frame rate 60 fps mode, successive frames need to be outputted every 16.7 ms (1/fps). This means that for a glass-to-glass delivery, all processes need to be completed within this time. Traditional mechanisms to achieve this are to introduce startup delay such that the player buffer could be filled before an initial frame would be outputted, then the duration of the buffer could extend the time that the next frame would need to be sent to the buffer to keep it filled. This adds latency to the video, but through approaches like this, continuous video can be achieved at a manageable bandwidth for the network.

Other ways to address real-time demands would be to reduce temporal compression by creating a simplified group of pictures (GOP) structure such as all I-frames or forward predicted pictures (FPP). Another way would be to increase the network bandwidth such that created content packets could be delivered faster than real-time; this would work more for file-based workflows and less for live feeds.

[4] Typical framerates in use also includes fractional framerates such as 23.97, 59.94, or 119.89 which can add some complexities to timing and synchronization factors, but conceptually remains the same. For this paper, integer number frame rates examples will be used for simplicity while discussing concepts.

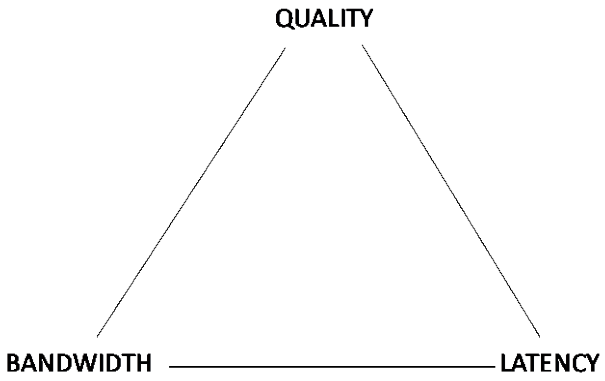

Figure 7 – Quality-Latency-Bandwidth Tradeoffs

But real-time processes are not the only factors to consider when delivering video. Depending on the service, the video delivered also must be of acceptable quality and of acceptable bandwidth for network distribution (see Fig 7). Increasing quality or bandwidth may result in increased delay and designing workflows must consider and balance all three of these factors in delivering content (see Fig 8).

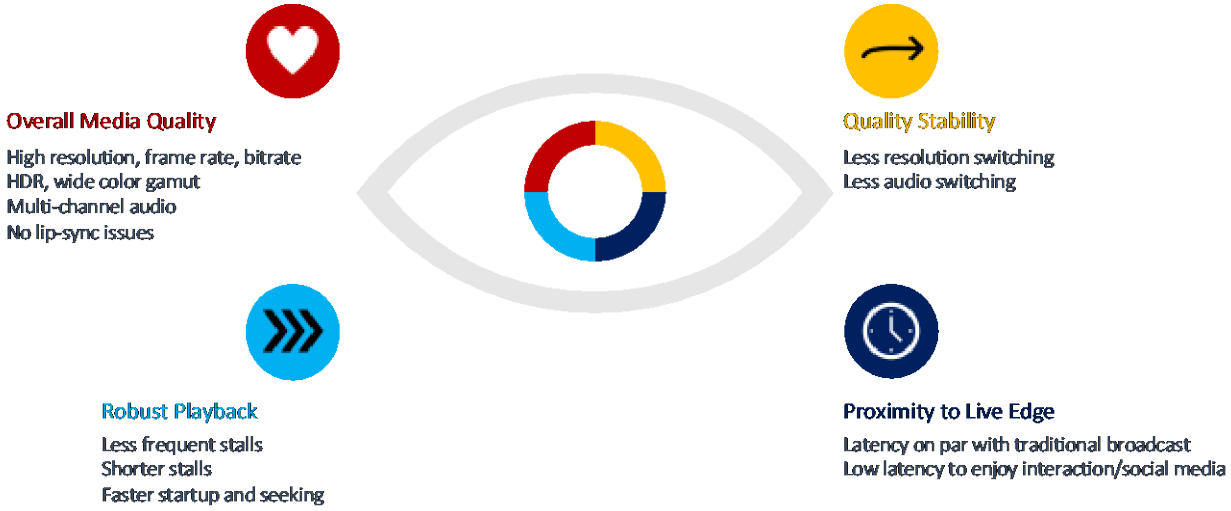

Figure 8 – Factors in Playback[5]

