Fast Metadata for ST 2110
The SMPTE FMX Project

Paul Briscoe, Televisionary Consulting
Toronto Canada

On behalf of Evertz Microsystems

Cautionary Note

This is active SMPTE project work in 32NF-60 DG SVIP

SMPTE work is confidential

Can't disclose the details of the activity

Can disclose the 'what' and discuss the ideas that are input to the process

Oh, Metadata

- Metadata, a necessary evil in media systems
- Exists all over the place from acquisition through consumption
- Many types and uses, comes and goes throughout the process
- 2110 lives in the same application space as SDI
  - Hybrid system era is upon us
  - Designed for easy migration to / from SDI
- SDI has a robust metadata system in ST 291
  - Uses VANC and HANC for transport
  - Registered applications through SMPTE RA

What is Metadata?

"Data that provides information about about some other data"

- Webster's

In our industries, that "other data" is usually video and audio.
Some metadata is necessary in order to use the data – at all even
Other metadata is 'nice to have' information – optional use
Some metadata is static / persistent (never changes / always there)
Other metadata is dynamic - may change, may come and go
Some other metadata is actually essence:
e.g. Closed Captioning and Haptic / Tactile are media essence masquerading as metadata just because it's the easiest way to move and store it.

Metadata is important

Characteristics of Metadata

- Temporally bound to the video essence – frame accurately
- Added at the sender, used selectively by receivers
  - Use what you want, ignore the rest
- There are many applications
  - Some intended for 'internal use' within production workflow (e.g. timecode)
  - Some intended to propagate to point of consumption (e.g. Closed Captions)
  - Some come and go throughout the production workflow
  - Some get lost
- Managing metadata in a facility is a challenge.

What About New Metadata?

- Legacy metadata is simple – easy to add
  - Sufficient for applications in SDI era
    - Non-interactive
    - Simple use contexts
- 2110 metadata will become rich
  - Interactive environment in IP
  - Technology has evolved
  - Opportunity to do more things with 2110
  - Far more than today's SDI applications to consider

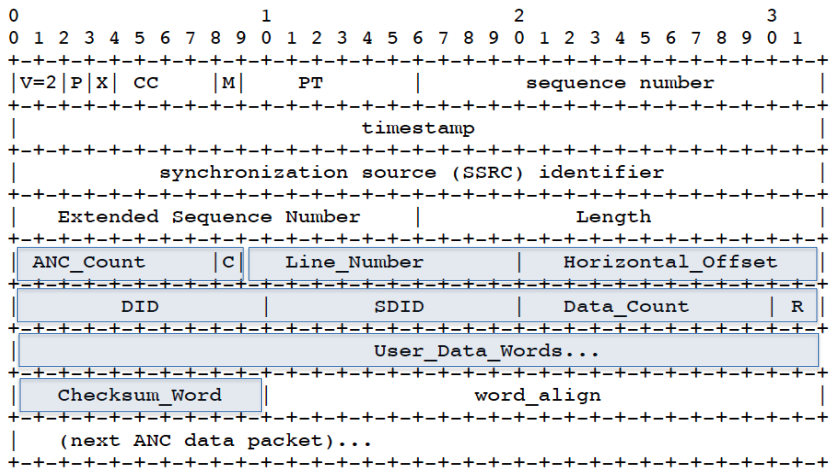
Problem to be Solved

- The ST 2110 suite does not include a method for time-aware (i.e. synchronized with some essence) transport of non-ST 291 metadata
- The existing ST 291 metadata standards are:
  a) intended for use with SDI transports
  b) insufficient for applications requiring significant amounts of metadata
- Many current and future 2110 applications would benefit from a new metadata transport that:
  - was not constrained to limitations of ST 291
  - supports rich and arbitrary metadata payloads
  - offers time-aware native transport using the ST 2110-10 model.

