

SMPTE STANDARD

for Television —

Material Exchange Format (MXF) — MXF Generic Container



Page 1 of 13 pages

Table of contents

- 1 Scope
- 2 Normative references
- 3 Glossary of acronyms, terms and data types
- 4 Introduction
- 5 MXF generic container format
- 6 System item coding
- 7 Picture, sound, data and compound item coding
- 8 SMPTE label for essence container identification
- Annex A Bibliography

1 Scope

This standard specifies the format of the MXF generic container. The MXF generic container is the native essence container of the material exchange format (MXF) file body. The MXF generic container is defined for the interchange of streamable audio-visual material.

This standard defines the data structure at the signal interfaces of networks or storage media. This standard does not define internal storage formats for MXF compliant devices.

Appropriate essence and metadata payloads that can be mapped into the MXF generic container are defined in associated documents.

The MXF specification includes operation pattern specifications that may define restrictions on the way in which this essence container type should be implemented. The reader is advised to carefully study the appropriate operational pattern document for compliance to a defined implementation.

2 Normative references

The following standards contain provisions which, through reference in this text, constitute provisions of this standard. At the time of publication, the editions indicated were valid. All standards are subject to revision and parties to agreements based on this standard are encouraged to investigate the possibility of applying the most recent of the standards indicated below.

SMPTE 336M-2001, Television — Data Coding Protocol using Key-Length-Value

SMPTE 377M-2004, Television — Material Exchange Format (MXF) — File Format Specification

3 Glossary of acronyms, terms and data types

The general glossary of acronyms, terms and data types used in the MXF specification is given in SMPTE 377M. It is not repeated here to avoid any divergence of meaning.

3.1 Acronyms used in this standard

CP: Content package – a generic term for a grouping of some combination of system, picture, sound, data and / or compound items.

GC: Generic container.

3.2 Terms used in this standard

Picture essence: A general term for all types of picture essence including video, still images, graphics, etc.

Sound essence: A general term for all types of sound essence including audio, MIDI, sampled data, etc.

Data essence: A general term for all types of data essence including tele-text, closed caption data, etc.

Compound essence: A general term for essence that contains an indivisible mixture of different essence types.

4 Introduction

The MXF generic container is a streamable data container that can be placed on any suitable transport and potentially stored. The concept of this container was based on the work done by the EBU/SMPTE Task Force in the Wrappers and Metadata sub-group. The MXF generic container defined in this standard is fully compatible with the work of the EBU/SMPTE Task Force Report.

The MXF generic container format is intended for inclusion into a MXF (Material eXchange Format) file as an essence container. This standard defines the MXF generic container for use in an MXF file body.

NOTES

1 A streamable data container is designed to allow the audio-visual material to be continuously decoded through mechanisms such as interleaving essence components with stream-based metadata.

2 The Task Force report defines: “Content is composed of **content packages**, which in turn are composed of **content items**, which are further composed of **content elements**”. These content packages are convenient groupings of the various Items where each Item is a group of similar element types. Although the term “content package” is also used in the SDTI-CP specification, the generic container content package is a more generalized arrangement that retains backwards compatibility with the SDTI-CP content package.

The MXF generic container comprises a contiguous sequence of content packages, each of which has up to five basic components known as items.

- A system item is a group of up to 127 metadata or control data elements related to the container itself may be related to the elements in the other four items below.
- A picture item is a group of up to 127 picture essence elements. Each essence element in a picture item should contain a predominance of picture essence although the element may contain metadata and other ancillary essence.
- A sound item is a group of up to 127 sound essence elements. Each essence element in a sound item should contain a predominance of sound essence although the element may contain metadata and other ancillary essence.

- A data item is a group of up to 127 data essence elements. Each essence element in a data item should contain a predominance of data essence although the element may contain metadata and other ancillary essence.
- A compound item is a group of up to 127 compound essence elements. Each essence element in a compound Item should contain a mixture of essentially indivisible essence and metadata components that, as a group, do not match the intent of the picture, sound or data items.