To avoid stalls, the decode buffer must have new frames coming in at regular intervals. To have fewer resolution switching factors, managed bandwidth network connections or smaller bandwidth requirements must be maintained while handling more bits to deal with higher bandwidth or color space demands. The need to reduce latency factors also brings into play reduced player buffer capacity, which puts increased pressure on real-time computation demands.

5. Adapting to Newer Technologies: IP Interfaces/ Cloud Processing

Carriage of Media workflows as data across IP interfaces bring many advantages, and the content production ecosystem is gradually adopting new IP based approaches. For instance, in production workflows, the same source point can be sent to multiple end points in a local network so that editorial or

[5] From IEEE ICIP Tutorial Ali Begen/Yuri Reznick Sept 19th, 2021

production tasks do not always have to be serially performed but could be performed in parallel (see Fig 9). Through SMPTE ST 2022-6, SDI video can now be encapsulated and carried through an IP distribution. A benefit of this is the reutilization of existing production equipment with only an SDI interface while using IP to carry the signals and IP routers to switch and distribute them. With SMPTE ST 2110, the IP carriage takes a further step to natively carry the video, audio, and data as independent essences that are sent separately, thus allowing for individual points to deal with specific media components without sending the entire content. With moving to a demuxed delivery of media components, a timing synchronization is now required to reassemble the media components into the complete content. In SMPTE 2110, this is achieved though using a Precision Time Protocol (PTP) timing mechanism.

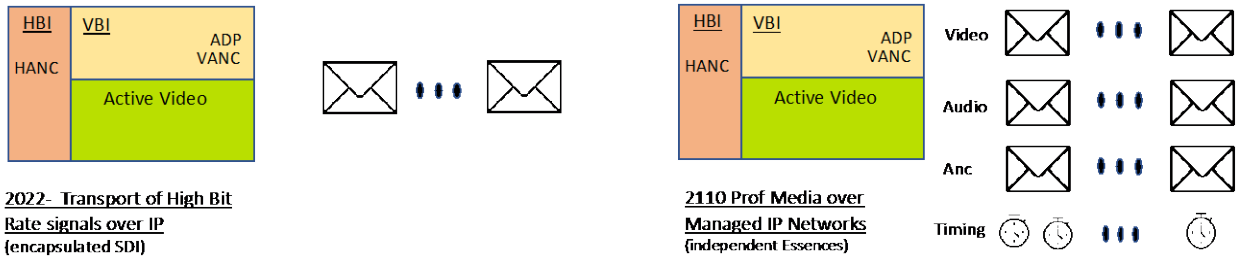

Figure 9 – IP Encapsulated SDI

In adaptive streaming delivery, a similar evolution occurred from what initially was IP carriage of MPEG-2 TS streams (e.g. HLS and DASH). This evolved to utilizing an ISO base media file format (ISOBMFF) delivery that allowed each media component (independent essence) to be requested from a set of options (adaptationSets) and delivered separately using segment timelines as the method to synchronize media components (see Fig 10). Further refinements with CMAF allow segments to be broken up and delivered separately as chunks; this assists in providing lower latency delivery.