Metadata in 2110 Today

- ST-2110-40 transports SDI metadata using ST 291 formatting
  - Sender prepares an SDI ANC-compliant ST291 payload
  - Sender wraps entire ANC payload in IP for transport
  - Transmits ANC packets on one or more unique IP addresses
    - May use one address per service
    - May group services on one IP address
  - Synchronization to a given video frame is through 2110-10 mechanisms
  - Receiver unwraps the IP container, then unwraps the SDI container to get to the payload to use it, or places it back into SDI in the SDI container.
    - Two-step abstraction
  - Supports all known ST-291 metadata
  - Good evolutionary bridge method
  - Limited by ST-291 capacity and capabilities

ST 291 over ST 2110-40 (RFC8331)



Challenges with ST 291 Metadata in IP

- Works fine for SDI ANC, but there are potential interop issues
  - 2110-40 allows one or more ANC payloads per -40 packet – “no rules”
    - Do you put one service per -41 stream? That’s IP-address-hungry
    - Do you group services in one or more -41 stream(s)?
      - If they’re glued together, how do you add / remove one (like SDI ANC)? - No “Marked for deletion” service
- SDI + ANC technology horizon is in view
    - Not all future metadata needs to be backwards compatible with SDI ANC.
- Inefficient / unclean
    - Wrapping up an already-wrapped entity
    - This is why we use AES67 as 2110-30 rather than AES3 over 2110-31
- Only supports ST-291 SDI ANC services
- Consumes one or more IP addresses

Handling New 2110 Metadata

- Today we get there via SDI ANC
  - Define an ST291 application
  - Register it? If intended to get back to SDI
  - Wrap / unwrap into / out of 2110-40
- Does it fit the constraints of 291? How to adapt?
- Is funny business required to accommodate?
- Why are we doing it this way?
  Natural first step, hybrid systems with SDI

The Fast Metadata (FMX) Proposal

- Rapid, time-aware delivery of arbitrary metadata streams in concert with a media stream – or not
- Standardized encoding method for transport
- Extensible and consistent data structure
- Does not consume additional IP addresses
- Payload agnostic – transport any arbitrary data
- Can be synchronized with the media stream (2110-10 + 2059)
- Can support legacy ST 291 SDI ANC metadata (if we want)
- Media-stream agnostic – any time-aware stream can use it
- Can be an independent data channel – doesn't need to be stream-associated
- No changes to other 2110 documents

SMPTE Project Statement

Problem to be solved:: In the ST 2110 suite, there is provision for carriage of legacy ST 291 ANC metadata. This legacy SDI-centric method is described in ST 2110-40 and provides an excellent bridge to IP for SDI services. In the IP-centric future, there will undoubtedly be new metadata requirements which will have no relevance or application in the SDI domain and additionally may not be appropriate for efficiently wrapping in legacy ST 291 SDI ANC packets for IP transport with 2110-40. This project will create a new transport method for arbitrary metadata using RTP packets in the same manner as the other 2110 standards. This method will not be strictly tied to any type or form of metadata. Use of the 2110 synchronization mechanism will permit transmission in association with a 2110 essence stream, for example transmitting the packet at the video Alignment Point. It is proposed that the SMPTE ST291 RA be used to index payload types for use in this Standard using a new RA Data Type and the existing DID / SDID addressing scheme. Two initial applications are proposed: carriage of a 2110 sender SDP record using a TLV structure (separate Project) and carriage of the (in-process) SMPTE TLX Time Label (SMPTE liaison).

Project scope:: Develop a Standard to enable arbitrary data to be transmitted and synchronized with another RTP stream in an ST 2110 environment.

Specific tasks:: Establish detailed design goals Look at IETF for existing RFCs that could be used research / establish exact method for transport and data formatting Investigate / establish use of SMPTE RA or other

What is Does

- Uses a consistent data structure (such as a “KLV” -type structure)
  - Well-known representation
  - Extensible – beyond foreseeable needs
  - Optionable / scalable
  - Easily parseable
- Carries the naked metadata – no unnecessary wrappers
- Delivers at a specific time
  - In concert with (or before) the first packet of video
    - Real (or predicted) timestamp
    - Early delivery means early processing
      - Could deliver immediately after last packet (M bit packet) of prior frame
- Can also be used on its own with no associated stream
  - Arbitrary data transport
  - System-level or other bulletins sent periodically (like ATIS)

What Does it Enable

Carriage of new payload types
- Standardized data structure can carry other specific structures
  - XML, JSON, etc.

