SMPTE RDD 9:2013
Revision of RDD 9-2009

SMPTE REGISTERED DISCLOSURE DOCUMENT

MXF Interoperability Specification of Sony MPEG Long GOP Products

The attached document is a Registered Disclosure Document prepared by the proponent identified below. It has been examined by the appropriate SMPTE Technology Committee and is believed to contain adequate information to satisfy the objectives defined in the Scope, and to be technically consistent.

This document is NOT a Standard, Recommended Practice or Engineering Guideline, and does NOT imply a finding or representation of the Society.

Errors in this document should be reported to the proponent identified below, with a copy to eng@smpte.org. All other inquiries in respect of this document, including inquiries as to intellectual property requirements that may be attached to use of the disclosed technology, should be addressed to the proponent identified below.

Proponent contact information:

Hiroshi Nakano
Sony Corporation
4-14-1 Asahi-cho, Atsugi
Kanagawa, 243-0014
Japan

Email: hiroshi.nakano@jp.sony.com

Approved February 6, 2013

Table of Contents

Page

1 Scope

This document specifies the mapping of MPEG-2 Picture (ES), AES3 audio and ANC packets into the MXF Generic Container. The MXF files created according to the details of this specification comply with the MXF specifications defined in the normative references, with a variation with respect to one new rule introduced in SMPTE ST 377-1. In conjunction with the referenced Standards, this RDD is intended to provide sufficient information to enable a developer to construct MXF files that will be compatible with Sony MPEG-2 Long GOP products.

2 Reference Documents

Note: All references in this document to other SMPTE documents use the current numbering style (e.g. SMPTE ST 326:2000) although, during a transitional phase, the document as published (printed or PDF) may bear an older designation (such as SMPTE 326-2000). Documents with the same root number (e.g. 326) and publication year (e.g. 2000) are functionally identical.

SMPTE ST 326:2000, Television — SDTI Content Package Format (SDTI-CP)

SMPTE ST 331:2011, Element and Metadata Definitions for the SDTI-CP

SMPTE ST 377-1:2011, Material Exchange Format (MXF) — File Format Specification

Amendment 2:2012 to SMPTE ST 377-1:2011

SMPTE ST 378:2004, Television — Material Exchange Format (MXF) — Operational Pattern 1a (Single Item, Single Package)

SMPTE ST 379-1:2009, Material Exchange Format (MXF) — MXF Generic Container

SMPTE ST 381-1:2005, Television — Material Exchange Format (MXF) — Mapping MPEG Streams into the MXF Generic Container

SMPTE ST 382:2007, Material Exchange Format — Mapping AES3 and Broadcast Wave Audio into the MXF Generic Container

SMPTE ST 385:2012, Material Exchange Format (MXF) — Mapping SDTI-CP Essence and Metadata into the MXF Generic Container

SMPTE ST 400:2012, Television — SMPTE Labels Structure

SMPTE ST 436:2006, Television — MXF Mappings for VBI Lines and Ancillary Data Packets

SMPTE RP 210, Metadata Element Descriptions

SMPTE RP 224, SMPTE Labels Register

SMPTE RDD 18:2012, Acquisition Metadata Sets for Video Camera Parameters

Recommendation ITU-R BT.709-5 (04/02), Parameter Values for the HDTV Standards for Production and International Programme Exchange

3 Introduction

The MXF Generic Container is a streamable Essence Container that can be placed on any suitable transport and stored. SMPTE ST 379-1 defines the MXF Generic Container as the native Essence Container in MXF

files. SMPTE ST 381-1 defines how MPEG streams, as identified by an ISO 13818-1 stream_id value, can be mapped in the MXF Constrained Generic Container. SMPTE ST 382 defines how AES3 and Broadcast Wave Audio can be mapped in the MXF Generic Container. SMPTE ST 385 defines the System Item that is compatible with SMPTE ST 326 (SDTI-CP) and also defines how SDTI-CP essence and metadata can be used in the MXF Generic Container.

This document specifies the mapping of MPEG Picture (ES) compatible with MPEG stream and AES3 audio into the MXF Generic Container. This document also specifies the MXF file format which includes unique identifiers, Operation Pattern, Partitions, Index Table Segments and RIP. The common basic structure is described in this document

4 Outline of MXF File Structure for this Mapping

Figure 1 shows the outline of the MXF file structure. The file consists of a Header Partition, segmented Body partition(s), a Footer Partition and a Random Index pack. Picture, Sound and System Items are mapped into the Essence Container and placed in each Body Partitions. The Data Item is optional. Because of the MPEG Long GOP structure of Picture Item, segmented Index Table is used together with Random Index pack. More detailed explanation can be found in Section 8 (Application Issues).

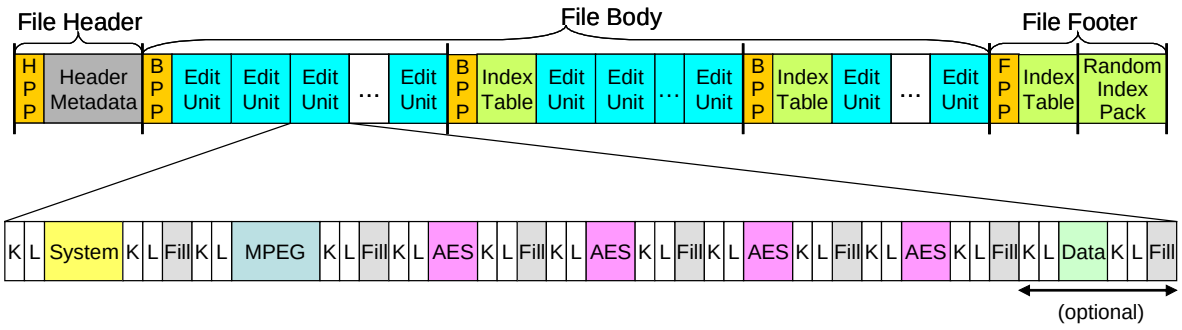


Figure 1 – Outline of MXF File

- **HPP**: Header Partition Pack
- **BPP**: Body Partition Pack
- **FPP**: Footer Partition Pack
- **Fill**[2]: The length of the KLV Fill Item should be required to align to a KAG boundary.

Some of the aspects of this structure are shown below.

- It is only necessary to include one Index Table Segment for each Body Partition period on the sender side.
- It is easy to perform the function “Play while receiving file” on the receiver side.
- It is easy to pick extract a “Partial file”.
- A list of major constraints for this file structure is given in Table 1.
- Detailed constraints are listed in Annex B.

Note: In Figure 1, the order of Items in the Content Package is System, Picture, Sound and Data Items, as described in other documents (e.g. SMPTE ST 379-1, SMPTE ST 386, SMPTE ST 387, and so on), while SMPTE ST 436 defines Data Item precedes Picture and Sound Items. This Element ordering issue is being discussed in the ST 436:2006 revision group and it is expected to be resolved in its revision.

Table 1 – Constraints for RDD 9 Stream Products

Item	Constraints
Operational Pattern	1a - *Origin* and *Duration*[1] are used to express GOP Pre-Charge and Roll-Out.
Wrapping (Interleaving)	Frame by Frame (coded order)
KAG[2] size	512
System Item	Compliant to ST 326 and ST 385, includes the Frame by Frame Timecode and UMID
Video packetization	Compliant to ST 381-1, MPEG-2 ES

GOP structure	Max 15 frames/GOP (N<=15, M<=3, open GOP / closed GOP) // 1920x1080i
	Max 12 frames/GOP (N<=12, M<=3, open GOP / closed GOP) // 1280x720p
	Variable length GOPs are permitted.

Audio sampling	48 kHz locked
Audio packetization	Compliant to ST 382, AES3, 1ch/Element (min 2 to max 8 channels)
Data Item (optional)	Compliant to ST 436, Optionally used for Ancillary packet
Timecode	System Item and Header Metadata

5 MPEG Picture Data and AES3 Data Mapping

The mapping of MPEG Picture (ES) data is as defined in SMPTE ST 381-1. The mapping of AES3 digital audio data is defined by SMPTE ST 382. This specification uses Frame Wrapping as defined by SMPTE ST 379-1. The System Item is defined by SMPTE ST 326, and mapped into the MXF by SMPTE ST 385. The order of Items in each Edit Unit is System, Picture, Sound and Data (where the Data Item is optional).

5.1 Frame Wrappings

This document requires the use of Frame Wrapping as defined by SMPTE ST 379-1, Section 5.4.1.

In the case of audio locked to video at 25 (or 50) content packages per second, each Element will contain the same number of samples, for example 1920 (or 960).

In the case of audio locked to video at 30*1000/1001 content packages per second, the number of samples in each Element will vary to maintain a correct aggregate rate. Typically the number of samples varies according to a 5-frame sequence, 1602, 1601, 1602, 1601, and 1602.

In the case of audio locked to video at 60*1000/1001 content packages per second, the number of samples in each Element will vary to maintain a correct aggregate rate. Typically the number of samples varies according to a 5-frame sequence, 801, 801, 800, 801, and 801.

The number of samples in each content package is calculated from the Length field of the surrounding KLV packet, divided by the value of the BlockAlign Property of the AES3 Audio Essence Descriptor.

In the case of audio locked to video at 24*1000/1001 content packages per second, each Element will contain the same number of samples, for example 2002.

An arrangement of System, Picture, and Sound Items in a Frame Wrapping is shown in Figure 2. It shows the case of 4 channels AES3 audio.

[1] In legacy RDD 9 files (version 1.2), the duration of the top level file package does not contain the Pre-Charge or Roll-Out as defined in SMPTE ST 378. In RDD 9 files (version 1.3), the duration of the top level file package contains the Pre-Charge but doesn't contain Roll-Out as defined in SMPTE ST 377-1.

[2] Refer to Section 8.1 Application of the KLV Fill Item.

Figure 2 – Frame Wrapping of System, Picture and Sound Elements

5.2 System Item Mapping

The System Item in each Edit Unit consists of System Metadata Pack, a Package Metadata set and Picture Metadata Set.

5.2.1 Overview of System Item

System Item is placed at the beginning of every Edit Unit and contains information on the essence item and the metadata attached to the frames, and it shall comply with SMPTE ST 385 (Mapping SDTI-CP Essence and Metadata into the MXF Generic Container).

Typical System Item consists of the following four Elements and its size is the same as one KAG size (200_h).

- System Metadata Pack — contains Package Rate, Multiple EC UL, LTC
- Package Metadata Set — contains UMID, Essence Mark (optional)
- Picture Metadata Set — contains Acquisition Metadata Sets (optional)
- Fill Item

Figure 3 shows the outline of System Item.

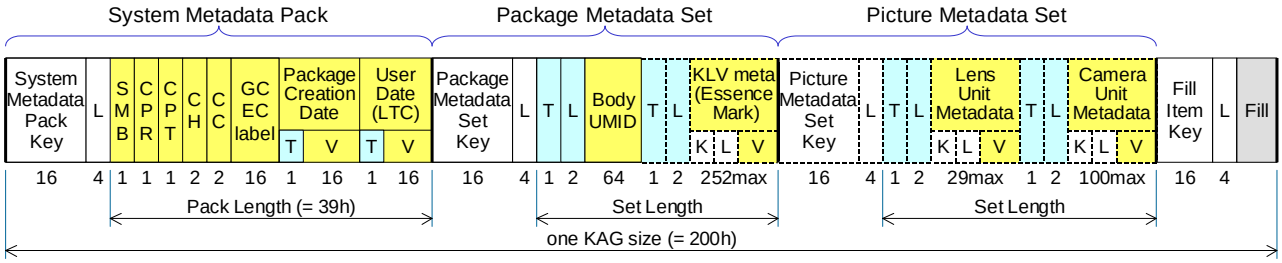


Figure 3 – Typical System Item structure

5.2.2 System Metadata Pack

The Pack Key is 06.0E.2B.34.02.05.01.01.0D.01.03.01.04.01.01.00 as defined in Table 1 of SMPTE ST 385. Length of this pack shall be fixed, i.e. 57-byte payload. Also, each Property shall be described in the provided field without tag and length. The sequence and values shall comply with SMPTE ST 326.

