SMPTE ST 385:2012
Revision of SMPTE 385M-2004

SMPTE STANDARD

Material Exchange Format (MXF) — Mapping SDTI-CP Essence and Metadata into the MXF Generic Container

Table of Contents

Approved
June 5, 2012

Foreword

SMPTE (the Society of Motion Picture and Television Engineers) is an internationally-recognized standards developing organization. Headquartered and incorporated in the United States of America, SMPTE has members in over 80 countries on six continents. SMPTE's Engineering Documents, including Standards, Recommended Practices, and Engineering Guidelines, are prepared by SMPTE's Technology Committees. Participation in these Committees is open to all with a bona fide interest in their work. SMPTE cooperates closely with other standards-developing organizations, including ISO, IEC and ITU.

SMPTE Engineering Documents are drafted in accordance with the rules given in Part XIII of its Operations Manual.

SMPTE ST 385 was prepared by Technology Committee 31FS.

Intellectual Property

At the time of publication no notice had been received by SMPTE claiming patent rights essential to the implementation of this Standard. However, attention is drawn to the possibility that some of the elements of this document may be the subject of patent rights. SMPTE shall not be held responsible for identifying any or all such patent rights.

Introduction

This section is entirely informative and does not form an integral part of this Engineering Document.

The MXF Generic Container is a stream-able data container structure 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 subgroup. In March 1998, a SMPTE ad-hoc group was formed to study the Content Package (CP) format based on an initial strawman proposal. The CP format was developed from the ground up to carry picture, sound and data essence together with metadata in a structured manner primarily over the SDTI (SMPTE ST 305) transport and is known as SDTI-CP. SDTI-CP comprises two SMPTE engineering documents:

– SMPTE ST 326 (SDTI-CP format) which defines the container format as applied to the SDTI; and

– SMPTE ST 331 (SDTI-CP element and metadata definitions) which defines the data that can be placed into the SDTI-CP container.

The MXF Generic Container is an abstraction of the CP data structure that can be made fully compatible with SDTI-CP by using the System Item data defined in this standard together with the appropriate essence mapping document where it is compliant to SMPTE ST 331.

This standard specifies a version of the System Item in the MXF Generic Container that is compatible with SMPTE ST 326 noting that Auxiliary Items and elements in SMPTE ST 326 are synonymous with data items and elements in the MXF Generic Container.

This standard also specifies how to map SMPTE ST 331 SDTI-CP elements and items into the MXF Generic Container.

Note that in tables within this standard, some properties are SMPTE Universal Labels (ULs). A list of appropriate values for these properties is provided in SMPTE RP 224.

1 Scope

This standard specifies the mapping of essence and metadata conforming to SMPTE ST 326 to the MXF Generic Container. The MXF Generic Container is the native Essence Container for use in an MXF file body.

This standard specifies an SDTI-CP compatible System Item as a sequence of metadata elements in the MXF Generic Container. The standard then describes how the SDTI-CP System Item, conforming to SMPTE ST 326, can be mapped into the SDTI-CP compatible System Item of the MXF Generic Container as defined by both SMPTE ST 379-1 and SMPTE ST 379-2.

This standard also specifies how SDTI-CP picture, sound and Auxiliary Items conforming to SMPTE ST 326 can be mapped into the MXF Generic Container. This includes mapping of SDTI-CP elements conforming to SMPTE ST 326.

Where required, this standard also defines how the MXF Generic Container containing the appropriate metadata and Essence Elements can be mapped into SMPTE ST 326.

2 Conformance Notation

Normative text is text that describes elements of the design that are indispensable or contains the conformance language keywords: "shall", "should", or "may". Informative text is text that is potentially helpful to the user, but not indispensable, and can be removed, changed, or added editorially without affecting interoperability. Informative text does not contain any conformance keywords.

All text in this document is, by default, normative, except: the Introduction, any section explicitly labeled as "Informative" or individual paragraphs that start with "Note:"

The keywords "shall" and "shall not" indicate requirements strictly to be followed in order to conform to the document and from which no deviation is permitted.