- Carriage of arbitrary metadata for any time-aware RTP stream
  - Metadata we haven't dreamed up yet
- Address conservation
- A cleaner way to carry today's ANC services
  - Overcomes limitations of 2110-40
  - Could still use existing registry and DID / SDID scheme if we want
  - Metadata payload would be identical – just not wrapped for SDI

High-level Proposed Methodology

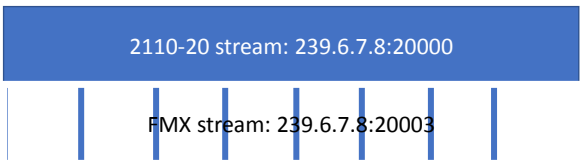
- Transport
  - Directly associated with a 2110-xx stream
    - Use same IP address as stream
    - Use "UDP port number + offset" enumeration to differentiate metadata stream
    - Use same associated stream's RTP timestamp clock rate and time value
      - RTP Timestamp can be altered for early delivery
- Wrapping
  - Use a *generic* TLV- / KLV-type structure (not ST-336 KLV)
    - Payload agnostic
    - Well-known and widely used format

Blob in a Blob

- Two KLV-like wrappers are used
  - Outer wrapper is the boxcar that 2110 carries
  - Can carry one or more outer protocol boxcars
  - Each can contain one or more inner wrappers
  - Inner wrapper(s) contain the actual metadata
  - Designed specifically for the application(s)

Just Another RTP Stream

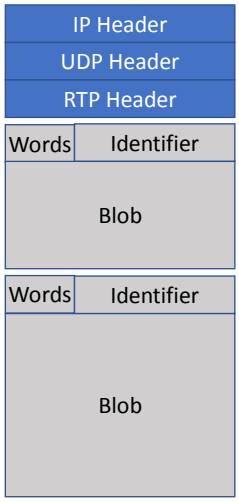
- Utilizes the same multicast address as the related media
  - It uses another port number
  - Similar to the approach used for FEC



- The FMX packets have RTP timestamps in the same units as the media stream
  - If the metadata has a time relationship to the media, then the timestamp indicates how it relates
- FMX packets are in the outer protocol format

FMX Packets – A Familiar Design

- Use Standard RTP transport
  - 16-bit sequence number is plenty
  - RTP-TS in units of related media
- KLV-type structure allows extensibility
  - Defined namespace structure for keys to allow experimental and user-private keys in addition to standardized, registered keys
  - KLV blobs fit inside a packet. Any fragmentation logic is the responsibility of the inner protocol.
  - KLV blobs are 32-bit aligned. Any required padding to achieve alignment is the responsibility of the inner protocol
  - Length = 10 bits = # of 32 bit words that follow
  - Leaves 22 bits for the Identifier – 4 million possibilities
    - Includes ranges for registered IDs and user-private IDs

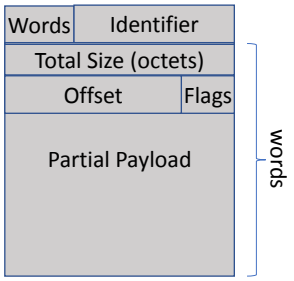


Mapping Arbitrary Payloads Into Blobs

- If the payload is always small enough to fit within a single blob, then it just drops in. The registration can indicate this.
- If the payload might require fragmentation then the registration can indicate how its done
  - The method of 2110-41 appendix A can be indicated
  - The payload document can define its own method
  - Private payloads do not need to disclose how
- A method is provided in -41 Appendix A to simplify mapping existing (already-defined) payloads into this system without needing yet another spec.

2110-41 Appendix A: Fragmentation

- Words == always the # of 10-bit words IN THIS BLOB
- Identifier == always the Identifier of the contents
- Blob == Fixed Header + partial payload
  - Header:TotalSize number of octets of payload, total
  - Header:Flag[0] “Last” 1=this is the last fragment
  - Header:Flag[1] “Reserved”
  - Header:Offset – How many 32-bit words came before this partial payload
  - Partial Payload: sequence of (Words-2) 32-bit words of payload. The last 32-bit word might only have some valid octets inside
- Can transport objects up to 2 GBytes in length



What’s Inside the Blobs?