- System Metadata Bitmap ("SMB" in the figure) indicates the presence of metadata in the Pack, and of essence data within the Edit Unit, should be set to 0101_1100_b, when Data Item is not recorded or 0101_1110_b when Data Item is recorded.

- The value of Continuity Count ("CC" in the figure) shall be monotonically increasing within a file. It does not have to start from 0, and is back to 0000_h following full count $FFFF_h$.
- SMPTE Universal Label ("GC EC label" in the figure) shall be set to a suited MXF Generic Container label which is the same with the Essence Container Property of Multiple Descriptor Set.
- Package Creation Date should be blank. Tag ("T" in the figure) and the remains are filled with 00_h.
- LTC shall be described in the User Date column. Since it complies with SMPTE ST 331, it starts with CP-Tag 81_h and digits of Frame, Second, Minute, and Hour are placed with flags such as DF, and then Binary Group data (4 bytes) is placed, and remaining 8 bytes are filled with 0.

5.2.3 Package Metadata Set

The Set Key is 06.0E.2B.34.02.43.01.01.0D.01.03.01.04.01.02.nn as defined in Table 2 of SMPTE ST 385. The parameter nn indicates the number of Metadata Blocks within the Package Metadata Set, and the value is set to 2 when there are UMID and KLV Metadata, 1 when there is just UMID, and 0 when there is no UMID nor KLV Metadata.

Each metadata block is described with 1-byte CP-Tag and 2-byte Length field. Typical kind of metadata in this specification, shown in Figure 3, is defined as follows:

- Frame-by frame UMID should be described as the first block.
  - Extended UMID (64 bytes) should be described with CP-Tag 83_h.
  - When omitting the block, Tag and later are not described.
  - Decoders should support the cases where only Basic UMID is described or the item is blank.
- EssenceMark™ is normally described as KLV Metadata as the second block.
  - Value is less than 32 bytes, recorded with CP-Tag 88_h.
  - When omitting the item, Tag and later are not described.
- Other kind of KLV Metadata may be present with CP-Tag 88_h instead of EssenceMark™.

If there are no items in Package Metadata Set, the length field shall be zero.

5.2.4 Picture Metadata Set

The Set Key is 06.0E.2B.34.02.43.01.01.0D.01.03.01.04.01.03.nn as defined in Table 2 of SMPTE ST 385. The parameter nn indicates the number of Metadata Blocks within the Picture Metadata Set.

This Set contains Acquisition Metadata Sets specified in SMPTE RDD 18. Those Sets are described as KLV Metadata with 1-byte CP-Tag 88_h and 2-byte Length field.

When the clip does not contain Acquisition Metadata at all, the Picture Metadata Set should be omitted as a whole. When there is nothing to record as Picture Metadata item for the Edit Unit, those later than Set Key may be omitted.

5.3 Picture Item Mapping

As shown in Figure 4, the Picture Item for this mapping specification is Frame Wrapped and the data length of each Picture Element in an Edit Unit has a variable value.

This Element contains the MPEG-2 Video elementary stream, and shall comply with SMPTE ST 381-1 (Mapping MPEG Streams into the MXF Generic Container).

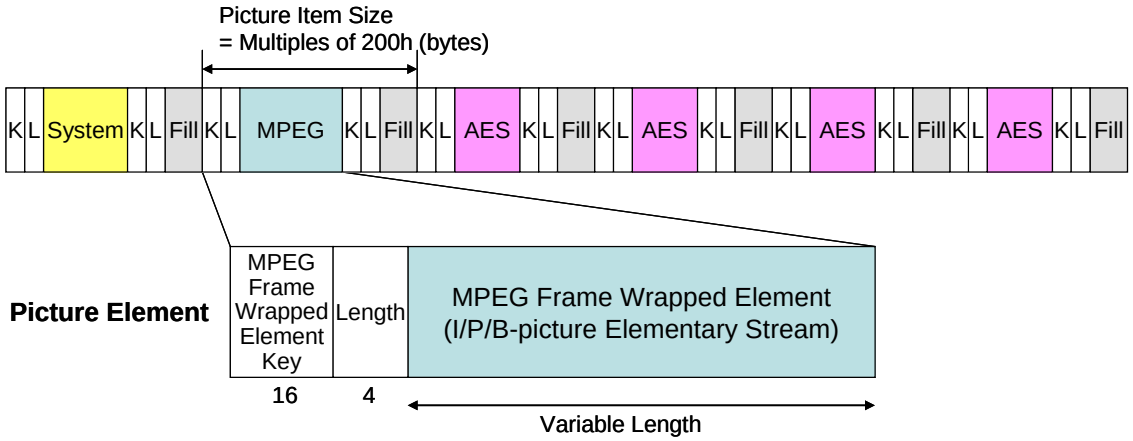


Figure 4 – Mapping of Picture Item in Edit Unit

5.3.1 MPEG Picture Element Key

The values of the first 12 bytes of the Essence Element Key are defined in SMPTE ST 379-1, MXF Generic Container Format. The Picture Element is defined by SMPTE ST 381-1, Mapping MPEG streams into the MXF Generic Container Format. This defines byte 13 with the value of 15h. The values of the last four bytes of the Essence Element Key are given in Table 2.

Table 2 – Key Value for the MPEG Picture Element

Byte No.	Description	Value (hex)	Meaning
1-12	Specified by the MXF Generic Container Specification SMPTE ST 379-1		
13	Item Type Identifier	15h	Picture Item
14	Essence Element Count	01h	Count of Elements in this Item
15	Essence Element Type	05h	Frame Wrapped Picture Element
16	Essence Element Number	00h	Element Unique Number

5.3.1.1 MPEG Picture Element Number — Byte 16

This is a number used as an index to identify this instance of the Element within the Item. It does not change within any instance in the Generic Container.

5.3.2 MPEG Picture Element Length

The length field of the KLV coded Element is 4 bytes BER long-form encoded (i.e. 83h.xx.yy.zz) for Frame wrapping.

5.3.3 MPEG Picture Element Value

The MPEG-2 Picture Element complies with SMPTE ST 381-1. The maximum frame number per GOP is fifteen frames for 1080-line systems and twelve frames for 720-line systems, and both open and closed GOP modes are supported.[3]

[3] Refer to Table 1.

5.4 AES3 Sound Item Mapping

The AES3 Sound Element complies with SMPTE ST 382, Mapping AES3 and Broadcast Wave Audio into the MXF Generic Container, as shown in Figure 5. Sound Item for this mapping specification is Frame Wrapped.

This Element contains Linear-PCM Audio data stream, and shall comply with SMPTE ST 382 (Mapping AES3 and Broadcast Wave Audio into the MXF Generic Container).

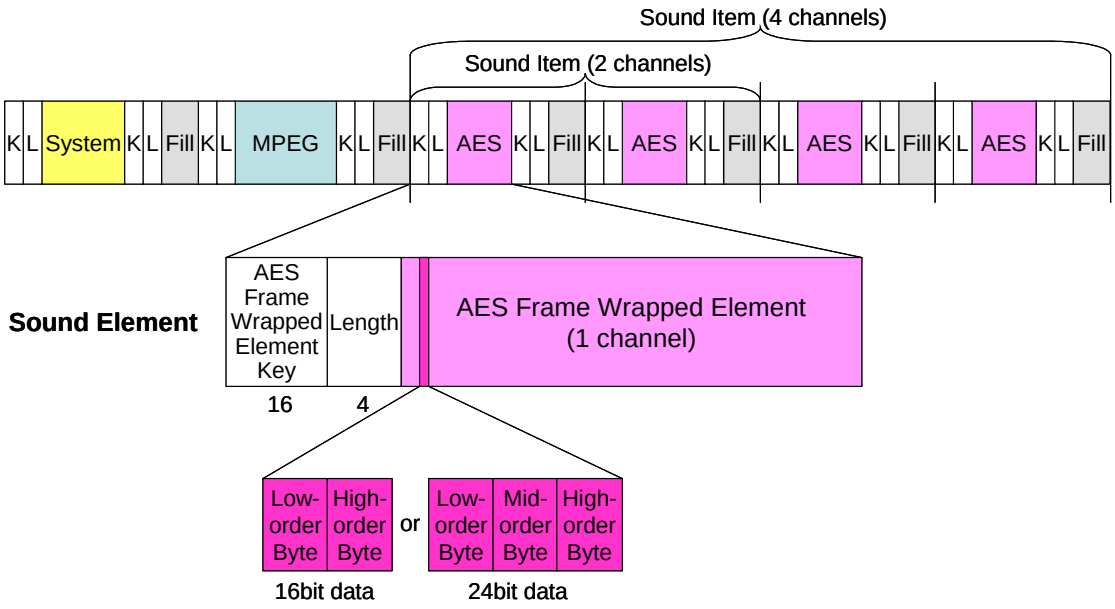


Figure 5 – Mapping of Sound Item

5.4.1 AES3 Sound Element Key

The values of the first 12 bytes of the Essence Element Key are defined in SMPTE ST 379-1, MXF Generic Container Format. The values of the last four bytes of the Essence Element Key are given in Table 3.

Table 3 – Key Value for the AES3 Sound Element

Byte No.	Description	Value (hex)	Meaning
1-12	Specified by the MXF Generic Container Specification SMPTE ST 379-1		
13	Item Type Identifier	16h	GC Sound Item
14	Essence Element Count	02, 04 or 08h	Count of Sound Elements in this Generic Container
15	Essence Element Type	03h	AES Frame Wrapped Element as listed in SMPTE RP 224
16	Essence Element Number	00, 01,02,03, 04,05,06 or 07h	Element Unique Number

5.4.1.1 AES3 Sound Element Count — Byte 14

This is a count of the number of Elements in the Sound Items in the Generic Container. The count of the Sound Elements for this mapping specification may be 2, 4 or 8 according to the count of available audio channels.

5.4.1.2 AES3 Sound Element Number — Byte 16

This is a number used as an index to identify this instance of the Element Type within the Item. Each Element within an Item has a unique value in sequence from 00h upwards as defined by SMPTE ST 379-1, which remain constant throughout any instance of a Generic Container.

5.4.2 AES3 Sound Element Length

The length field of the KLV coded Element is 4 bytes BER long-form encoded (i.e. 83h.xx.yy.zz) for Frame wrapping.

5.4.3 AES3 Sound Element Value

In a Frame Wrapped file, the Start Position of the first sample of AES3 audio data should be the same as the Start Position of the synchronized Picture Element. Note that the order of Frame Wrapped Picture Item of MPEG Picture Element is not display order, but coding order (i.e. I2 B0 B1 P5 B3 B4 P8 B6 B7…). Therefore, Sound and Picture Elements are not necessarily synchronized within each Edit Unit.