The premise for the MXF generic container format is that of a general purpose essence data and metadata container for the containment of many different kinds of essence and metadata elements into a single entity by interleaving the data streams in a defined and time-synchronous manner (typically over a 1-frame duration). Associated mapping documents define the essence data and metadata elements that can be placed in the container. Some mapping documents may define complete mappings for an entire content package while others may simply define mapping of metadata or essence data into an element.

The MXF generic container defined by this document complies with the requirements for essence containers defined in the MXF file format specification.

5 MXF generic container format

The MXF generic container comprises a contiguous sequence of one or more content packages as illustrated in figure 1. The content packages may be of constant or variable length depending on the application. The example in figure 1 shows a generic container with content packages of variable length.

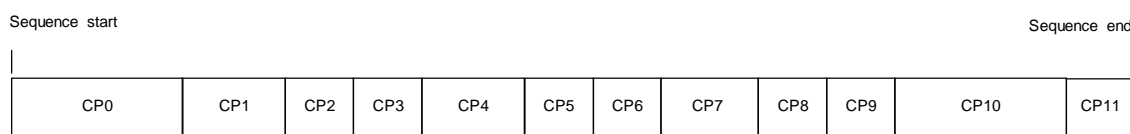


Figure 1 – Generic container as a contiguous sequence of content packages

5.1 Content package

Each content package represents the essence and metadata elements interleaved over a defined duration (typically 1 picture frame) and is constructed of up to five Items which are the system, picture, sound, data and compound items.

Picture and sound items are essentially carrying the primary video and audio elements that are often routed to specialist storage or processing equipment. The data item is used to carry data-centric elements such as subtitles and teletext data and is frequently created, processed and stored on computer media. The system item provides services for each content package through metadata elements such as time stamps, metadata for essence elements in the other Items and, optionally, downstream control data elements. All Items in the MXF generic package are optional and their presence depends on the requirements of the associated mapping document.

The system item in any content package can carry up to 127 metadata or control data elements. The picture, sound, data and compound items in any content package can each comprise up to 127 essence elements. It is a key feature that each content package contains the associated contents of one defined duration starting with a system item (where present) and optionally containing picture, sound, data or compound items. Figure 2 shows the layered structure of each generic container content package.

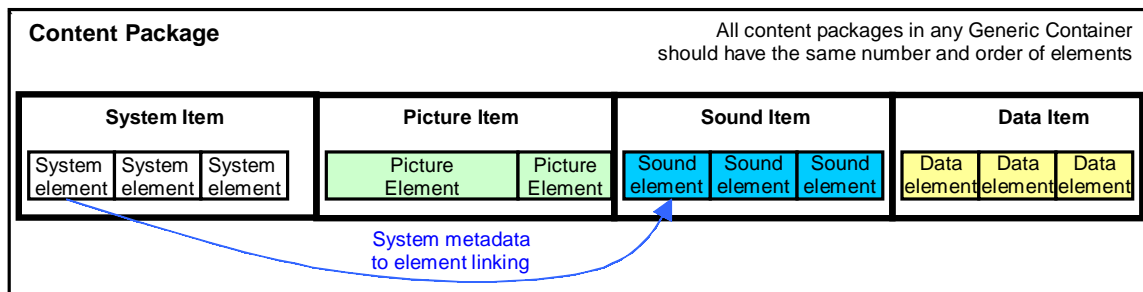


Figure 2 – Logical structure of items and elements in a content package

NOTE – The metadata contained in the system Item may include local links which associate any metadata item uniquely with its corresponding essence element. In many cases, metadata is embedded into each essence element (e.g., in the case of MPEG-2, the metadata is embedded in the various headers of the MPEG-2 essence bitstream). The external metadata link is provided to provide metadata in addition to the essence bitstream. The system metadata can, for example, be a partial or whole extraction of embedded metadata extracted at the data packing process to provide quick access to key metadata without a requirement to re-parse the essence bitstream. The metadata can also be temporally sensitive metadata such as time-code information or camera coordinates.

5.2 Content package structure

There shall be only one instance of an item of any type in any one content package (i.e., you cannot have two sound items in one content package). If a system item is present, then every content package should start with that system item and the package is completed with any of the other Items as needed. If a system item is not present, then the content package should comprise only one element (which may be in a picture, sound, data or compound item).