- Every kind of data needs some document that says how its structured
  - Private data with a Private ID need not disclose the data format/structure
- The -42 project will define a payload structure for 2110-20 tech metadata
- What is required to register a 2110-41 ID (public or private) ?
  - ID (to be allocated by SMPTE from the requested range)
  - Title for the registration entry
  - (for private IDs) Company/Contact Information
  - (for public IDs) Reference to the document that describes the payload contents & structure
    - Detailed reference to a sub-part of a document is possible
  - (for public IDs) Method or Reference for payload mapping into 2110-41 WIBs
    - Can specify “Direct Mapping” for payloads that always fit within a WIB
    - Can refer to 2110-41 appendix A in order to use the fragmentation method described there
    - Can refer to another document

Example Registrations

- SMPTE 2109 Audio Metadata
  - ID: <to be allocated>
  - Title: ST 2109 Audio Metadata
  - Payload: SMPTE ST 2109 section 7.4 “Audio Metadata Pack Value”
  - Mapping: ST 2110-41 appendix A
- SMPTE 2110-42 tech metadata
  - ID: <to be allocated>
  - Title: Technical Metadata for ST2110-20 streams
  - Payload: ST 2110-42
  - Mapping: direct
- Joe Secret Private Magic Data
  - ID: <to be allocated>
  - Title: Joe’s secret magic data
  - Contact: Joe @ magic-is-your-friend.com

Buffer Models – Needed?

- We cannot ignore this aspect
- Metadata associated with an existing stream must be included in the buffer modeling of the associated stream
- Metadata which is not associated with an existing stream must pass a network compatibility model (Cmax model) similar to that defined in 2110-21, with the following parameters:
  - Beta = 1.1
  - T_{DRAIN} = [1 / (Beta * long-term-average-packets-per-second)]
  - C_{MAX} = 4

SDP Considerations

- -41 streams associated with a media stream should be (somehow) signaled in the SDP of the associated stream
  - Maybe just another FMTP parameter listing the ID?
  - Or ID & port offset if non-default
- -41 streams which are standalone will require their own SDP like an essence stream.

Handling the Past – and the Future

- For parameters and values from past documents (2110-x existing)
  - we'll define the key/value representations in the 2110-42 document
- For new documents or revisions after 2110-42 is at FCD ballot
  - we expect the new revision of a document (or any new document) to also specify any new keys or any new values for existing keys
- This is the approach that 2059-x took for defining alignment points and other parameters of new –vs- existing video formats.

In Parallel – a 2110-centric Application

- In addition to 2110-41 FMX Transport, a first application is proposed
  - Will be called 2110-42
  - Carriage of 2110 Technical Metadata
    - Associated sender SDP data (some / all) and additional useful tech data
- Not intended to replace IS-04 and the SDP
  - Instead provides an ”in-band” transport method
  - Useful for simple systems without NMOS registry
  - Receiver only needs the Multicast Group address, can get all necessary reception metadata from the stream on next frame.

Project Status

- FMX (2110-41)
  - Initial proposals have been discussed
  - Expansion on concepts
  - Looking at some other immediate applications
  - Document drafting to begin soon
- Tech Metadata (2110-42)
  - No activity yet

Summary

Like ST 291 metadata before it, ST 2110-41 offers a 'custom fit' method of transporting temporally-aligned metadata in 2110 systems

Can augment / replace ST 291 as applicable

Offers immense dataspace capabilities

Will enable new metadata applications not possible in SDI

Ideas? Join our work on the ST 2110 SVIP Drafting Group.

Thank-you! Questions?

Paul Briscoe, Televisionary Consulting

televisionary@teksavvy.com | +1 647 460 4466

On behalf of

Evertz Microsystems, Burlington, Ontario, Canada

Thank you to our Media Partners