5.5 Data Item Mapping (Optional)

The Data Item contains data stream, e.g. closed captions or subtitles, and shall comply with SMPTE ST 436 (MXF Mappings for VBI Lines and Ancillary Data Packets). Figure 6 shows the ANC packets wrapped in a Data Item.

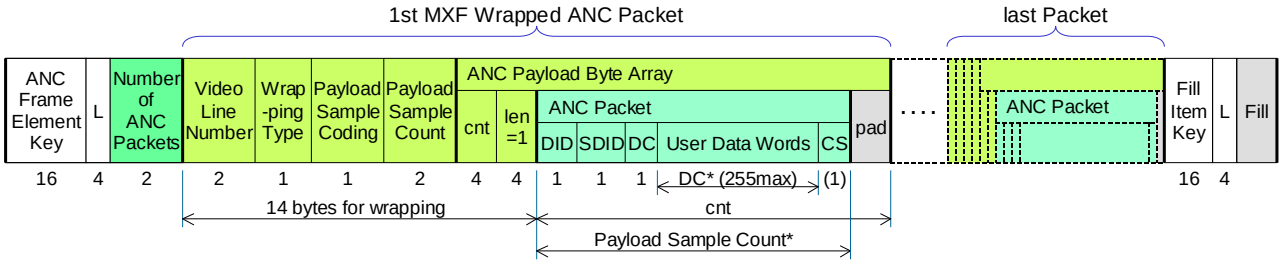


Figure 6 – Mapping of ANC Data in an Edit Unit

The Element Key is 06.0E.2B.34.01.02.01.01.0D.01.03.01.17.01.02.01 as defined in Table 6 of SMPTE ST 436. Essence Element Type is limited to Frame Wrapped ANC Data and Essence Element Count is limited to one.

Note: In SMPTE ST 436, Byte 16 of the Element Key is defined as 01h and indicates the Number (used as an Index) of this Element. RDD-9 specifies the constraint value of 01h irrespective of any revision of SMPTE ST 436.

5.6 Temporal Reordering

In the case of MPEG long GOP video, the compressed video pictures are reordered from their display order according to the MPEG specification. This reordering is applied only to the video Elements[4].

[4] Refer to Figure 15 in Section 8.3.5.

6 SMPTE Labels for Essence Container Identification

The values for the MPEG Picture Essence Container UL are given in Table 4.

Table 4 – Specification of the MPEG Picture Essence Container Label

Byte No.	Description	Value (hex)	Meaning
1-12	Specified by the MXF Generic Container Specification SMPTE ST 379-1		
13	Essence Container Kind	02h	MXF Generic Container
14	Mapping Kind	04h	MPEG ES as listed in SMPTE RP 224
15	Locally defined	60h	ISO13818-1 stream_id bits 6..0 as described in SMPTE ST 381-1
16	Locally defined	01h	MPEG ES, Frame Wrapping as defined in SMPTE RP 224

The AES3 Audio specific values for the Essence Container UL are given in Table 5.

Table 5 – Specification of the AES3 Audio Essence Container Label

Byte No.	Description	Value (hex)	Meaning
1-12	Specified by the MXF Generic Container Specification SMPTE ST 379-1		
13	Essence Container Kind	02h	MXF Generic Container
14	Mapping Kind	06h	AES-BWF as listed in SMPTE RP 224
15	Content Kind	03h	AES Frame Wrapped Element as listed in SMPTE RP 224
16	Reserved	00h	

These SMPTE Labels are the values of the ‘Essence Container’ Property used in the Partition Pack, Preface Set and appropriate File Descriptor Sets.

7 SMPTE Labels for Essence Coding Identification

The values for the Picture Essence Coding UL are given in Table 6.

Table 6 – Specification of the Picture Essence Coding Label

Byte No.	Description	Value (hex)	Meaning
1-8	Specified by the SMPTE Labels Structure Specification SMPTE ST 400		
9	Parametric	04h	Node used to define parametric data
10	Picture Essence	01h	Identifies picture essence coding
11	Picture Coding Characteristics	02h	Identifies picture coding characteristics
12	Compressed Picture Coding	02h	Identifies compressed picture coding
13	MPEG picture coding	01h	Identifies MPEG picture coding
14	MPEG-2 Profile and Label	03h 04h 05h	MPEG-2 MP@HL MPEG-2 422P@HL MPEG-2 MP@H14
15	Long GOP Coding	03h	Identifies long gop coding
16	Reserved	00h	

The Picture Essence Coding UL is used in the Generic Picture Essence Descriptor. This UL is listed in the SMPTE labels registry (SMPTE RP 224).

The Generic Sound Essence Descriptor does not include the Sound Essence Coding Property. MXF decoders should assume the value of the Property is UL of SMPTE ST 382 uncompressed sound coding (i.e. 06.0E.2B.34.04.01.01.0A.04.02.02.01.01.00.00.00).

8 Application Issues

8.1 Application of the KLV Fill Item

Within any MXF partition containing an Essence Container with this mapping specification, the KAG value defined in the Partition Pack has the value of 512 (02.00h) and the first byte of the Key of the first Element of each Item is aligned to the KLV alignment grid of that Partition.

For each Item in a Content Package, the length of the KLV Fill Item should be the minimum required to align to a KAG boundary.

Where possible, any immediately preceding Partition should align the start of each MXF Partition containing an Essence Container with this mapping specification to a byte offset that is an integer multiple of the defined KAG relative to the start of the Header Partition Pack.

The length field of the KLV Fill Item is 4 bytes, BER long-form encoded (i.e., 83h.xxh.yyh.zzh).

8.2 Application of MXF Structure and Indexing Style

This section illustrates the structures of variation of the MXF file styles, and explains how the target pictures are indexed. Applications have to handle all of the following variations.

8.2.1 Segmented Body Partition Style

This is the original basic style of SMPTE RDD 9 MXF structure. As shown in Figure 7, the MPEG Long GOP file consists of a Header Partition, segmented Body Partition(s), a Footer Partition, and a Random Index Pack. Every Partition except Header and the first Body Partitions has one Index Table Segment which carries the Index Entries indexing the Edit Units.

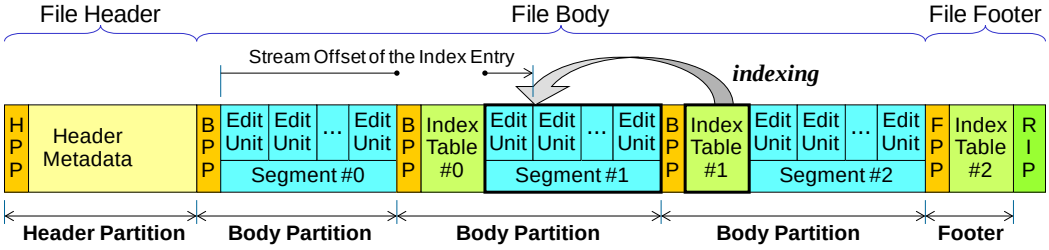


Figure 7 – Segmented Body Partition style

The purpose of partitioning is to place the Index Table Segment just after the corresponding essence data. All Index Table Segments follow Essence Container Segments that they index. Thus, while on streaming the file, decoders can use Index Table Segments for indexing without long delay.

It is recommended to have the Index Layout Properties defined in Amendment 2 to SMPTE ST 377-1:2011 in each Index Table Segment. For this style of MXF, the values for the Properties are the following.

- Index Table Segment::Single Index Location FALSE (Distributed Location)
- Index Table Segment::Single Essence Location FALSE (Distributed Location)
- Index Table Segment::Forward Index Direction FALSE (Backward)
- Preface:: is RIP present TRUE

8.2.1.1 Segmented Body Partition style with repeated Index Table

This style is an extension of the basic style specified in the previous section. As shown in Figure 8, all Index Table Segments are repeatedly placed in the Footer Partition so that applications can obtain all of them just by reading the File Footer.

Note: The last Index Table Segment (e.g., #2 shown in Figure 8) is not repeated but is placed in the Complete Index Table in the Footer. This mechanism conflicts with rule 6 of Section 11.2.1 'Index Table Segments' in SMPTE ST 377-1. In order to maintain compatibility with legacy products, RDD 9 inherits the mechanism used in its initial implementation in conformance with SMPTE ST 377:2004 which does not include the aforementioned rule.

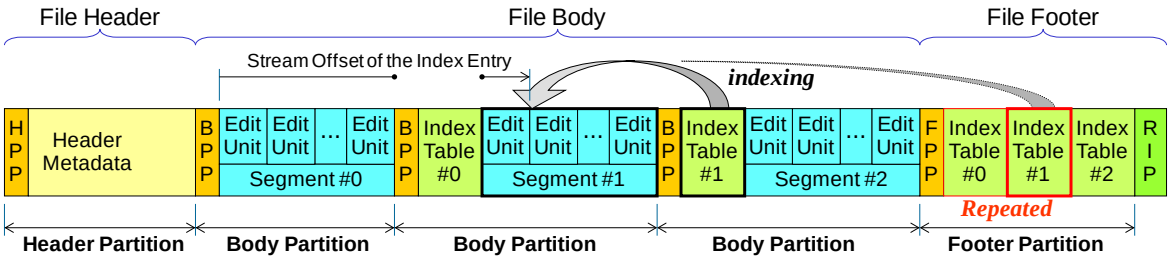


Figure 8 – Segmented Body Partition style with Repeated Index Table

The repeated Index Table Segments are completely identical with those in the Body.

It is recommended to have the following Index Layout Properties.

- Index Table Segment::Single Index Location — FALSE (Distributed Location)
- Index Table Segment::Single Essence Location — FALSE (Distributed Location)
- Index Table Segment::Forward Index Direction — FALSE (Backward)
- Preface:: is RIP present — TRUE
- Essence Container Data:: Following Index Table — TRUE (A Complete Index Table follows all Essence)

8.2.1.2 Segmented Body Partition style for on-the-fly generation

This style is a further extension of the style specified in the previous sections. If an MXF file starts to be generated whilst recording and has not yet been closed, the duration of the clip and the offset of the Footer Partition relative to the Header Partition cannot be determined. In such a situation, in the File Header the Header Metadata is incomplete, and in the File Footer the complete Header Metadata will be completed as shown in Figure 9. Decoders can refer to the complete Header Metadata in the File Footer.

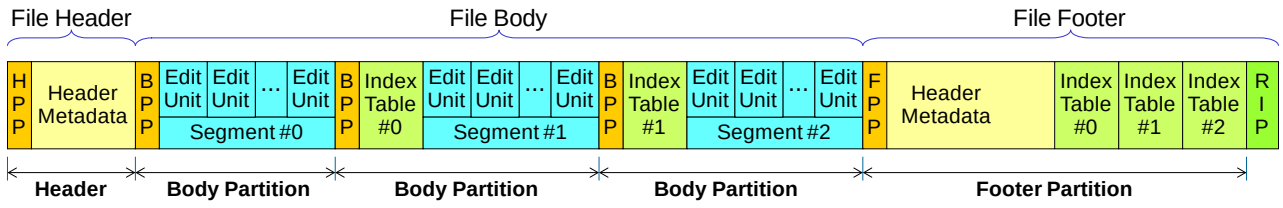


Figure 9 – Segmented Body Partition style for on-the-fly generation

In this case, Partition States, FooterPartition Property of the Partitions and the instance number generation method of Package UID of the Material Package are as follows (where *pFP* means FooterPartition Property in the Partition Packs, *imUMID* means the instance number generation method (i.e. 12th byte lower nibble of the UMID of Material Package)).