NOTE – In content packages with no system item, there may be no mechanism to identify the first item or element in the content package, hence only one element is recommended.

5.3 Content package mappings

The MXF generic container offers two forms of essence mapping; frame-based mapping and clip-based mapping.

5.3.1 Frame-based mappings

In frame-based mappings, there may be one or more content packages in the essence container. If there is only one content package, it shall represent the contents of a single frame or field. The frame or field duration is defined by the sample unit of the primary essence component carried by the content package. This may be a video frame, a video field, an audio frame (as a 192-sample block), a motion picture frame or any other value that represents the basic sample unit of the primary essence component.

An example arrangement of system, picture, sound and data items in a frame-based mapping is shown in figure 3.

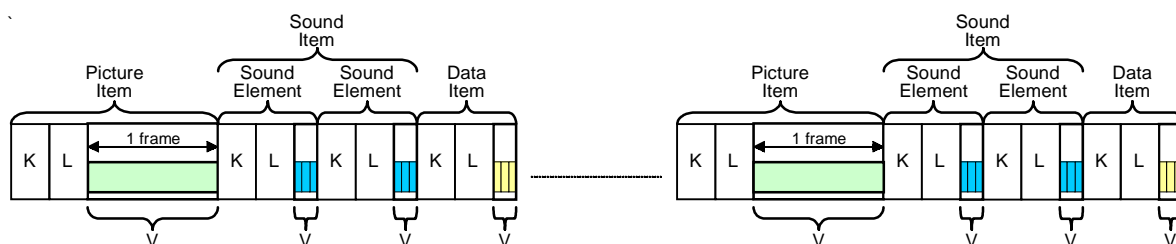


Figure 3 – Example of frame wrapping items and elements in the generic container

Following any system item, each element of a content package is added in a sequence which should follow the MXF encoding behaviour defined in 5.4. As a consequence, the sequence of Items within a content package should remain consistent over any sequence of content packages within an essence container. An example arrangement of the system Item followed by one essence element in the picture item, two essence elements in the sound item and one data element in the data item is shown in figure 4.

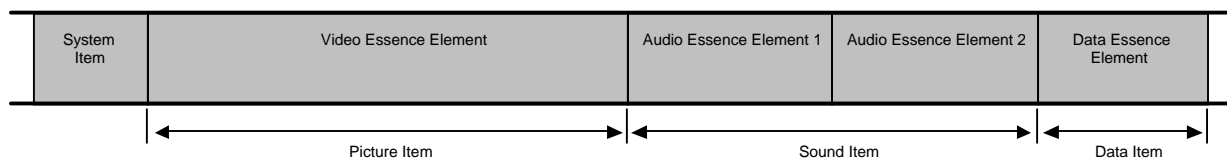


Figure 4 – Example arrangement of system item, video, audio and data essence elements in a content package

5.3.2 Clip-based mappings

In clip-based mappings, there shall be only one content package in the essence container. The duration of the clip may be one or more frames as defined in 5.3.1.

An example arrangement of system, picture, sound and data items in a clip-based mapping is shown in figure 5.

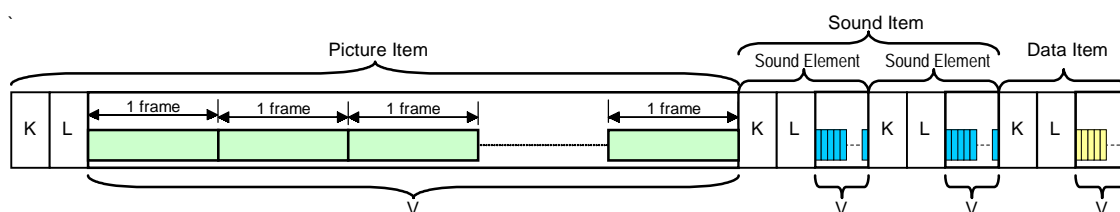


Figure 5 – Example of clip wrapping items and elements in the generic container

5.4 Default MXF encoder behavior

The generic container is intended to wrap a wide variety of essence types. Individual mapping documents may constrain MXF encoder behavior so that consistent MXF files are created. This section provides overall guidelines which are intended to give default behavior which should be followed in the absence of more specific rules. This default behavior is intended to aid interoperability by ensuring MXF encoder implementations create consistent MXF files. It is recognized that the default behavior may not be possible in all circumstances:

- 1 each CP should have a constant number of elements;
- 2 the order of elements in the CP should not change;
- 3 every element in the CP should have the same duration;
- 4 each CP should have one element which is the primary timebase (usually the video);
- 5 each CP should have a duration which is an integer multiple of the atomic size of the underlying essence of the primary timebase (video frame, audio frame, audio block etc.);
- 6 synchronized elements should be grouped in the same CP

5.4.1 Multiple elements from the same track in a content package

The generic container may be used for some Interleaved streams where the primary timebase element duration (e.g., a video frame) is not the same as the other elements (e.g., compressed audio frames). The default MXF encoder behavior above may lead to a situation where there are multiple elements from the same track in a content package.

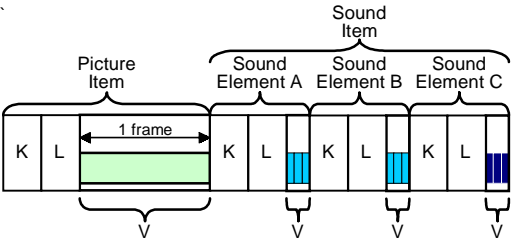


Figure 6 – Multiple separate elements from the same track

The picture in figure 6 shows three sound elements in a content package. Sound elements A and B are from the same sound track, and sound element C is from a different sound track. MXF decoders are able to determine which sound elements are grouped together from the last 4 bytes of the element key. Sound elements A and B will have identical keys, whereas sound element C will have a different key. This is the default MXF behavior where each element should have equal duration.

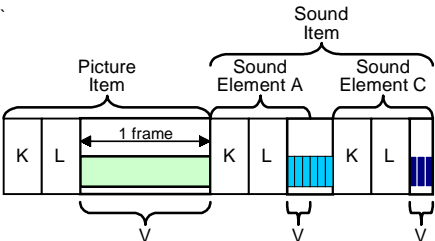


Figure 7 – Combination of elements from the same track

An alternative approach is to combine elements A and B into a single large element A within the content package as shown in figure 7. Although this may be useful in certain applications, its use is not recommended as it is not an easily reversible process and may lead to downstream problems when MXF files are being manipulated.

5.5 KLV coding structure

The system item metadata elements, together with all picture, sound, data and compound essence elements, are each coded using the key-length-value (KLV) coding protocol according to SMPTE 336M.

KLV coding allows a decoder to identify each component by its 16-byte Universal label 'key' and skip any component it cannot recognize using the 'length' value to continue decoding data types with recognized 'key' values.

The general data structure of each KLV packet is shown diagrammatically in figure 8.

Key (16 byte Universal Label)	Length (variable length BER coded)	Value (variable length)
--	---	----------------------------

Figure 8 – Data structure of each KLV packet

SMPTE 336M defines the structure of the Key value and the options for the format of the length field. Section 6 defines the KLV coding of the metadata elements in the system item, while section 7 defines the KLV coding of essence elements in the picture, sound, data and compound items.

5.5.1 KLV length field

The KLV length field and its application shall comply with SMPTE 377M.

6 System item coding

The system item contains metadata elements which describes the operation of the content package in various modes and provides key metadata items related to the whole package. It can include metadata elements linked to essence elements in the picture, sound, data and compound items. The system Item may include optional downstream control elements. This section defines the details of the system Item coding and format.

6.1 System item components

The system item shall comprise a contiguous sequence of up to 127 KLV packets where each packet comprises metadata elements or control data elements for support of different aspects of the content package. Depending on the requirements, each packet may be coded as a fixed-length (FL) pack, a variable-length (VL) pack or a local set according to SMPTE 336M.

Figure 9 illustrates the system item data structure.