The keywords, "should" and "should not" indicate that, among several possibilities, one is recommended as particularly suitable, without mentioning or excluding others; or that a certain course of action is preferred but not necessarily required; or that (in the negative form) a certain possibility or course of action is deprecated but not prohibited.

The keywords "may" and "need not" indicate courses of action permissible within the limits of the document.

The keyword “reserved” indicates a provision that is not defined at this time, shall not be used, and may be defined in the future. The keyword “forbidden” indicates “reserved” and in addition indicates that the provision will never be defined in the future.

A conformant implementation according to this document is one that includes all mandatory provisions ("shall") and, if implemented, all recommended provisions ("should") as described. A conformant implementation need not implement optional provisions ("may") and need not implement them as described.

Unless otherwise specified, the order of precedence of the types of normative information in this document shall be as follows: Normative prose shall be the authoritative definition; Tables shall be next; followed by formal languages; then figures; and then any other language forms.

3 Normative References

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 326M-2000). Documents with the same root number (e.g. 326) and publication year (e.g. 2000) are functionally identical.

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 editions of the standards indicated below.

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 336:2007, Data Encoding Protocol Using Key-Length-Value

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

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

SMPTE ST 379-2:2010, Material Exchange File (MXF) — MXF Constrained Generic Container

4 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 ST 377-1. Supplementary glossaries of terms and acronyms used in the Generic Container are given in SMPTE ST 379-1 and SMPTE ST 379-2. They are not repeated here to avoid any divergence of meaning.

4.1 Acronyms used in this Standard

SMB: System Metadata bitmap

CPR: Content Package rate

CPT: Content Package type

CH: Channel handle

CC: Continuity count

4.2 Terms used in this Standard

Little-endian: Any multi-byte value with the least significant byte first.

5 SDTI-CP Compatible System Item

The SDTI-CP compatible System Item contains metadata 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 linked to Essence Elements in the picture, sound and data items. Finally, the SDTI-CP compatible System Item includes an optional downstream control element for future possible extension.

This section defines the data structure of the SDTI-CP compatible System Item.

The SDTI-CP compatible System Item shall be a combination of fixed-length pack data and local set data elements, KLV coded according to SMPTE ST 336. Throughout the remainder of this document, the SDTI-CP compatible System Item will be called simply the “System Item”.

5.1 System Item Elements

The System Item shall be coded as a sequence of up to six KLV packets where each packet comprises metadata elements for different aspects of the Content Package.

Figure 1 illustrates the System Item data structure.

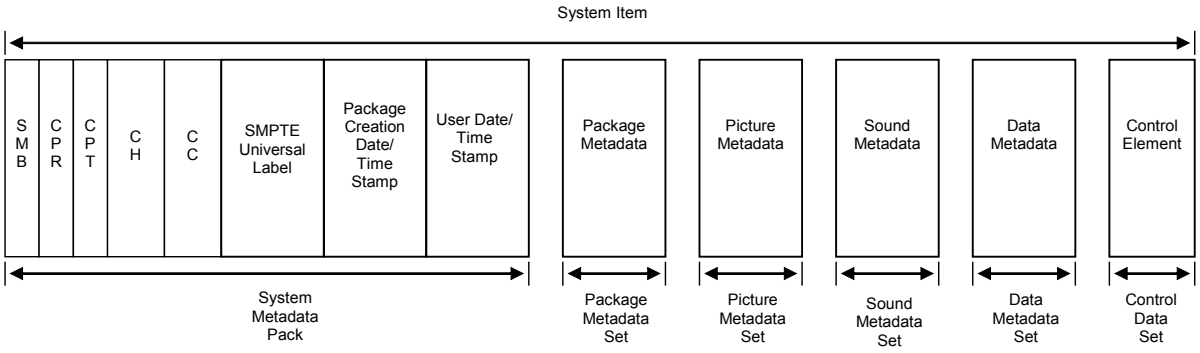

Figure 1 – Elements of the SDTI-CP compatible System Item

This System Item shall be divided into the following elements according to Figure 1:

- A System Metadata pack comprising a first 7 bytes and up to 50 further bytes (as defined by the first byte of the pack).
- A package metadata set that provides metadata for all Essence Elements in the Content Package.
- A picture metadata set that provides metadata for any Essence Element in the Picture Item. The set includes identification to link the metadata item to a specific Essence Element in the Picture Item.
- A sound metadata set that provides metadata for any Essence Element in the Sound Item. The set includes identification to link the metadata items to a specific Essence Element in the Sound Item.
- A data metadata set that provides metadata for any Essence Element in the data item. The set includes identification to link the metadata items to a specific Essence Element in the data item.
- A control data set comprising of only one data item that defines a downstream control element for the Content Package.

Where the System Metadata pack is present in the MXF Generic Container it shall be the first KLV packet in every Content Package. As defined in SMPTE ST 326, the Content Package metadata set immediately follows the System Metadata pack. If there are no items of Content Package metadata, then the length field of the packet shall be zero. Also per SMPTE ST 326, the possible presence of each of the picture, sound and data metadata sets and the control data set is determined by bits b3 to b0 of the first byte of the System Metadata pack.

Where present, the order of the picture, sound and data metadata sets shall be the same as the order of the picture, sound and data items in the MXF Generic Container.

Where present, the control data set shall be the last KLV packet of this System Item.

There shall be a maximum of one system, package, picture, sound, data or control set/pack in any CP-compatible System Item.

5.2 System Metadata Pack

Figure 2 illustrates the first 7 data words of the System Metadata pack.

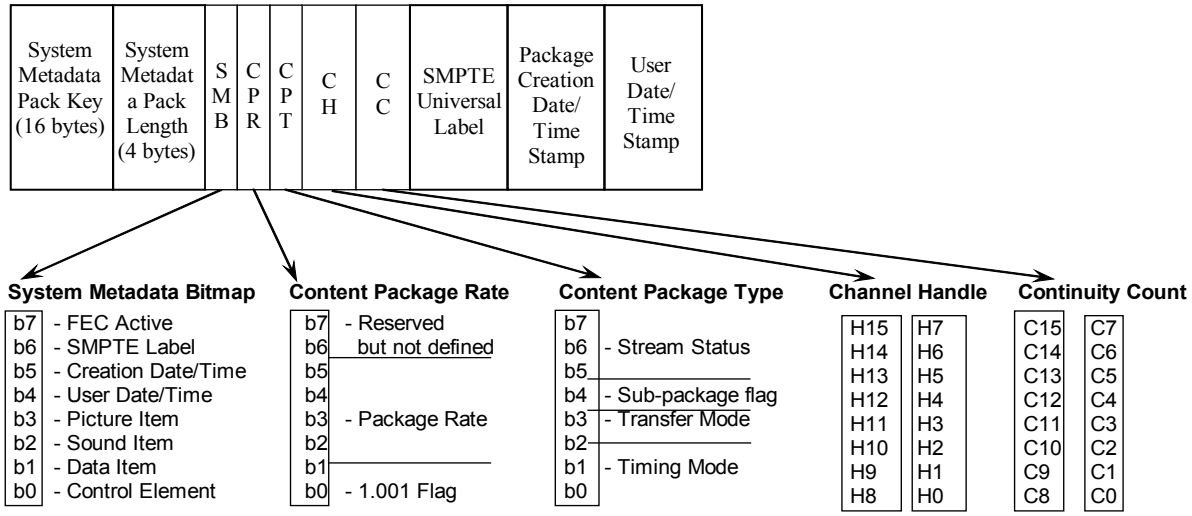

Figure 2 – System Metadata pack structure

5.2.1 System Metadata pack key

The System Metadata pack key shall be as defined in Table 1.