- With respect to the File Header, the Partition States are Open and Incomplete, FooterPartition Property is set to 0, the Instance Number of the Package UID of the Material Package is recommended to be Fh, and Duration values of Sequence and Source Clip are set to -1 (Distinguished Value).
- With respect to the File Body, the Partition States are Open and Complete, FooterPartition Property is set to 0.
- With respect to the File Footer, the Partition States are Closed and Complete, FooterPartition Property is set to a normal value (e.g. 3h) and Duration values are set to the conclusive value.

Note that the Instance Number value Fh means "Live stream" and 3h means “Copy number and 16-bit PRS generator”.

8.2.2 Single Body Partition Style

This style places the Index Table prior to the Essence Container, to facilitate the use of editing while file transferring. Figure 10 shows an instance of the structure style.

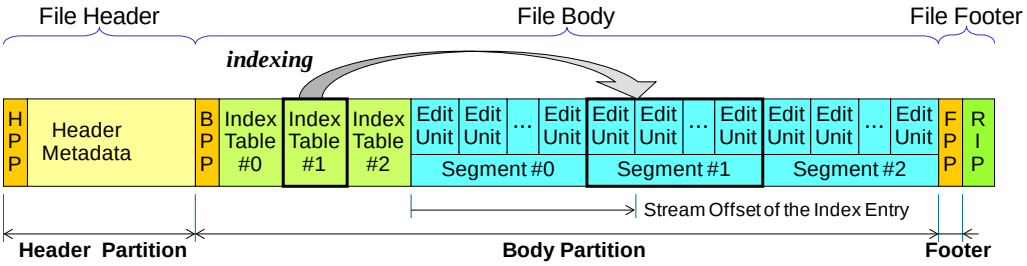


Figure 10 – Single Body Partition style

Though the File Body is not partitioned, the Index Table may be segmented, and the Random Index Pack shall be present. In this style, all information for indexing is present prior to the File Body so that quick access will be achieved for editing while file transferring.

It is recommended to attach a sufficient size (e.g. 64KB) of a Fill Item after the Header Metadata so that rewriting it afterward is possible without updating the offset values in the Partition Packs.

It is recommended to have the Index Layout Properties defined in Amendment 2 to SMPTE 377-1:2011 in each Index Table Segment. For this style of MXF, the values for the Properties are the following.

- Index Table Segment::Single Index Location TRUE (Single Location)
- Index Table Segment::Single Location TRUE (Single Location)
- Index Table Segment::Forward Index Direction TRUE (Forward)
- Preface:: is RIP present TRUE

8.2.2.1 Single Body Partition style with Repeated Index Table

As an extension of the style, Figure 11 shows Single Body Partition style with Repeated Index Table.

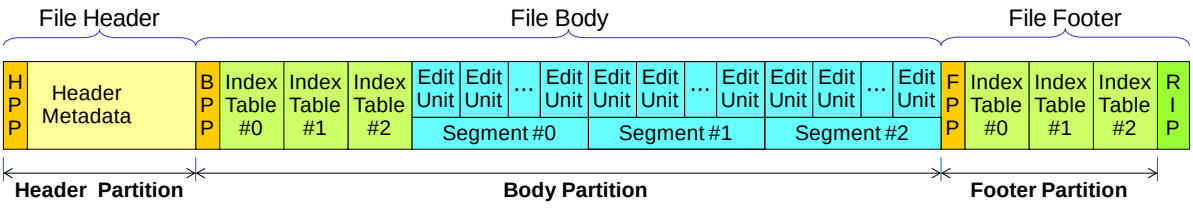


Figure 11 – Single Body Partition style with Repeated Index Table

It is recommended to have the following Index Layout Properties.

- Index Table Segment::Single Index Location TRUE (Single Location)
- Index Table Segment::Single Location TRUE (Single Location)
- Index Table Segment::Forward Index Direction TRUE (Forward)
- Preface:: is RIP present TRUE
- Essence Container Data:: Following Index Table TRUE (A Complete Index Table follows all Essence)

8.3 Application of Index Table for Frame Wrapped MPEG Picture and AES Sound Essence

8.3.1 Essence Container and Index Table

Long GOP essence requires the use of Index Table segments, as shown in Figure 12. Index Entry and Delta Entry arrays are required. One Index Table Segment is placed in each Body Partition except the first Body Partition immediately following Header Partition. The definition of the Index Table format is given in the MXF File Format Specification (SMPTE ST 377-1).

Note: The limit of 10 seconds for each Partition ensures that only a single Index Table Segment is required to index the long GOP essence in each Partition.

As noted in the previous section, each Index Table indexes the Essence Container in the previous Body Partition. Owing to this Index Table arrangement, real time creation of MXF file can be performed without buffering each Body Partition. In this mapping specification Picture Essence is VBE (Variable Bytes per Element) and Sound Essence is CBE (Constant Bytes per Element).

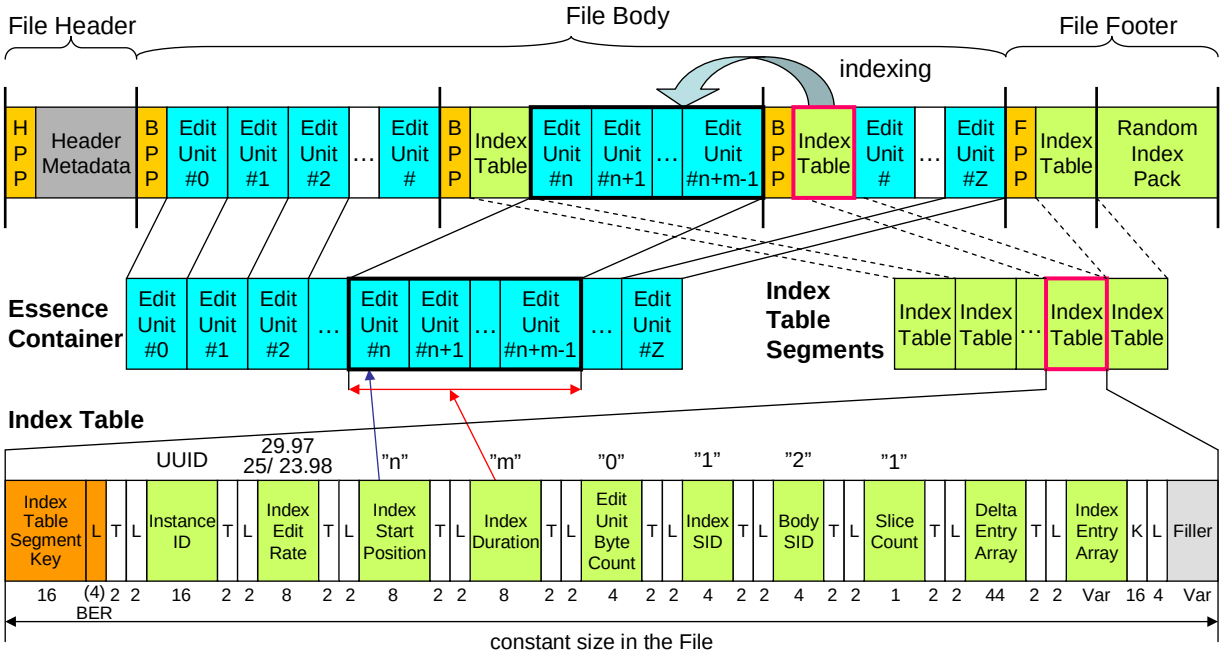


Figure 12 – Essence Container and Index Table Segments

8.3.2 Index Table Items

The Index Table Segment is constructed as shown in Table 7.

Table 7 – Index Table Segment Set Example

Item Name	Req ?	Meaning	Use for this mapping
Index Table Segment Key	Req	An Index Table Segment set	
Length	Req	Set Length	
Instance UID	Req	Unique ID of this instance	
Index Edit Rate	Req	Edit Rate copied from the tracks of the Essence Container	{60000,1001}, {50,1}, {30000,1001}, {25,1} or {24000, 1001}
Index Start Position	Req	The first editable unit indexed by this Index Table segment measured in File Package Edit Units	This sets the temporal start point for an Index Table segment
Index Duration	Req	Time duration of this table segment measured in Edit Units of the referenced Package	Combined with Start Position, allows an application to determine if this Index Table Segment spans a particular Position value
Edit Unit Byte Count	D/Req	Defines the byte count of each and every Edit Unit. A value of 0 defines the byte count of Edit Units is only given in the Index Entry Array	Edit Unit length is variable, so the value is 0.
Index SID	D/Req	Stream Identifier (SID) of Index Table	
Body SID	Req	Stream Identifier (SID) of the indexed Essence Container	
Slice Count	D/Req	Number of slices minus 1 (NSL)	1
Delta Entry Array	Opt	Map Elements onto Slices	
Index Entry Array	Req	Index from Edit Unit number to stream offset	

An Index Table provides byte offset information within an Essence Container for a given time offset from the start of that Essence Container. If the Essence Container has interleaved data within in it, then extra mechanisms are provided for finding the offsets to the individual Essence Elements once the correct time offset is located. Each Index Entry provides the byte offset within the file for a given time offset measured in Edit Units.

To locate the individual Elements within the Index Entry, the Delta Entries and Slice Offsets are required. The extent to which the Essence is indexed depends on an application. For many applications, simple indexing of the start of each Edit Unit will suffice. It is then up to the decoder within the application to find the start of each Element by parsing.

If the overall length of all the Elements in each frame is constant, then a Delta Entry Array and an "Edit Unit Byte Count" Item are sufficient to define the Index Table Segment. However, in this mapping type, Picture Element is VBE so that an Index Entry array is required.

8.3.3 Delta Entry Array

The Slice mechanism is introduced to describe all Elements' offsets in each Edit Units effectively. Each Slice starts with a number of CBE Elements and ends with a single VBE Element (or end of the Unit). The start of Slice zero corresponds to the start of the Index Entry.

The Delta Entry Array contains an entry for every indexed Element in the Generic Container. The order of the Elements in the Delta Entry Array matches the order of the Elements in the Generic Container.

Any Element that has minor sample variations (e.g. audio 5-frames sequence) is padded to a constant size if it is to be regarded as a CBE, otherwise it is a VBE Element and uses the slice mechanism. The Essence Type of the Delta Entry is determined by inspection of the Key of the Essence (e.g. Picture, Sound, Fill, etc.).

In this mapping type, there are two Slices in the Edit Unit.