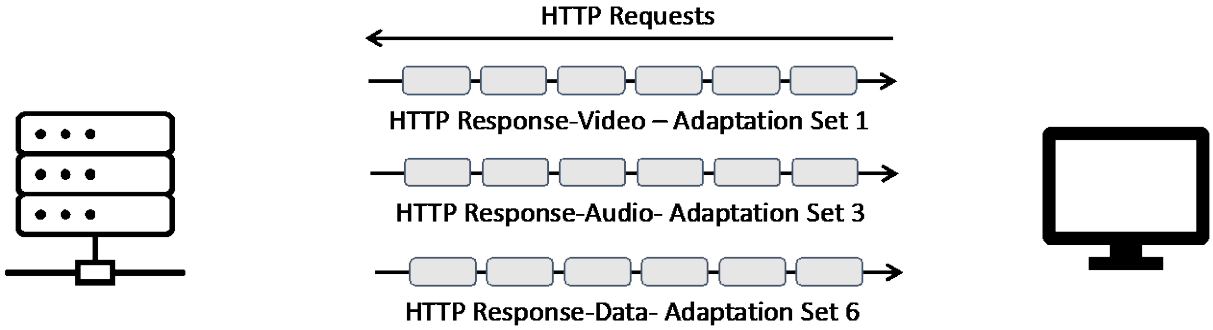

Figure 10 – Adaptive Streaming using HTTP

Cloud processing is now becoming utilized in many places for high performance computing, AI/ML analysis, or on-premise offloading. The modes of service in the cloud can be infrastructure as a service (IAAS), platform as a service (PAAS), software as a service (SAAS), or serverless. For content media workflows, cloud processing can operate in several areas for real-time captioning services, basic SDR/HDR conversion processes, scene detection for more efficient guided encoding, third party processing, or manifest manipulation. For better integration of cloud processing into content media

workflows, it requires more efficiencies in bandwidth workflows to handle even editable workflows and IP addressing to multiple source delivery points (see Fig 11). To make it more practical, compute latency, storage latency, and network latency need to be reduced. For compute latency, more pipelining and parallelization strategies as well as more powerful processors need to be available, depending on the processing task. For storage latency, better caching and reading/writing strategies already exist but need to significantly improve to reduce latency. Lastly, network latency needs to be improved through the analysis and adoption of new strategies in regard to compute placement (e.g. edge compute).

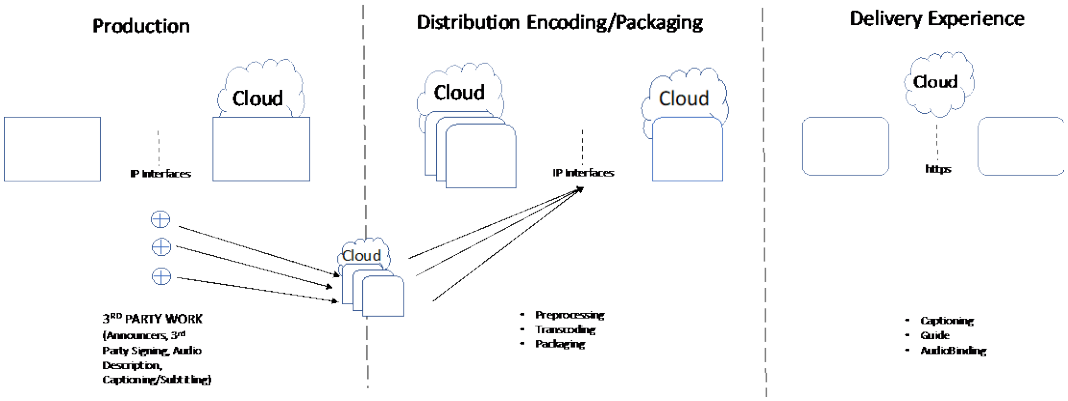

Figure 11 – Roles of Cloud in Media Workflows

6. Content Processing and Content Experiences

In content media workflows, processes can be categorized as automatic or manual (see Fig 12). Automatic processes typically can be done without manual work; examples in production are up/down sampling, tone mapping, and some speech recognition/captioning; while examples in distribution include such functions as transcoding and packaging. Manual processes require more hands-on work and can describe functions like editing, shading, audio description, or third person signing. With cloud processing and artificial intelligence (AI)/machine learning (ML), some of the manual tasks can be done or accelerated with assistance by trained AI/ML approaches. With post-production, developing a format that allows for IP distribution while allowing for frame accurate editing is needed to reduce the bandwidth demands and compute demands to allow for this. Through developments in I-Frame HEVC/VVC or JPEG-XS or J2K, future image/video coding techniques may address these needs in the future.