Table 1 – Specification of the key for the system metadata pack

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	05h	Defined length Packs
7	Structure Designator	01h	Sets & Packs Registry
8	Version Number	01h	Version 1 of the registry
9	Item Designator	0Dh	Organisationally Registered
10	Organisation	01h	AAF Association
11	Structure	03h	MXF generic container Keys
12	Structure Version	01h	MXF-GC Version 1
13	Item Type Identifier	04h	CP-compatible system item
14	System Scheme Identifier	01h	SDTI-CP, Version 1
15	Metadata Element Identifier	01h	System Metadata Pack
16	Reserved	00h	

5.2.2 System Metadata pack length

The System Metadata pack length field shall use 4-byte BER long-form encoding (83.xx.yy.zz).

5.2.3 System Metadata pack value

The component parts in the System Metadata pack value field shall be as defined in SMPTE ST 326. They are summarized as follows, including any constraints on their use:

The System Metadata pack comprises up to 57 bytes defined by the following pack items as illustrated in Figure 2:

- A System Metadata bitmap word (1 byte);
- A Content Package rate word (1 byte);
- A Content Package type word, including stream status flags (1 byte);
- A channel handle word (2 bytes);
- A continuity count word (2 bytes);
- A SMPTE Universal Label (16 bytes);
- A creation date/time stamp (17 bytes); and
- A user date/time stamp (17 bytes).

These component parts are briefly described in the following paragraphs. The full description of each component part can be found in SMPTE ST 326.

The first component fields of the System Metadata pack shall be completed as follows:

5.2.3.1 Core fields

System Metadata bitmap:

b7 = 0 (FEC not used),
b6 = 1 (SMPTE Universal Label is present, see below)
b5 = 1 (creation date/time stamp is present) [Note 1]
b4 = 0 or 1 (user date/time stamp) [Note 1]
b3 = 0 or 1 (Picture Item present) [Note 1]
b2 = 0 or 1 (Sound Item present) [Note 1]
b1 = 0 or 1 (data item present) [Note 1]
b0 = 0 or 1 (control element present) [Note 1]

Content Package rate: shall be completed to reflect the correct value as defined in SMPTE ST 326

Content Package type

Stream status = 0, or 1~6 as required [the first value is the default]
Sub-package flag = 0
Transfer mode = 0 (default value) [Note 2]
Timing mode = 0 (default value) [Note 2]

Channel handle = 0 (default value)

Continuity count = modulo 65536 count as per SMPTE ST 326 [Note 3]

Note 1: The value depends on the application specification.

Note 2: These bits do not have the definition specified in SMPTE ST 326 and have no defined meaning in the MXF Generic Container. They are set to the default value of zero.

Note 3: The continuity count is not required in an MXF file because the Header Metadata correctly describes the timeline of the Essence Container. However, to maintain compatibility with the SDTI-CP System Item definition, the continuity count is provided to comply with SMPTE ST 326.

5.2.3.2 SMPTE Universal Label

The SMPTE Universal Label used in the System Metadata pack identifies the MXF Generic Container and its payload. The value of this UL should be as defined in Table 3 of SMPTE ST 379-1 and SMPTE ST 379-2.

Note: The SMPTE UL value in the MXF file then becomes a link to all Content Packages in the MXF Generic Container having the same UL value.

5.2.3.3 Creation date/time stamp

A creation date/time stamp value should be entered in the System Metadata pack according to SMPTE ST 326. The format of this item is defined by the first byte that has the metadata type value of '81h' (timecode) or '82h' (date-timecode) as defined in SMPTE ST 331. In compliance with SMPTE ST 331, of the 17 bytes available, only the first 9 bytes shall be used and the last 8 bytes shall be zero-filled.

5.2.3.4 User date/time stamp

A user defined date/time stamp value may be entered in the System Metadata pack according to SMPTE ST 326. The format of this item shall be identical to that defined for the creation date/time stamp.

5.2.4 Package, picture, sound and data metadata sets and the control data set

The first metadata set shall be the package metadata set that contains metadata for the package as a whole; such as, for example, a time code. This metadata does not require a 'link' item as each metadata item of the set is related to the package as a whole rather than any individual part.