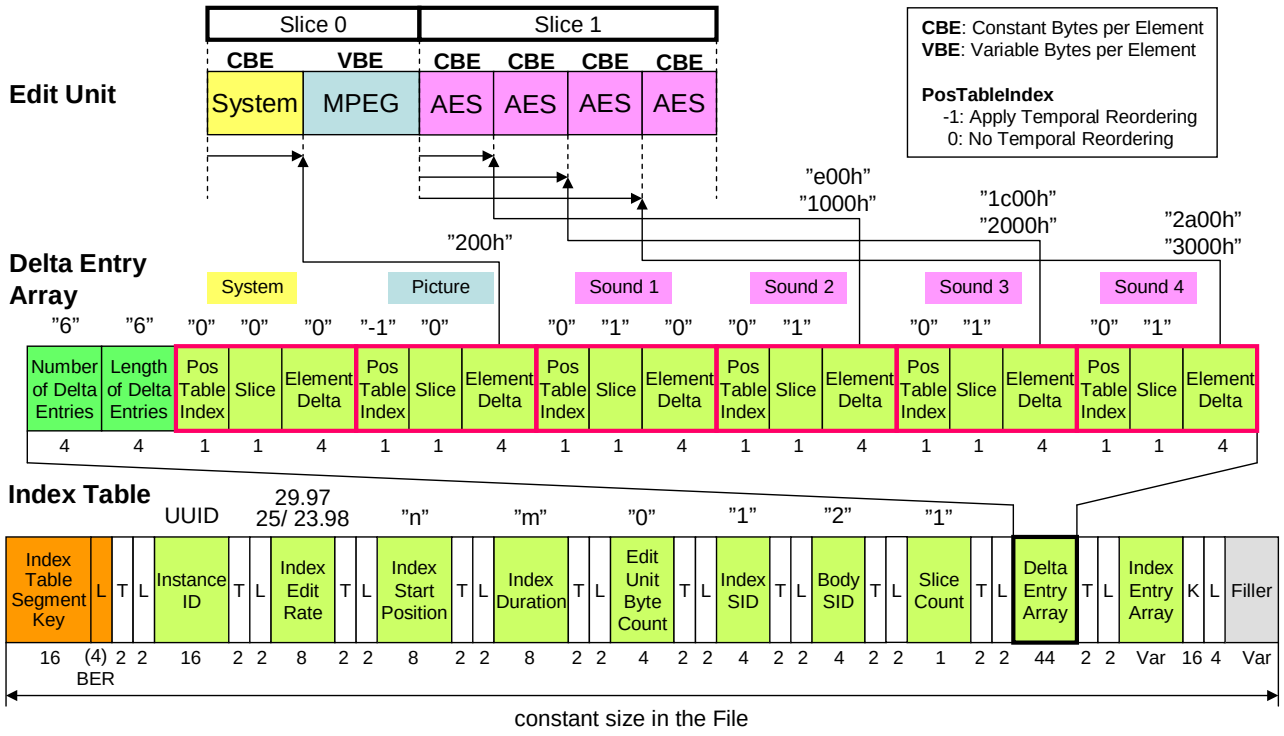


Figure 13 – Delta Entry Array

Pos Table Index is used to discover if this Element has been temporally reordered or not. If the value is zero, there is no reordering and no Temporal offsetting for this Element. If the value is negative then the Temporal Offset Property of the Index Entry Array is used to determine the difference between presentation order and storage order of the Indexed Element. In this mapping type, positive value is not used.

Slice value is the Slice number in the Edit Unit.

Element Delta is the byte offset from start of the Slice to this Element. Element Delta values include the lengths of the "KL" for each Element. The result is that each Element Delta points to the first byte of the Key in the KLV which wraps an Element.

For this mapping specification, there are 4 or 6 or 10 Delta Entries (for System, Picture and 2/4/8 Sounds) defined as follows:

The example in Table 8 below is a Delta Entry Array designed to match Figure 13, which is the case of 16-bit and 4-channels audio.

Table 8 – Structure of Delta Entry Array

	Field Name	Type	Meaning	Use for this mapping
	NDE	UInt32	Number of delta entries	6
	Length	UInt32	Length of each delta entry	6
System Delta Entry	Pos Table Index	Int8	0= No temporal reordering -1 = Apply temporal reordering	0
	Slice	UInt8	Slice number in Index Entry	0
	Element Delta	UInt32	Delta from start of slice to this Element	0
Picture Delta Entry	Pos Table Index	Int8	0= No temporal reordering -1 = Apply temporal reordering	-1 (reordered Long GOP content)
	Slice	UInt8	Slice number in Index Entry	0
	Element Delta	UInt32	Delta from start of slice to this Element	200h
Sound 1 Delta Entry	Pos Table Index	Int8	0= No temporal reordering -1 = Apply temporal reordering	0
	Slice	UInt8	Slice number in Index Entry	1
	Element Delta	UInt32	Delta from start of slice to this Element	0
Sound 2 Delta Entry	Pos Table Index	Int8	0= No temporal reordering -1 = Apply temporal reordering	0
	Slice	UInt8	Slice number in Index Entry	1
	Element Delta	UInt32	Delta from start of slice to this Element	e00h (59.94i 29.97p) 1000h (50i, 25p, 23.98p)
Sound 3 Delta Entry	Pos Table Index	Int8	0= No temporal reordering -1 = Apply temporal reordering	0
	Slice	UInt8	Slice number in Index Entry	1
	Element Delta	UInt32	Delta from start of slice to this Element	1c00h (59.94i 29.97p) 2000h (50i, 25p, 23.98p)
Sound 4 Delta Entry	Pos Table Index	Int8	0= No temporal reordering -1 = Apply temporal reordering	0
	Slice	UInt8	Slice number in Index Entry	1
	Element Delta	UInt32	Delta from start of slice to this Element	2a00h (59.94i 29.97p) 3000h (50i, 25p, 23.98p)

8.3.4 Index Entry Array

The Index Entry provides the byte offset to the start of each Edit Unit within the Essence Container, as shown in Figure 14. Each Index Entry value marks the start of the Key for the KLV packet of the first Element in each Edit Unit. The temporal distance between Index Entries within an Index Table Segment is one Edit Unit.

The Essence Type of the Index Entry is determined by inspection of the Key of the Essence (e.g. Picture, Sound, Fill, etc.).

The Index Entry also provides some Properties specific for Long GOP MPEG described in the next clause.

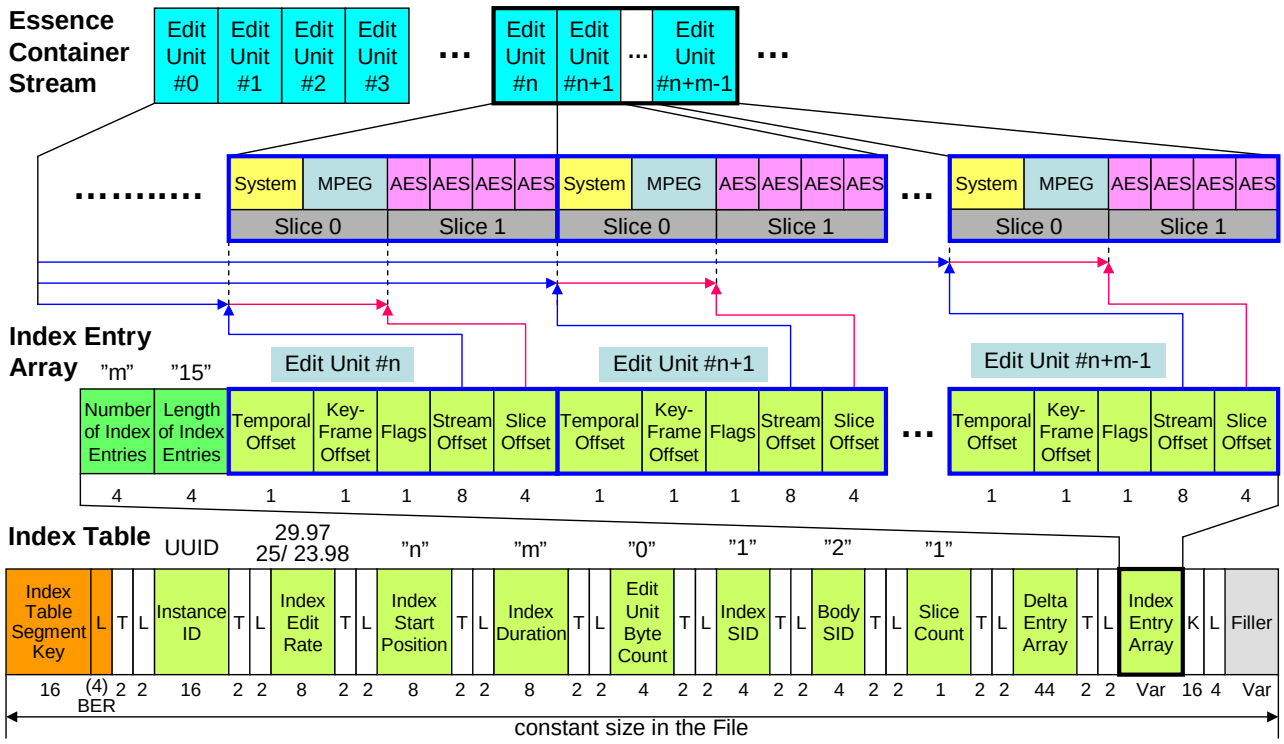


Figure 14 – Index Entry Array

Temporal Offset and **Key-Frame Offset** are Properties specific for Long GOP MPEG. In that case, the compressed video pictures may be reordered from their display order according to the MPEG specification. This reordering is applied only to the video Elements.

Flags represent Long GOP MPEG picture type.

Stream Offset is the byte offset from the beginning of the Essence Container Stream to the first KLV Element in this Edit Unit.

Slice Offset values provide the byte offset within the Edit Unit to the start of any Slices. In this mapping type, there are 2 slices, so just 1 Slice Offset value in an Index Entry.

Each Index Entry value may have zero or more Slice Offset values that provide the byte offset within the Edit Unit to the end of any Elements which are VBE.

Table 9 shows the structure of each Index Entry.

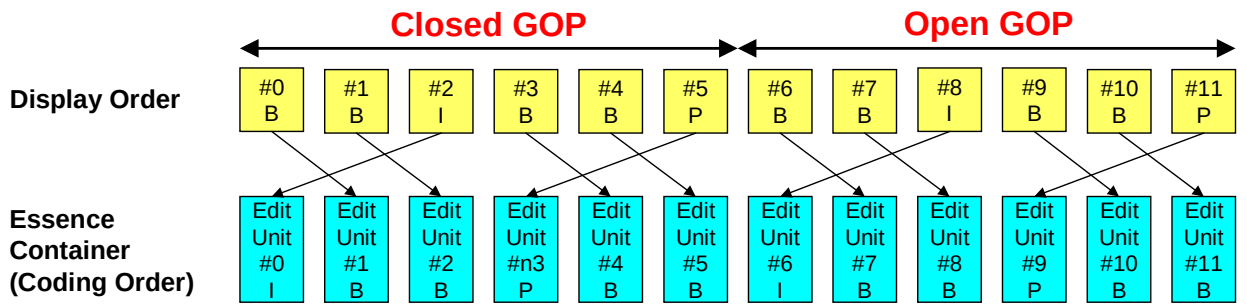
Table 9 – Index Entry Array Description

	N	Field Name	Type	Meaning	Use for this mapping
	1	NIE	UInt32	Number of index entries	Number of frames indexed by this Index Table segment
	1	Length	UInt32	Length of each index entry	15
One Index Entry for every frame	N I E	Temporal Offset	Int8	Offset in Edit Units from Display Order to Coded Order	According to picture type
		Key Offset	Int8	Offset in Edit Units to previous Key Frame. The value is zero if this is a Key frame.	According to picture type
		Flags	EditUnitFlag	Flags for this Edit Unit Bit 7: Random Access Bit 6: Sequence Header Bit 5: forward prediction flag Bit 4: backward prediction flag Bit 3: Offsets out of range Bit 2: Not used by MPEG Bits 1, 0: MPEG Picture type	Strict setting of bits 5 and 4 shall be used. 00: No prediction. This is always the case for I frames. 10: Forward prediction from previous frame. This is the case for P frames which have non zero motion vectors, or B frames which have only forward prediction motion vectors. 01: Backward prediction to future frame. This is the case for B frames which commence a closed GOP. 11: Forwards and backwards prediction. This is the general case for bi-directionally predicted B frames.
		Stream Offset	UInt64	Offset in bytes from the first KLV Element in this Edit Unit within the Essence Container Stream	Offset from the first byte of the key of the KLV for the first frame to the first byte of the Key of the KLV for the Data Element in this frame.
		Slice Offset	NSL x UInt32	The offset in bytes from the Stream Offset to the start of this slice.	Optional depending on the complexity of the VBR items. In this case there are 2 slices and NSL is set to 1

8.3.5 Setting the Properties Specific for Long GOP MPEG

There are several Properties in the Index Entry which have specific meanings for a Long GOP MPEG Index Table. These flags are correctly set according to Table 9.