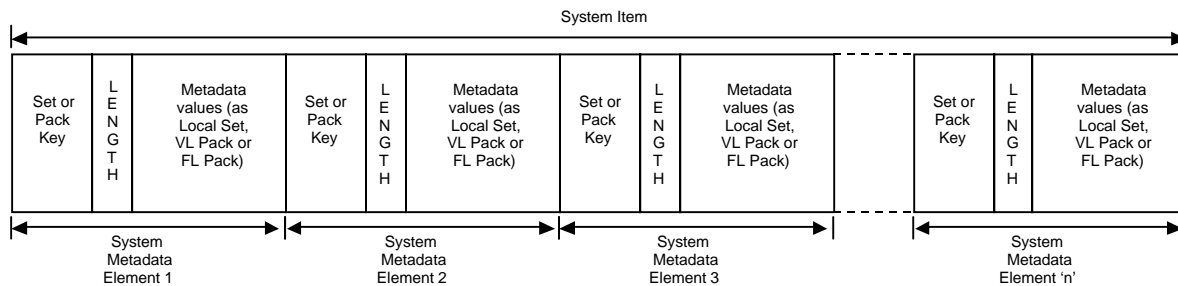


Figure 9 – System item as a sequence of metadata elements

6.2 System item metadata element definitions

6.2.1 Pack and set keys

The key of a system item metadata element shall be as defined in table 1.

Table 1 – Specification of the set or pack key for a system item metadata element

Byte No.	Description	Value (hex)	Meaning
1	Object Identifier	06h	
2	Label size	0Eh	
3	Designator	2Bh	ISO, ORG
4	Designator	34h	SMPTE
5	Registry Category Designator	02h	Sets & packs
6	Registry Designator	xxh (See SMPTE 336M)	Fixed-length Pack, Variable-length Pack or Local Set as required
7	Structure Designator	01h	Sets & Packs Registry
8	Version Number	vvh	Registry Version at the point of registration of this Key
9	Item Designator	0Dh	Organisationally Registered
10	Organisation	01h	AAF Association
11	Application	03h	MXF Generic Container Keys
12	Structure Version	01h	MXF-GC Version 1
13	Item Type Identifier	04h 14h	CP-Compatible System Item (see SMPTE 326M) GC-Compatible System Item
14	System Scheme Identifier	xxh	See appropriate System Item definition document
15	Metadata or Control Element Identifier	yyh	See appropriate System Item definition document
16	Reserved for use by metadata Element	zzh	

The choice of fixed-length pack, variable-length pack or local set coding is defined by an associated metadata element or control data element specification.

Where a system item is present in the content package, the first metadata element shall set the metadata element identifier (byte 15) to '01h'. No other system item element shall precede this element.

NOTE – This allows decoders to identify an unambiguous starting point of a content package.

6.2.2 Element value

The value of a system item metadata element or control data element shall be a sequence of metadata properties coded as a local set, a variable-length pack or a fixed-length pack as defined by byte 6 of the key value. The definition of the properties in a metadata element or control data element can be found in associated documents.

6.2.3 System item status

If the system item is required for a generic container mapping document, then the mapping document shall specify the following:

- Bytes 13 to 16 of the element key value (see table 1);
- Either the definition of the element value or the reference document where the definition of the element value can be found;
- If the element is describing an essence element, then the method of linking shall be defined.

6.3 Element to track relationship

Each metadata element or control data element in a system Item may be described by a track in a MXF header metadata package. This track will have an associated track number which shall be derived as follows:

The track number is a UInt32 value comprising bytes: 'A.B.C.D' (most significant byte first). These byte values shall be assigned as follows:

A = Byte 13 of the element key value;
 B = Byte 14 of the element key value;
 C = Byte 15 of the element key value;
 D = Byte 16 of the element key value.

This technique ensures that each track referenced by the header metadata package will have a unique number that is directly linked to the element key value.

Figure 10 explains that the track number item in a track of the file package has the same values as bytes 13-16 of the key used to KLV wrap the picture element in the stream. The key shall be unique within a partition and shall remain constant for each element within a generic container. This mechanism shall be used for all MXF generic container mappings. To identify the correct partition in which to resolve the track number item, the BodySID mechanism shall be used as detailed in the MXF format specification.