The subsequent metadata sets shall be the picture, sound and data metadata sets. These shall only be present if the associated picture, sound or data item is present in the Content Package. Each metadata set may have zero or more 'link' metadata items that link each metadata item in the set to the associated Essence Element in the picture, sound or data item.

Note: These metadata sets have no formal class structure. Instead, each metadata item is coded in a metadata set with a local tag value for identification and a 'link' metadata item that provides a unique link for subsequent metadata items to the associated Essence Element.

Where present, the last data set shall be the control data set. Although coded as a data set, this set shall contain only one control data item for compatibility with SMPTE ST 326.

Each set shall be coded as a KLV local set. Figure 3 illustrates the basic structure of the package, picture, sound and data metadata sets and the control data set.

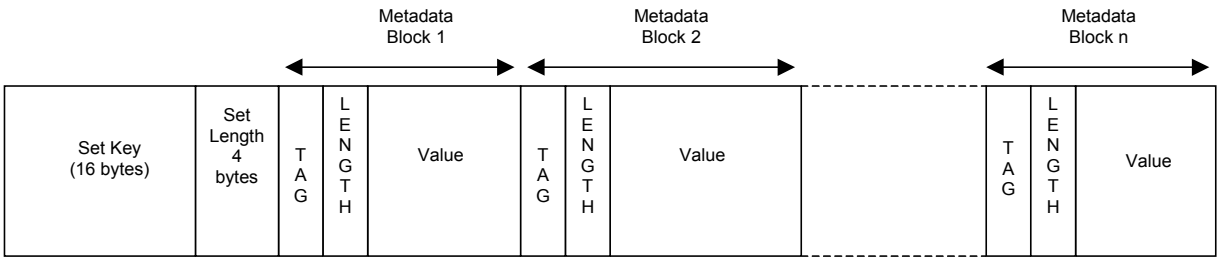

Figure 3 – Structure for the package, picture, sound and data metadata sets

5.2.5 Metadata and control data set keys

The metadata and control data set keys shall be as defined in Table 2.

Table 2 – Specification of the key for the system item sets

Byte No.	Description	Value (hex)	Meaning
1	Object Identifier	06h	
2	UL size	0Eh	
3	Designator	2Bh	ISO, ORG
4	Designator	34h	SMPTE
5	Registry Category Designator	02h	Sets & packs
6	Registry Designator:	Package, Picture, Sound & Data Metadata Sets = 43h Control data set = 63h	Metadata sets (43h) = 1-byte tag, 2-byte length Control set (63h) = 1-byte tag, 4-byte length
7	Structure Designator	01h	Sets & Packs Registry
8	Version Number	01h	Version 1 of the registry
9	Item Designator	0Dh	Organizationally Registered
10	Organisation	01h	AAF Association
11	Structure	03h	MXF generic container keys
12	Structure Version	01h	MXF-GC version 1
13	Item Type Identifier	04h	CP-compatible system item
14	System Scheme Identifier	01h	SDTI-CP, version 1
15	Metadata Element Identifier	02h~06h	Package metadata set = 02h Picture metadata set = 03h Sound metadata set = 04h Data metadata set = 05h Control data set = 06h
16	Metadata Block Count	xxh	Number of metadata blocks in the element (zero for the control data set))

The package, picture, sound and data metadata sets embed the ‘metadata block count’ byte into the set key to maintain compatibility with SMPTE ST 326. The control data set does not require a ‘metadata block count’ byte and thus byte 16 shall be set to zero.

5.2.6 Metadata and control data set length

Each metadata and control data set length field shall use 4-byte BER long-form encoding (83.xx.yy.zz).

Note: The control set has data blocks with 4-byte big-endian length fields that can exceed the capability of 4-byte BER long-form encoding. However, in practice, the range is more than sufficient for all typical applications.

5.2.7 Metadata and control data set values

A metadata set shall comprise one or more metadata items. Each item of a set shall consist of a 1-byte tag followed by a 2-byte or 4-byte item length and completed by the metadata value. The item length shall be 2-bytes for data blocks in the picture, sound and data metadata sets and 4-bytes in the control data set.