Figure 15 shows an example of Temporal Offset, Key-Frame Offset, and Flags.



Temporal Offset	+1	+1	-2	+1	+1	-2	+1	+1	-2	+1	+1	-2
Key-Frame Offset	0	-1	-2	-3	-4	-5	0	-7	-8	-3	-4	-5
Flags	c0h	13h	13h	22h	33h	33h	40h	33h	33h	22h	33h	33h

Figure 15 – Long GOP MPEG Properties

Table 10 shows the Index entries for the first 6 frames of a Long GOP sequence in Figure 15.

The following example values are used in creating the table: (All lengths below are including Fill.)

- System item length is constant (= 200h bytes).
- Picture item length is variable (= 5e000h, f000h, f800h, 22000h, 10000h, and f000h in Edit Unit order).
- Sound item length is constant (= 3800h), which is the case of 4 channels.

Table 10 – Index Entry Array for the first 6 frames

	N	Field Name	Type	Value	Note
		NIE	UInt32	m	Number of Index Entries in this Index Table Segment
		Length	UInt32	15	Size of Index Entry
	Index Entry[0] contains Index data for I_2, and a Temporal offset from B_0 to Index Entry[1]				
0	Index Entry[0]	Temporal Offset	Int8	1	B_0 index data is stored in Index Entry[1]
		Key Frame Offset	Int8	0	The Key frame is IndexEntry[0]
		Flags	Edit Unit Flag	c0h	I frame – no prediction, sequence_header & random access point
		Stream Offset	UInt64	0h	Offset of the 1st Stored Edit Unit in Essence Container – I_2
		Slice Offset	UInt32	5e200h	sizeof(System Item) + sizeof(I_2 Picture Item)
	Index Entry[1] contains Index data for B_0, and a Temporal offset from B_1 to Index Entry[2]				
1	Index Entry[1]	Temporal Offset	Int8	1	B_1 index data is stored in Index Entry[2]
		Key Frame Offset	Int8	-1	The Key frame is IndexEntry[0] and 0-1= -1
		Flags	Edit Unit Flag	13h	Closed GOP B frame – backward prediction
		Stream Offset	UInt64	61a00h	Offset of the 2nd Stored Edit Unit in Essence Container – B_0
		Slice Offset	UInt32	f200h	sizeof(System Item) + sizeof(B_0 Picture Item)

	N	Field Name	Type	Value	Note
	Index Entry[2] contains Index data for B_1, and a Temporal offset from I_2 to Index Entry[0]				
2	Index Entry[2]	Temporal Offset	Int8	-2	I_2 index data is stored in Index Entry[0]
		Key Frame Offset	Int8	-2	The Key frame is IndexEntry[0] and 0-2= -2
		Flags	Edit Unit Flag	13h	Closed GOP B frame – backward prediction
		Stream Offset	UInt64	74400h	Offset of the 3rd Stored Edit Unit in Essence Container – B_1
		Slice Offset	UInt32	fa00h	sizeof(System Item) + sizeof(B_1 Picture Item)
	Index Entry[3] contains Index data for P_5, and a Temporal offset from B_3 to Index Entry[4]				
3	Index Entry[3]	Temporal Offset	Int8	1	B_3 index data is stored in Index Entry[4]
		Key Frame Offset	Int8	-3	The Key frame is IndexEntry[0] and 0-3= -3
		Flags	Edit Unit Flag	22h	P frame – forward prediction
		Stream Offset	UInt64	87600h	Offset of the 4th Stored Edit Unit in Essence Container – P_5
		Slice Offset	UInt32	22200h	sizeof(System Item) + sizeof(P_5 Picture Item)
	Index Entry[4] contains Index data for B_3, and a Temporal offset from B_4 to Index Entry[5]				
4	Index Entry[4]	Temporal Offset	Int8	1	B_4 index data is stored in Index Entry[5]
		Key Frame Offset	Int8	-4	The Key frame is IndexEntry[0] and 0-4= -4
		Flags	Edit Unit Flag	33h	B frame – bidirectional prediction
		Stream Offset	UInt64	ad000h	Offset of the 5th Stored Edit Unit in Essence Container – B_3
		Slice Offset	UInt32	10200h	sizeof(System Item) + sizeof(B_3 Picture Item)
	Index Entry[5] contains Index data for B_4, and a Temporal offset from P_5 to Index Entry[3]				
5	Index Entry[5]	Temporal Offset	Int8	-2	P_5 index data is stored in Index Entry[3]
		Key Frame Offset	Int8	-5	The Key frame is IndexEntry[0] and 0-5= -5
		Flags	Edit Unit Flag	33h	B frame – bidirectional prediction
		Stream Offset	UInt64	c0a00h	Offset of the 6th Stored Edit Unit in Essence Container – B_4
		Slice Offset	UInt32	f200h	sizeof(System Item) + sizeof(B_4 Picture Item)

8.4 Application of Random Index Pack

The Random Index Pack is a device to help find Partitions scattered throughout an MXF file, as shown in Figure 16. It is a fixed length pack which defines the Body SID and Byte Offset to the start of each Partition (i.e. the first byte of the Partition Pack Key). This pack can be used by decoders to rapidly access Index Tables and to find the partitions to which an Index Table points.

In this specification, the Random Index Pack is strongly recommended. Because of the variable length Edit Unit, it could be much difficult to find each Element without the Random Index Pack.

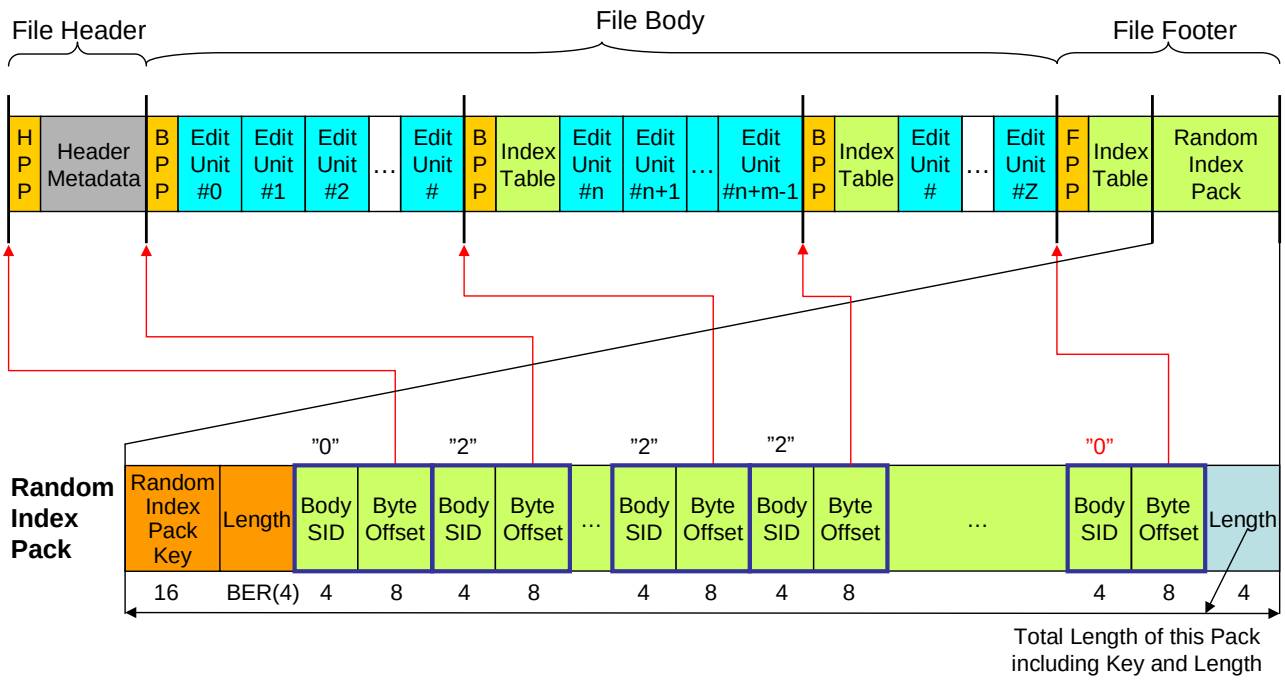


Figure 16 – Random Index Pack

Annex A UL Code List

The following is a sample list of UL codes used by SMPTE RDD 9 products.