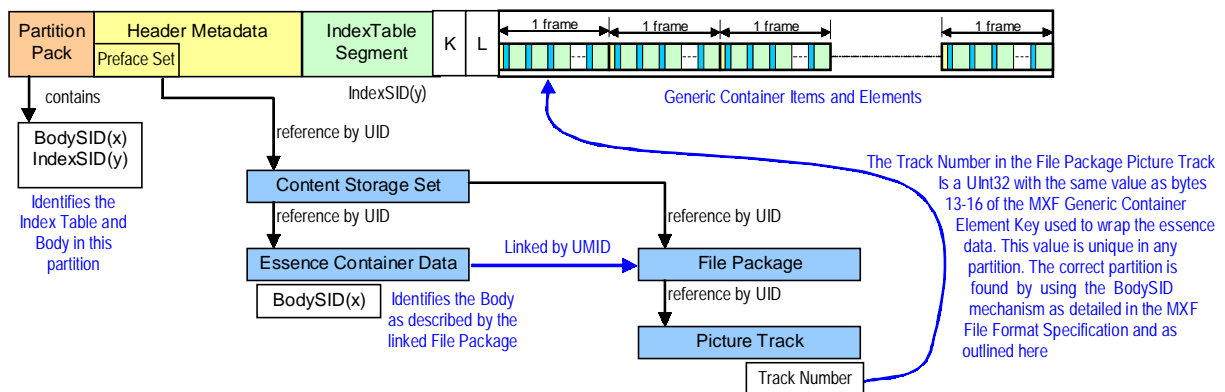


Figure 10 – Linking of the generic container element key to the track number item

7 Picture, sound, data and compound item coding

Where picture, sound, data and compound items are present in a content package, each essence element shall be coded as a single KLV item according to SMPTE 336M. The content package is thus a sequence of KLV packets contained within the duration of the content package.

Each KLV coded element starts with a 16-byte element key value to identify the type of element and the Item to which the element belongs, followed by a length field and completed by the element value itself.

7.1 Essence element key

The essence element key is defined in table 2.

Table 2 – Key values for picture, sound, data and compound elements

Byte No.	Description	Value (hex)	Meaning
1	Object Identifier	06h	
2	Label size	0Eh	
3	Designator	2Bh	ISO, ORG
4	Designator	34h	SMPTE
5	Registry Category Designator	01h	Dictionaries
6	Registry Designator	02h	Essence Dictionary
7	Structure Designator	01h	Dictionary Standard
8	Version Number	vvh	Registry Version at the point of registration of this Key
9	Item Designator	0Dh	Organisationally Registered
10	Organisation	01h	AAF Association
11	Application	03h	MXF Generic Container Keys
12	Structure Version	01h	MXF-GC Version 1
13	Item Type Identifier	05h 06h 07h 15h 16h 17h 18h	CP Picture (SMPTE 326M) CP Sound (SMPTE 326M) CP Data (SMPTE 326M) GC Picture GC Sound GC Data GC Compound
14	Essence Element Count	xxh	See below
15	Essence Element Type	yyh	See below
16	Essence Element Number	zzh	See below

Byte 13 of the key value identifies the Item to which the element belongs and the correct item value shall be entered.

Values for byte 13 of '05h', '06h' and '07h' shall be reserved for essence elements defined in SMPTE 331M.

NOTE – Item type '07h' is known in SMPTE 326M as an auxiliary item, but is identical to the data item of the MXF generic container.

Values of '15h', '16h', '17h' and '18h' shall be reserved for essence elements defined in other MXF documents which specifically define essence mappings for the MXF generic container.

Byte 14 of the key value shall be used to define the number of essence elements in this Item of the content package. A single essence element within an Item will result an essence element count value of '01'h. For a given essence element, this value shall be constant within the entire generic container (even when new elements are added). This is to maintain track linking as detailed in 7.3.

NOTE – This byte is the same value as the Item Header (which is the essence element count limited to the range 01h~7Fh) in SMPTE 326M.