The common structures for data blocks in System Item metadata sets are illustrated in Figure 4.

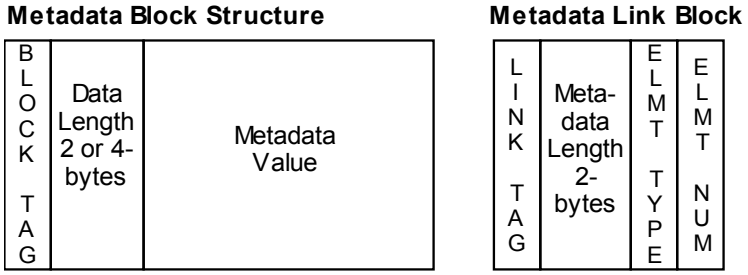

Figure 4 – Structure for metadata items including the metadata link item

5.2.7.1 Metadata link item

Picture, sound and data metadata items shall be preceded by a metadata 'link' item which provides a link between the metadata in the System Item set and the associated Essence Element in a picture, sound and data item. A metadata 'link' item should occur at least as many times as there are Essence Elements to link. Each time a metadata 'link' item is found, the immediately following metadata items shall all refer to the linked Essence Element until the next metadata 'link' item is found or the set is complete.

The value of the link metadata item shall be as defined in SMPTE ST 331. In SMPTE ST 326, the coding of this item does not include a length field. For consistency, this field has been added in this specification. The value of this length field shall be '0002h'.

5.2.7.2 Tag values

The tag value for any metadata block shall be as defined in SMPTE ST 331.

5.2.7.3 Length values

The length value of package, picture, sound or data metadata items shall be 2-bytes encoded as big-endian.

The length value of a control data item shall be 4-bytes encoded as big-endian.

5.3 System Item Relationship to the MXF Header Metadata

5.3.1 Mapping track numbers to metadata elements

The Header Metadata may include descriptors to describe the metadata in one or more of the CP-compatible System Item metadata elements used. The track number value shall be defined as described in the MXF Generic Container specification.

5.3.2 Use of descriptors for CP-compatible metadata elements

The System Metadata pack and the control data set are the only CP-compatible System Item metadata elements that may require a Header Metadata descriptor set. The package, picture, sound and data metadata sets may use the pre-defined scheme as defined in this standard. Each metadata element shall have a unique track number as defined in the previous section.

Where a metadata element requires more than one descriptor set, then the multiple descriptor set defined in SMPTE ST 377-1 shall be used to reference each individual descriptor set.

5.3.3 Descriptors for CP-compatible metadata elements

The appropriate Header Metadata package should have a descriptor for the ‘creation date/time’ item. The appropriate Header Metadata package should have a descriptor for the ‘user date/time’ item where that item is present.

The definition of the descriptor for both creation and user date/time items shall be as defined in Table 3. This descriptor shall be a sub-class of the File Descriptor as defined in SMPTE ST 377-1. The last four rows of this table are metadata items added specifically for this Date/Time Descriptor.

Table 3 – Date/Time Descriptor

Item Name	Type	Len	Local Tag	UL Designator	Req ?	Meaning	Default
Date/Time Descriptor	Set Key	16		06.0E.2B.34 02.xx.01.vv 0D.01.01.01 01.01.46.00	Req	Defines the Date/Time Descriptor set	
Length	BER Length	Var			Req	Set length	
All items in the File Descriptor as defined in SMPTE ST 377-1	See SMPTE ST 377-1 Table F.2						
Date/Time Rate	Rational	8	35.01	06.0E.2B.34 01.01.01.05 04.04.01.02 01.00.00.00	Opt	Defines the Date/Time rate where this differs from the essence container rate	Sample Rate
Date/Time Drop Frame	Boolean	1	35.02	06.0E.2B.34 01.01.01.05 04.04.01.02 02.00.00.00	Opt	TRUE if drop-frame is active	FALSE
Date/Time Embedded	Boolean	1	35.03	06.0E.2B.34 01.01.01.05 04.04.01.02 03.00.00.00	Opt	Is it embedded in other data?	TRUE
Date/Time Kind	UL	16	35.04	06.0E.2B.34 01.01.01.05 04.04.01.02 04.00.00.00	Req	Date/Time format kind. Values are listed in SMPTE RP 224	