Table A.1 – Sample of UL Code List

Header Partition Pack	Closed Complete or Open Incomplete	06 0e 2b 34 02 05 01 01 0d 01 02 01 01 02 04 00 06 0e 2b 34 02 05 01 01 0d 01 02 01 01 02 01 00
Operational Pattern	1a	06 0e 2b 34 04 01 01 01 0d 01 02 01 01 01 09 00
Essence Containers	MPEG ES Video Frame Wrapped	06 0e 2b 34 04 01 01 02 0d 01 03 01 02 04 60 01
	AES Audio Frame Wrapped	06 0e 2b 34 04 01 01 01 0d 01 03 01 02 06 03 00
Fill Item		06 0e 2b 34 01 01 01 02 03 01 02 10 01 00 00 00
Header Metadata	Primer Pack	06 0e 2b 34 02 05 01 01 0d 01 02 01 01 05 01 00
	Preface Set	06 0e 2b 34 02 53 01 01 0d 01 01 01 01 01 2f 00
	Identification Set	06 0e 2b 34 02 53 01 01 0d 01 01 01 01 01 30 00
	Content Storage Set	06 0e 2b 34 02 53 01 01 0d 01 01 01 01 01 18 00
	Essence Container Data Set	06 0e 2b 34 02 53 01 01 0d 01 01 01 01 01 23 00
	Material Package Set	06 0e 2b 34 02 53 01 01 0d 01 01 01 01 01 36 00
	File Package Set	06 0e 2b 34 02 53 01 01 0d 01 01 01 01 01 37 00
	Generic Picture Essence Descriptor Set	06 0e 2b 34 02 53 01 01 0d 01 01 01 01 01 27 00
	Picture Essence Coding (MPEG-2)	06 0e 2b 34 04 01 01 03 04 01 02 02 01 04 03 00
	Multiple Descriptor Set	06 0e 2b 34 02 53 01 01 0d 01 01 01 01 01 44 00
	Essence Container (Multiple EC UL)	06 0e 2b 34 04 01 01 03 0d 01 03 01 02 7f 01 00
	MPEG Video Descriptor Set	06 0e 2b 34 02 53 01 01 0d 01 01 01 01 01 51 00
	AES3 Audio Descriptor Set	06 0e 2b 34 02 53 01 01 0d 01 01 01 01 01 47 00
	Timecode Definition	06 0e 2b 34 04 01 01 01 01 03 02 01 01 00 00 00
	Picture Definition	06 0e 2b 34 04 01 01 01 01 03 02 02 01 00 00 00
	Sound Definition	06 0e 2b 34 04 01 01 01 01 03 02 02 02 00 00 00
	Track Set	06 0e 2b 34 02 53 01 01 0d 01 01 01 01 01 3b 00
	Sequence Set	06 0e 2b 34 02 53 01 01 0d 01 01 01 01 01 0f 00
	Source Clip Set	06 0e 2b 34 02 53 01 01 0d 01 01 01 01 01 11 00
	Timecode 12M Component Set	06 0e 2b 34 02 53 01 01 0d 01 01 01 01 01 14 00
	XML Document Text[5]	06 0e 2b 34 01 01 01 05 03 01 02 20 01 00 00 00
Body Partition Pack	Closed Complete or Open Complete	06 0e 2b 34 02 05 01 01 0d 01 02 01 01 03 04 00 06 0e 2b 34 02 05 01 01 0d 01 02 01 01 03 03 00
Index Table Segment	Index Table Segment Key	06 0e 2b 34 02 53 01 01 0d 01 02 01 01 10 01 00
System Item	System Metadata Pack	06 0e 2b 34 02 05 01 01 0d 01 03 01 04 01 01 00
	Package Metadata Set	06 0e 2b 34 02 43 01 01 0d 01 03 01 04 01 02 xx
Picture Item	MPEG Frame Wrapped Picture Element	06 0e 2b 34 01 02 01 01 0d 01 03 01 15 01 05 00
Sound Item	AES Frame Wrapped Sound Element	06 0e 2b 34 01 02 01 01 0d 01 03 01 16 04 03 0x
Data Item	ANC Frame Wrapped Data Element	06 0e 2b 34 01 02 01 01 0d 01 03 01 17 01 02 01
Footer Partition Pack	Closed Complete	06 0e 2b 34 02 05 01 01 0d 01 02 01 01 04 04 00
Random Index Pack	Random Index Pack Key	06 0e 2b 34 02 05 01 01 0d 01 02 01 01 11 01 00

[5] This key is for a stand alone KLV packet whose value contains XML document text.

There is a legacy base of MXF files that use a different Fill Item Key. This Key has the value “06.0e.2b.34.01.01.01.01.03.01.02.10.01.00.00.00”. MXF decoders should be able to recognize both Fill Item keys.

Annex B Constraints of the Conformant Implementation

Picture and Sound Essences of SMPTE RDD 9 products are categorized as follows.

- MPEG HD420: MPEG-2 MP@HL at 17.5 or 35 Mbps and MP@H-14 at 25 Mbps, with 16 bits, 2 or 4 channels uncompressed audio
- MPEG HD422: 4:2:2P@HL at 50 Mbps, with 24 bits, 8 channels uncompressed audio

This section describes the constraints on the conformant implementations of SMPTE RDD 9 products.

B.1 Structure

An SMPTE RDD 9 file shall be an MXF file which has the following structure.

- A file shall have a KAG size of 512.
- A file shall be signaled as OP-1a.
- A file shall use the MXF generic container.
- A file shall be Frame Wrapped.
- The order of Items in the Content Package shall be System, Picture, Sound and Data Items.
- A file shall include one CP System Item.
- A file shall include one or two MPEG Frame Wrapped Picture Elements.
- A file shall include one or more AES Frame Wrapped Sound Elements.
- A file shall include zero or one ANC Element.

B.2 Header and Body Partition Pack Values

The FooterPartition Property specifies the byte offset of the start of the Footer Partition relative to the start of the Header Partition. In Open Partitions, the value of FooterPartition Property in the Header or Body Partition is zero (0). In Closed Partitions, the value of FooterPartition Property in the Header or Body Partition is as defined in Section 7.1 of SMPTE ST 377-1.

B.3 Essence Descriptors

In addition to the Properties required by SMPTE ST 377-1, SMPTE ST 381-1 and SMPTE ST 382, each Essence Descriptor instance shall contain the “D/Req” / “Opt” Properties listed in Table B.1.

Table B.1 – D/Req and Opt Properties Required in RDD 9

Set	Property	Tag
Content Storage	Essence Container Data	19.02
Essence Container Data	Index SID	3F.06
File Descriptor	Linked Track ID	30.06
Generic Picture Essence Descriptor	Signal Standard, Sampled Height, Sampled Width, Sampled X Offset, Sampled Y Offset, Display Height, Display Width, Display X Offset, Display Y Offset, Stored F2 Offset, Display F2 Offset, Transfer Characteristic, Image Alignment Offset, Field Dominance, Image Start Offset, Image End Offset, Picture Essence Coding	32.15, 32.04, 32.05, 32.06, 32.07, 32.08, 32.09, 32.0A, 32.0B, 32.16, 32.17, 32.10, 32.11, 32.12, 32.13, 32.14 32.01
CDCI Picture Essence Descriptor	Color Siting, Black Ref Level, White Ref Level, Color Range, Padding Bits, Vertical Subsampling, Reversed Byte Order	33.03, 33.04, 33.05, 33.06, 33.07, 33.08, 33.0B
MPEG Video Descriptor	all Properties defined in SMPTE ST 381-1	dynamic tags
Generic Sound Essence Descriptor	Locked / Unlocked Audio Ref Level	3D.02 3D.04
AES3 Audio Essence Descriptor	Channel Status Mode Fixed Channel Status Data	3D.10 3D.11

B.4 Identification Set Value

The optional Generation UID Property of the Interchange Object Class is not be encoded in Identification Set instances as defined in SMPTE ST 377-1.

B.5 Timecode Representation in MXF Header and an Essence Container

- In Material Package, there shall be only one continuous Timecode Track.
- In File Package, there shall be only one continuous Timecode Track.
- System Item timecode may contain discontinuities.

B.6 Index Table Segments

If Index Table Segments are distributed into multiple Partitions, all Index Table Segments and Partitions except the last should have the values as shown in Table B.2. Table B.2 shows the values of the Index Duration of Index Table Segments and the Index Byte Count of Partitions, which associate with the Partition duration rounded into 10 seconds. The Index Duration of the last Index Table Segment has the remaining values. The Index Byte Count in the last Partition (i.e. Footer Partition) specifies the size of Complete Index Table if it includes a Complete Index Table. The last Index Table Segment should have a KLV Fill Item so that the Index Byte Count of the Index Table Segment is equal with the value of other Index Table Segments. If a Complete Index Table is in a Partition, all Index Table Segments which compose the Complete Index Table should have the same value of the Index Duration as shown in Table B.2 except the last Index Table Segment.

Table B.2 – Segmentation and Index Table size

Frame Rate	Index Duration [Edit Unit]	Index Byte Count [byte]	Partition Duration [sec]
23.98p	240	4096 (1000h)	10.01
25p, 50i	240	4096 (1000h)	9.6
29.97p, 59.94i	300	5120 (1400h)	10.01
50p	480	7680 (1E00h)	9.6
59.94p	600	9216 (2400h)	10.01

B.7 Random Index Pack

The Random Index Pack (RIP) shall be present.

B.8 Essence

B.8.1 System Item

An SMPTE RDD 9 file includes one CP System Item as defined in Annex B.1.

B.8.2 Picture Item

The Picture Item includes one or more MPEG Video Elements as defined in Annex B.1.

In the MPEG-2 stream:

- Sequence Header and GOP Header shall be present for each GOP.
- Sequence End Code shall not be present even at the end of the stream in a file.

B.8.3 Sound Item

The Sound Item includes one or more AES3 Elements as defined in Annex B.1.

- 5-frame sequence of audio sampling number should be 801-801-800-801-801 for 59.94fps, or 1602-1601-1602-1601-1602 for 29.97fps.

B.8.4 Data Item

The Data Item includes zero or one ANC Frame Element as defined in Annex B.1.

- Wrapping type of the ANC Frame Element shall be “VANC Frame” (001h: Interlaced or segmented progressive frame) or “VANC Progressive frame” (004h).
- Payload Sample Coding shall be “8-bit luma samples” (4)

The size of Data Item, i.e. from the first byte of the Element Key to the end of the Fill Item, shall be less than or equal to 2560 bytes for 50/59.94 progressive systems, and up to 5632 bytes for other systems.

Annex C Property Values of the Essence Descriptors

Tables C.1, C.2 and C.3 enumerate the Property values of Picture, Sound and Data Essence that specify the constraints on the conformable implementation of the RDD 9 file.

Table C.1 – An Instance of MPEG Video Descriptor

Set Key	Value	Description	Note
MPEG Video Descriptor	06.0e.2b.34.02.53.01.01. 0d.01.01.01.01.01.51.00	Defines the MPEG Video Descriptor Set	
File Descriptor			
Property	Value	Description	Note
Sample Rate	00 00 ea 60 00 00 03 e9	59.94p	
	00 00 00 32 00 00 00 01	50p	
	00 00 75 30 00 00 03 e9	59.94i, 29.97p	
	00 00 00 19 00 00 00 01	50i, 25p	
	00 00 5d c0 00 00 03 e9	23.98p	
Essence Container	06 0e 2b 34 04 01 01 02 0d 01 03 01 02 04 60 01		GC MPEG ES Frame Wrap
Generic Picture Essence Descriptor			
Property	Value	Description	Note
Signal Standard	04	HD420 1440x1080, 23.98p/25p/29.97p/50i/59.94i HD420 1440x540, 23.98p/25p/29.97p (Over Crank) HD422 1920x1080, 23.98p/25p/29.97p/50i/59.94i	SMPTE ST 274 (1920x1080)
	05	HD420 1280x720, 50p/59.94p HD422 1280x720, 23.98p/25p/29.97p/50p/59.94p	SMPTE ST 296 (1280x720P)
Frame Layout	00	23.98p/25p/29.97p/50p/59.94p	FULL_FRAME (progressive)
	01	50i/59.94i	SEPARATE_FIELDS (interlaced)
Stored Width	00 00 05 00	HD420 1280x720, 23.98p/50p/59.94p HD422 1280x720, 23.98p/25p/29.97p/50p/59.94p	1280
	00 00 05 a0	HD420 1440x1080, 23.98p/25p/29.97p/50i/59.94i HD420 1440x540, 23.98p/25p/29.97p (Over Crank)	1440
	00 00 07 80	HD422 1920x1080, 23.98p/25p/29.97p/50i/59.94i	1920
Stored Height	00 00 02 20	HD420 1440x1080, 50i/59.94i HD420 1440x540, 23.98p/25p/29.97p (Over Crank) HD422 1920x1080, 50i/59.94i	544
	00 00 02 d0	HD420 1280x720, 23.98p/50p/59.94p HD422 1280x720, 23.98p/25p/29.97p/50p/59.94p	720
	00 00 04 40	HD420 1440x1080, 23.98p/25p/29.97p HD422 1920x1080, 23.98p/25p/29.97p/50p/59.94p	1088
StoredF2Offset	00 00 00 00	Stored and Sampled Rectangles are first field upper.	Valid with an interlaced signal.
Sampled Width	00 00 05 00	HD420 1280x720, 23.98p/50p/59.94p HD422 1280x720, 23.98p/25p/29.97p/50p/59.94p	1280
	00 00 05 a0	HD420 1440x1080, 23.98p/25p/29.97p/50i/59.94i HD420 1440x540, 23.98p/25p/29.97p (Over Crank)	1440
	00 00 07 80	HD422 1920x1080, 23.98p/25p/29.97p/50i/59.94i	1920
Sampled Height	00 00 02 1c	HD420 1440x1080, 50i/59.94i HD420 1440x540, 23.98p/25p/29.97p (Over Crank) HD422 1920x1080, 50i/59.94i	540
	00 00 02 d0	HD420 1280x720, 23.98p/50p/59.94p HD422 1280x720, 23.98p/25p/29.97p/50p/59.94p	720