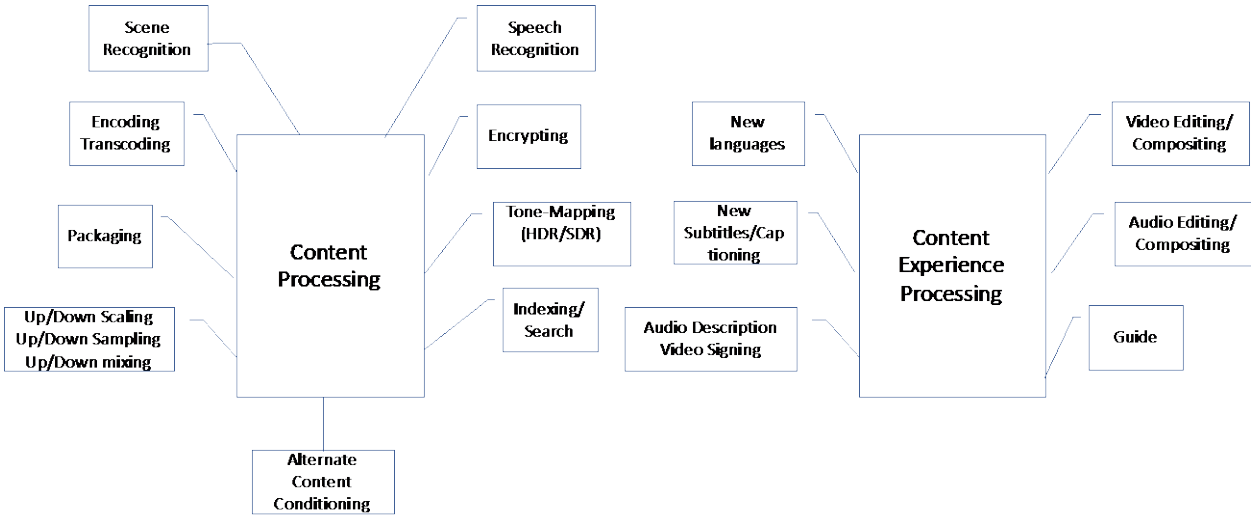

Figure 12 – Content Processing and Content Experience Processing

Another type of category is processing of existing content or adding to experiences of the content asset (see Fig 12 again). Processing of content type of operations does not change the version of the content but may change format, or appearance of the content. These can include adding more subtitle language tracks, or new dubbed languages in audio, or simply adding audio description tracks to the content. Most of these types of processes are more automated tasks rather than manual tasks. Content experience processing usually alters the version of the content and provides additional personalization of the content such as additional languages, alternate versions of content, adding announcers to sporting events, or adding audio descriptions to events. Many of these added experiences are manual processes but with new types of cloud processing , IP carriage, and bandwidth efficiencies, some of these tasks can reduce latency through assistance using these new technologies. An example of this that is now seen frequently is the real-time creation of subtitles from the audio track.

7. Demuxing Content for Improved Workflows

Demuxing content into separate media components can benefit media workflows in both production and distribution by reducing the amount of content that needs to be wrapped, transported, unwrapped, processed and re-wrapped many times in the overall media workflow. For production, demuxing can help with real-time operational processing demands such as real-time generation of closed captioning through implementation of an audio track or in translation an existing closed caption track. With third party signing, it may require generating a video proxy linear stream to send with the audio track. Content processing transformations (see Figure 12) can also be done offsite but requires some sort of frame accessible video format that is protected and is transportable across IP networks instead of the local network. In addition to processing content, information can also be gathered for scenes that can be beneficial for distribution transcoders to do higher quality encodes. Lastly, some additional experience tracks can be generated during this production process.