6 Mapping the SDTI-CP System Item to the MXF Generic Container

This section describes how the contents of the SDTI-CP System Item shall be mapped to the MXF Generic Container. The 8-bit data words extracted from an SDTI-CP interface shall be those data words that reside between the SDTI-CP separators and end code words. The separator and end code words are used in SDTI-CP to provide unique start and stop code words for each CP item as defined in SMPTE ST 326 and shall not be considered as part of this mapping process.

6.1 System Item Mapping

In SDTI-CP, the System Item is a single composite data format that must be parsed and divided into the metadata elements defined in Section 5.1. These elements shall be KLV coded with keys, lengths and values as follows.

6.1.1 System Item element keys

In the MXF Generic Container, the 16-byte System Item keys shall be created for each element, as a pack or set, and integrate a number of bytes defined in the System Item of SDTI-CP.

For each System Item element, the key values shall be as defined in Table 1 and Table 2 with the following requirements:

- The registry designator value (byte 6) of the key shall correctly represent the format of the metadata pack or control data set.
- Byte 13 of the key shall be set to the System Item type value (04h) for all metadata packs or control data sets.
- Byte 14 of each key is the System Item scheme identifier. For all keys, the value shall be '01h' identifying the scheme to be SDTI-CP compatible.
- The System Item element number is identified in byte 15 and the values shall lie in the range '01h' to '06h'. The sequence of the elements at the SDTI-CP interface port shall follow the rules defined by SMPTE ST 326. The values in byte 15 of the key can be used to bring together the System Item elements of the MXF Generic Container back into the composite form used in SDTI-CP.
- For any System Metadata packs and any control data sets, byte 16 shall be set to '00h'.

6.1.2 System Item element lengths

The System Item elements shall each be coded as individual KLV packets which shall use 4-byte BER long-form encoding (83.xx.yy.zz). Each element length shall be calculated from the length of the related element value.

Note: The length value in the SDTI-CP interface gives the length of the composite CP System Item and is of no practical use for this calculation.

For the package, picture, sound and data metadata sets only, the metadata count of SDTI-CP shall be encapsulated into the key values. The metadata count shall not be included in the calculation of the length value. Thus a package, picture, sound or data metadata set may comprise simply a key followed by a length having a value of zero (and hence no value field).

Note: In converting back to SDTI-CP, the word count in CP is a 4-byte little-endian value that defines the entire length of the SDTI-CP System Item value. This composite length value is the sum of the length values of each SDTI-CP compatible System Item element. However, this sum will be modified by the following conditions:

- The presence of any metadata link items (see Figure 4) since a 2-byte length value has been added to the MXF Generic Container where it is not present in SDTI-CP.
- The reinsertion of the metadata count values in the package, picture, sound and data metadata sets where present.

6.1.3 System Item element values

The value from the SDTI-CP System Metadata pack of the MXF Generic Container shall be directly mapped from the first element of the SDTI-CP System Item data area. Subsequent metadata sets shall be mapped as follows:

- The metadata count value shall be removed from the first byte of each metadata set.
- For each metadata block in each metadata set, the length value shall be byte swapped from the 2-byte little-endian length values defined in SDTI-CP to the 2-byte big-endian values defined in the SDTI-CP compatible sets. The one exception to this rule applies to the metadata link block where the 2-byte length field shall be added to maintain a consistent KLV coding structure.
- Any control data set in the SDTI-CP compatible System Item uses 4-byte big-endian length values. Each 4-byte little-endian length value of an SDTI-CP source shall be byte swapped to 4-byte big-endian values.