	00 00 04 38	HD420 1440x1080, 23.98p/25p/29.97p HD422 1920x1080, 23.98p/25p/29.97p/50p/59.94p	1080
SampledXOffset	00 00 00 00	The left edge of the Stored and Sampled Rectangles is the same.	
SampledYOffset	00 00 00 00	The upper edge of the Stored and Sampled Rectangles is the same.	
Display Height	00 00 02 1c	HD420 1440x1080, 50i/59.94i HD420 1440x540, 23.98p/25p/29.97p (Over Crank) HD422 1920x1080, 50i/59.94i	540
	00 00 02 d0	HD420 1280x720, 23.98p/50p/59.94p HD422 1280x720, 23.98p/25p/29.97p/50p/59.94p	720
	00 00 04 38	HD420 1440x1080, 23.98p/25p/29.97p HD422 1920x1080, 23.98p/25p/29.97p/50p/59.94p	1080
Display Width	00 00 05 00	HD420 1280x720, 23.98p/50p/59.94p HD422 1280x720, 23.98p/25p/29.97p/50p/59.94p	1280
	00 00 05 a0	HD420 1440x1080, 23.98p/25p/29.97p/50i/59.94i HD420 1440x540, 23.98p/25p/29.97p (Over Crank)	1440
	00 00 07 80	HD422 1920x1080, 23.98p/25p/29.97p/50i/59.94i	1920
DisplayXOffset	00 00 00 00	The left edges of the Sampled and Display Rectangles are the same.	
DisplayYOffset	00 00 00 00	The upper edges of the Sampled and Display Rectangles are the same.	
DisplayF2Offset	00 00 00 00	The Sampled and Display Rectangles are first field upper.	Valid with an interlaced signal.
Aspect Ratio	00 00 00 10 00 00 00 09	The horizontal to vertical aspect ratio of the source image is 16:9.	
Video Line Map	00 00 00 02 00 00 00 04 00 00 00 15 00 00 02 48	HD420 1440x1080, 50i/59.94i HD422 1920x1080, 50i/59.94i	(21, 584) : Interlace
	00 00 00 02 00 00 00 04 00 00 00 1a 00 00 00 00	HD420 1280x720, 23.98p/50p/59.94p HD422 1280x720, 23.98p/25p/29.97p/50p/59.94p	(26) : Progressive 720p
	00 00 00 02 00 00 00 04 00 00 00 2a 00 00 00 00	HD420 1440x1080, 23.98p/25p/29.97p HD420 1440x540, 23.98p/25p/29.97p (Over Crank) HD422 1920x1080, 23.98p/25p/29.97p/50p/59.94p	(42) : Progressive 1080p
Transfer Characteristic	06 0e 2b 34 04 01 01 01 04 01 01 01 01 02 00 00	The color primaries, color matrix and gamma equation conform to ITU-R BT.709. The registered value is in SMPTE RP224.	ITU-R BT.709 transfer characteristic
Image Start Offset	00 00 00 00	Unused bytes from the start of the Stored Data to the start of the Stored Rectangle are zero.	
Image End Offset	00 00 00 00	Unused bytes from the end of the Stored Rectangle to the end of the Stored Data are zero.	
Field Dominance	01	The first field is first in temporal order	Valid with an interlaced signal.
Picture Essence Coding	06 0e 2b 34 04 01 01 03 04 01 02 02 01 03 03 00	HD420 1440x1080, 17.5M/35M HD420 1280x720, 25M/35M	MP@HL Long GOP
	06 0e 2b 34 04 01 01 08 04 01 02 02 01 05 03 00	HD420 1440x1080, 25M	MP@H-14 Long GOP
	06 0e 2b 34 04 01 01 03 04 01 02 02 01 04 03 00	HD422 1920x1080, 50M HD422 1280x720, 50M	422P@HL Long GOP
	06 0e 2b 34 04 01 01 03 0e 06 41 02 01 03 03 01	HD420 1440x540, 8.75M/17.5M (Over Crank of HD420 1440x1080, 17.5M/35M)	MP@HL Long GOP Over Crank
	06 0e 2b 34 04 01 01 03 0e 06 41 02 01 05 03 01	HD420 1440x540, 12.5M (Over Crank of HD420 1440x1080, 25M)	MP@H-14 Long GOP Over Crank
	06 0e 2b 34 04 01 01 03 0e 06 41 02 01 04 03 01	HD422 1920x540, 25M (Over Crank of HD422 1920x1080, 50M)	422P@HL Long GOP Over Crank
CDCI (Color Difference Component Image) Picture Essence Descriptor			

Property	Value	Description	Note
Component Depth	00 00 00 08	Active bits per sample are eight bits.	8-bit
Horizontal Subsampling	00 00 00 02	Color difference component sampling is 4:2:0 or 4:2:2.	4:2:0, 4:2:2
Vertical Subsampling	00 00 00 01	HD422	4:2:2
	00 00 00 02	HD420	4:2:0
Color Siting	00	HD422	4:2:2
	06	HD420 (MXF Version 1.3)	
	ff	HD420 (MXF Version 1.2)	
Padding Bits	00 00	Bits to round up each pixel to stored size are zero.	
Black Ref Level	00 00 00 10	Digital code for reference black is 16.	16
White Ref Level	00 00 00 eb	Digital code for reference white is 235.	235
Color Range	00 00 00 e1	Digital code of the range for color difference samples is 225.	225 (8 bit)
MPEG Video Descriptor			
Property	Value	Description	Note
Single Sequence	00	There are differences among sequence headers within the Essence stream.	FALSE
	01	All sequence headers in the Essence stream are identical.	TRUE
Constant B frames	00	The number of B frames is not always constant.	FALSE
	01	The number of B frames is always constant.	TRUE
Coded Content Type	01	Progressive: 23.98p/25p/29.97p/50p/59.94p	Progressive
	02	Interlaced: 50i/59.94i	Interlaced
Low Delay	00	Low delay mode is not used in the sequence.	FALSE
Closed GOP	00	The closed_gop flag is not always set in all GOP Headers.	Open GOP
	01	The closed_gop flag is set in all GOP Headers.	Closed GOP
Identical GOP	00	Every GOP in the sequence is not always constructed the same.	FALSE
	01	Every GOP in the sequence is constructed the same.	TRUE
Max GOP	00 0c	Max 12 frames / GOP: 23.98p/25p/50i/50p/59.94p	
	00 0f	Max 15 frames / GOP: 29.97p/59.94i	
B Picture Count	00 02	The maximum number of B pictures between P or I frames is two.	
Bit Rate	01 0b 07 60	HD420 1440x1080, 17.5M	
	01 7d 78 40	HD420 1440x1080, 25M HD420 1280x720, 25M	
	02 16 0e c0	HD420 1440x1080, 35M HD420 1280x720, 35M	
	02 fa f0 80	HD422 1920x1080, 50M HD422 1280x720, 50M	
	00 85 83 b0	HD420 1440x540, 8.75M (Over Crank of HD420 1440x1080, 17.5M)	
	00 be bc 20	HD420 1440x540, 12.5M (Over Crank of HD420 1440x1080, 25M)	
	01 0b 07 60	HD420 1440x540, 17.5M (Over Crank of HD420 1440x1080, 35M)	

Profile And Level	44	MP@HL: HD420 1440x1080, 17.5M/35M HD420 1280x720, 25M/35M HD420 1440x540, 8.75M/17.5M (Over Crank of HD420 1440x1080, 17.5M/35M)	MP@HL
	46	MP@H-14: HD420 1440x1080, 25M HD420 1440x540, 12.5M (Over Crank of HD420 1440x1080, 25M)	MP@H-14
	82	422P@HL: HD422 1920x1080, 50M HD422 1280x720, 50M HD422 1920x540, 25M (Over Crank of HD422 1920x1080, 50M)	422P@HL

Table C.2 – An Instance of AES3 Audio Essence Descriptor

Set Key	Value	Description	Note
AES3 Audio Essence Descriptor	06.0e.2b.34.02.53.01.01. 0d.01.01.01.01.01.47.00	Defines the AES3 Audio Essence Descriptor Set (a collection of Parametric metadata)	
File Descriptor			
Property	Value	Description	Note
Sample Rate	00 00 ea 60 00 00 03 e9	59.94p	in MXF version1.2
	00 00 00 32 00 00 00 01	50p	in MXF version1.2
	00 00 75 30 00 00 03 e9	59.94i, 29.97p	in MXF version1.2
	00 00 00 19 00 00 00 01	50i, 25p	in MXF version1.2
	00 00 5d c0 00 00 03 e9	23.98p	in MXF version1.2
	00 00 bb 80 00 00 00 01	48000, 1	in MXF version1.3
Essence Container	06 0e 2b 34 04 01 01 02 0d 01 03 01 02 06 03 00		GC AES Frame Wrap
Generic Sound Essence Descriptor			
Property	Value	Description	Note
Audio sampling rate	00 00 bb 80 00 00 00 01	48000, 1	
Locked/Unlocked	01	Number of samples per frame is locked to video	
Audio Ref Level	00	Number of dBm for 0VU	
Channel Count	00 00 00 01	Number of sound channels	
Quantization bits	00 00 00 10	HD420	16 bits
	00 00 00 18	HD422	24 bits
Wave Audio Essence Descriptor			
Property	Value	Description	Note
Block Align	00 02	HD420	16 bits
	00 03	HD422	24 bits
Average Bytes Per Second	00 01 77 00	HD420	96k bytes/sec (16bit)
	00 02 32 80	HD422	144k bytes/sec (24bit)
AES3 Audio Essence Descriptor			
Property	Value	Description	Note
Channel Status Mode	00 00 00 01 00 00 00 01 01		MINIMUM mode
Fixed Channel Status Data	00 00 00 01 00 00 00 18 85 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00		Professional use Linear PCM No emphasis 48KHz sampling

	00 00 00 01 00 00 00 18 87 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00		Professional use Non-Linear PCM No emphasis 48KHz sampling

Table C.3 – An Instance of ANC Packets Descriptor

Set Key	Value	Description	Note
ANC Data Descriptor	06.0e.2b.34.02.53.01.01. 0d.01.01.01.01.01.5C.00	Defines the ANC Data Descriptor Set	
File Descriptor			
Property	Value	Description	Note
Sample Rate Essence Container	00 00 ea 60 00 00 03 e9	59.94p	
	00 00 00 32 00 00 00 01	50p	
	00 00 75 30 00 00 03 e9	59.94i, 29.97p	
	00 00 00 19 00 00 00 01	50i, 25p	
	00 00 5d c0 00 00 03 e9	23.98p	
Essence Container	06 0e 2b 34 04 01 01 09 0d 01 03 01 02 0e 00 00		GC Generic ANC data with an undefined payload
Generic Data Essence Descriptor			
No Property			
ANC Data Descriptor			
No Property			