Using separate media components for processing provides some step up advantages to that triangle of quality, latency, and bandwidth considerations (see Fig 7). For bandwidth, there are fewer bits to be sent across IP networks to process media through the production and distribution workflows. For latency, the ability to send information in a parallel fashion earlier in the media workflow and the ability for the cloud to assign elastic processing power to what the task demands are strong advantages. Additionally, developing AI/ML techniques that may be continually refined would be helpful to get better quality

distribution encodes (including pre-processing techniques) at lower bit rates. Lastly, third party contributions to content assets and feeds for such content enhancements as home/away announcers, third party signing, or real-time indexing for more searchable content could be done across IP networks as well.

8. What Further Work is Required?

Some additional development that will be required – some of which already is happening – to better handle demuxed workflows falls into five categories:

- Automation in production and distribution workflows;
- Synchronization across media components;
- Integration of third party created media components;
- Placement of new content experiences across adaptationSets in manifests; and
- Playability of content assets and linear feeds in an OTT system.

To pull off media components of the content for separate processing, metadata is needed to automate job processes. This metadata, which can include metadata generated by AI and ML processes, can aid in other future processes in the workflow such as enhanced quality encodes. This information when generated or provided needs to be stored in a format that may originate in production but may be needed upon distribution. A format needs to be developed that can provide this time sensitive information in both the production and distribution domains and may need to be stored as a metadata track retrievable in the manifest.

For automation in production, an editable format of the media component needs to be created that can span across IP networks rather than be limited to a local LAN, at a reasonable bandwidth. Some formats being looked at are JPEG-XS, I-frame only AVC/HEVC/ VVC, or J2K. Audio Formats may not need to be compressed since audio file and feeds are smaller but may require video proxies to be sent along with them, but there is also a need also to carry some of the distribution signalling metadata for audio back to baseband carriage. For third party components such as signing, audio description, or subtitling, these type of files or feeds may need to be fed into the distribution workflow rather than returned to the production workflow.

Synchronization between media components is also a factor that needs development. In one area it is more conforming additional tracks to the content asset or file; this basically means each media component should have an aligned timeline that can be used to generate synced timelines on newly created assets. In the production domain and with the SMPTE 2110 format, the PTP timing format is used. In the distribution area, however, time is more aligned with segment timelines and alignment between both approaches needs to be realized. Another area is synchronization between audio and video; in OTT the video segment determines the edit point but there are instances in which the audio segment that goes along with the video may naturally be slightly lagging the video; in those cases that slight lag needs to be maintained.

To integrate third party media assets and linear feeds to the content, IDs need to be created and matched such that manifests can add these new adaptationSets and authentication systems can validate that content. URLs involving the retrieval of content need to consider content distribution network (CDN) design and whether one or more CDNs are involved. There also needs to be a way to identify tracks that are being processed versus tracks that are finished and retrievable by the client player. Gaps in timelines of media components need to be considered and handled as well so mechanisms such as content failover can be considered. Additionally, player behavior needs to be redefined such that new content experiences can be added to an existing media playout which may be added as the manifest is updated. Even when the

concept of a linear feed or a content file being complete no longer exists, the playability of the asset needs to be known.

9. Conclusion

Content is nowadays not a static asset or channel. It is evolving according to the expectations of the viewer and can involve multiple parties creating the overall content experience. More formats, better quality, multiple types of playback devices, and more ways to experience content is an expectation. The platform for media workflows is also evolving to incorporate faster and intelligent processing not limited to all work being done in a specific location. Media workflows for production and service workflows have traditionally been linear, but with the advents of higher bandwidth, IP interfaces for both uncompressed and compressed workflows, and access to cloud processing, linear workflows can evolve. Production efforts can be outputted directly into OTT feeds and into CDNs to be ingested by OTT players as they become aware of what new options they have to experience the content, and authorized third party contributors can also provide new experiences to the content without delaying the workflow. Cloud processing can accelerate the workflows by just adapting processor power to the job demand or by pre-processing analysis to aid in encoding of the content. But these evolutionary efforts do require some adaptation of the current system to make it easier to conform and synchronize separate media assets to the content and to create adaptationSets, IDs and URLs to incorporate these workflows and to make players aware of new choices in the manifest.