Byte 15 shall be the value of the element type as defined by either SMPTE 331M or an associated document. Element type values shall be constrained to the range 01h~7Fh. For a given essence element, this value shall be constant within the entire generic container (even when new elements are added). This is to maintain track linking as detailed in 7.3.

Byte 16 shall be used to define the value of the element number in the range 00h~7Fh. It shall be set by the encoder to be unique amongst the elements in any one Item. In most cases, the element number will be increment by one for each new essence element in sequence within an item. For a given essence element, this value shall be constant within the entire generic container (even when new elements are added). This is to maintain track linking as detailed in 7.3.

7.2 Essence element value

The picture, sound, data or compound element value is as defined in an associated mapping document.

7.2.1 Picture, sound, data and compound item status

If the Item is required for a mapping document, then the mapping document shall specify the following:

- Bytes 13 to 16 of the essence element key value (see table 2);
- Either the definition of essence element value or the reference document where the definition of the essence element value can be found.

7.3 Element to track relationship

Each essence element in a picture, sound, data or compound Item shall be described by a track in a MXF header metadata package. This track will have an associated track number which shall be derived as follows:

The track number is a UInt32 value comprising bytes: 'A.B.C.D' (most significant byte first). These byte values shall be assigned as follows:

A = Byte 13 of the element key value;
 B = Byte 14 of the element key value;
 C = Byte 15 of the element key value;
 D = Byte 16 of the element key value.

This technique ensures that each track referenced by the header metadata package will have a unique number that is directly linked to the element key value. The method of linking the track number in the header metadata to the essence element key is identical to that described in 6.3 for metadata.

NOTE – If there is one video element in the picture item and there are two sound elements in the sound item, then the header metadata package will contain one picture track and two sound tracks. The value of the track number item in each track will be linked to the essence element keys in the generic container essence using the mechanism described above.

8 SMPTE label for essence container identification

The common framework for a SMPTE label that identifies the essence container payload shall be as defined in table 3.

Table 3 – Specification of the essence container label

Byte No.	Description	Value (hex)	Meaning
1	Object Identifier	06h	
2	Label size	0Eh	
3	Designator	2Bh	ISO, ORG
4	Designator	34h	SMPTE
5	Registry Category Designator	04h	Labels
6	Registry Designator	01h	Labels Registry
7	Structure Designator	01h	Labels Structure
8	Version Number	vvh	Version of the Registry
9	Item Designator	0Dh	Organizationally Registered
10	Organization	01h	AAF Association
11	Application	03h	Essence containers
12	Structure Version	01h	Version 1
13	Essence container Kind	02h 01h	MXF Generic Container Experimental MXF Generic Container for prototyping only
14	Mapping Kind	xxh	Defines the kind of mapping
15~16	Locally defined	yyh	Defined by the application specification

NOTE – Byte 14 is defined by the appropriate GC mapping document and will have a value in the range '01'h - '7F'h.

This SMPTE label is the individual 'essence container' property used in the partition pack, in the preface set and in the appropriate file descriptor.

This SMPTE label may also be added to the system item where the definition of the system item allows.

A value of 01h in byte 13 is provided to document generic container mappings which were experimental. This value shall not be used.

Annex A (informative)

Bibliography

ANSI/SMPTE 298M-1997, Television — Universal Labels for Unique Identification of Digital Data

SMPTE 305.2M-2000, Television — Serial Data Transport Interface (SDTI)

SMPTE 326M-2000, Television: SDTI Content package Format (SDTI-CP)

SMPTE 331M-2004, Television — Element and Metadata Definitions for the SDTI-CP

SMPTE RP 210, Metadata Dictionary Registry of Metadata Element Descriptions

SMPTE RP 224, SMPTE Labels Registry

SMPTE EG 41-2004, Material Exchange Format (MXF) Engineering Guideline

EBU/SMPTE Task Force for Harmonized Standards for the Exchange of Programme Material as Bitstreams, Final Report: Analyses and Results, Sept 1998

SMPTE Journal, Vol 109, No 3, March 2000, pp 205..210, "A Tutorial on SDTI-CP"