SCTE has a number of working groups that are already involved in these areas. In the Digital Video Services committee, there are two related working groups that are involved in these areas. Working Group 1 (Video/Audio) just recently published SCTE 277 (Linear Contribution Encoding Specification) which defines ingestion of signals that originate from production/post-production services. Additionally, WG1 also defines the Video and Audio codec streaming constraints for consumer distribution including recent modifications for adaptive streaming to consumers. On the consumer distribution side, WG7 (adaptive streaming /DASH) works on the suite of SCTE 214 specification which define manifest and segment constraints for HTTP IP delivery of content through MVPD networks.

Abbreviations

ADP	ancillary data packet
AI	artificial intelligence
AVC	advanced video coding
CDN	Content Distribution Network
CMAF	Common Media Application Format
FPP	forward predicted picture
fps	frame per second
GOP	group of pictures
HANC	horizontal ancillary data
HBI	horizontal blanking interval
HDR	high dynamic range
HEVC	High Efficiency Video Coding
HTTP	Hypertext Transfer Protocol
IASS	infrastructure as a service
ID	identification
I Frame	Intraframe
IP	Internet Protocol
ISOBMFF	ISO base media file format
J2K	JPEG 2000
JPEG	Joint Photographic Experts Group
LAN	local area network
LUT	lookup table
ML	machine learning
MPEG	Moving Pictures Experts Group
OTT	over the top
PAAS	platform as a service
PTP	Precision Time Protocol
QAM	Quadrature Amplitude Modulation
RAP	random access point
SAAS	software as a service
SCTE	Society of Cable Telecommunications Engineers
SDI	serial digital interface
SDR	standard dynamic range
SMPTE	Society of Motion Picture and Television Engineers
Url	uniform resource locator
VANC	vertical ancillary data
VBI	vertical blacking interval
VVC	versatile video coding

Bibliography & References

ISO/IEC 13818-1:2020, Information technology - Generic coding of moving pictures and associated audio information: Systems.

ITU-T H-Series Supplement 19| ISO/IEC TR23091-4 Usage of Video Signal Type Code Points

ANSI/SCTE 277 2022 Linear Contribution Encoding Specification

ANSI/SCTE 223 2018 Adaptive Transport Stream

SCTE 214-1 2022 MPEG DASH for IP-Based Cable Service Part 1: MPD Constraints and Extensions

ANSI/SCTE 214-4 2018 MPEG DASH for IP-Based Cable Service Part 4: SCTE Common Intermediate Format (CIF/TS)

Ali C. Begen and Yuriy A. Reznick, Advances in Multimedia Streaming: Algorithms, Standards, and Optimization Techniques IEEE ICIP Tutorial September 19th, 2021

SMPTE ST 2022-6: 2012 SMPTE Standard - Transport of High Bit Rate Media Signals over IP Networks (HBRMT)

SMPTE ST 2110-20: 2017 SMPTE Standard – Professional Media over Managed IP Networks: Uncompressed Active Video

SMPTE ST 2110-30: 2017 SMPTE Standard – Professional Media over Managed IP Networks: Uncompressed PCM Audio

SMPTE ST 2110-40: 2017 SMPTE Standard – Professional Media over Managed IP Networks: Ancillary Data

AWS Media Blog Part 1:Background and key benefits of SMPTE 2022-6 on AWS Elemental Live, Aug. 16th, 2021, https://aws.amazon.com/blogs/media/awse-part-1-background-key-benefits-smpte-st-2022-6-aws-elemental-live/

AWS Media Blog Part 1: Background and key benefits of SMPTE 2110 on AWS Elemental Live, Nov 3rd, 2020, https://aws.amazon.com/blogs/media/part-1-background-and-key-benefits-of-smpte-st-2110-on-aws-elemental-live/