7 Mapping SDTI-CP Picture, Sound and Auxiliary Items to the MXF Generic Container

This section defines how the contents of the SDTI-CP system, picture, sound and Auxiliary Items are mapped to the MXF Generic Container. The 8-bit data words extracted from an SDTI-CP interface shall be those data words that reside between the SDTI-CP separator and end code words. The separator and end code words are used in SDTI-CP to provide unique start and stop code words for each CP item as defined in SMPTE ST 326 and shall not be used in this mapping process.

7.1 Picture, Sound and Auxiliary Item Mapping Details

The general data structure of SDTI-CP picture, sound and Auxiliary Items is illustrated in Figure 5.

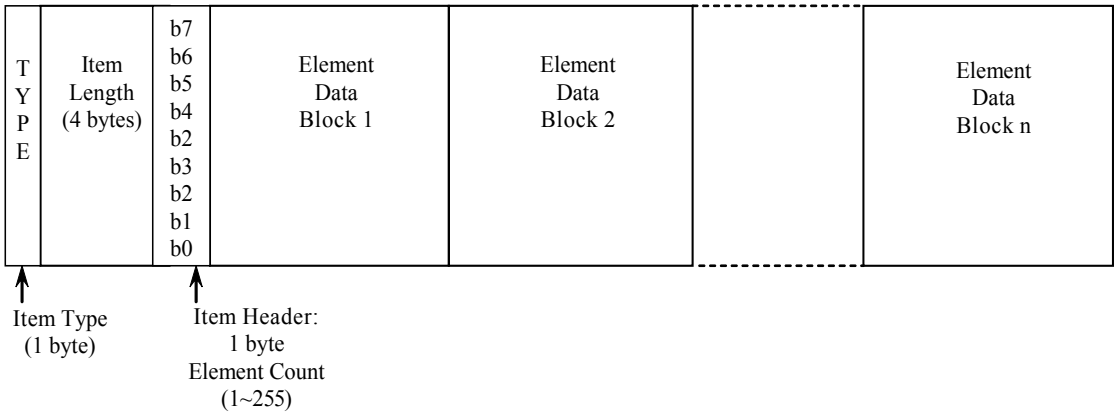

Figure 5 – Structure of SDTI-CP picture, sound and Auxiliary Items

7.1.1 Item Keys

The picture, sound and Auxiliary Items of SDTI-CP have a common data structure where the Item starts with an item type value that shall be used in byte 13 of a MXF Generic Container element key.

The item header is a 1-byte count of the number of elements in the item as illustrated if Figure 5. This value shall be used for byte 14 of a MXF Generic Container element key. Per SMPTE ST 326, the value lies in the range 1 to 127 (0 is not valid). The item header value is provided so that the number of element data blocks in the item is known in advance of parsing the item.

7.1.2 Item Lengths

The item length value shall define the length of the element value field. The value ‘0’ may be used to indicate that there is no defined length value and that the end of the item shall be defined by the presence of the SDTI end code word.

7.1.3 Item Values

The item value in SDTI-CP is a concatenation of the SDTI-CP elements as illustrated in Figure 5.

7.2 Essence Element Mapping

Each SDTI-CP compatible Essence Element in the MXF Generic Container shall be coded as a separate KLV packet even though the logical structure places all picture elements in a Picture Item, all sound elements in a Sound Item and all data elements in a data item.

Each SDTI-CP compatible Essence Element value shall be as defined in SMPTE ST 331. This section will define only the differences in the element key values and the length values.

7.2.1 Element Keys

The SDTI-CP compatible Essence Element keys integrate a number of component parts defined in SMPTE ST 326 as follows:

- The SDTI-CP item value shall be defined in byte 13 of each SDTI-CP compatible Essence Element key. Therefore, elements in the same Item can be grouped for mapping into SDTI-CP.

- The element count value that is present after the SDTI-CP word count field shall be defined in byte 14 of each MXF Generic Container element key. Therefore, each Essence Element can know whether it is a single element in an item or one of two or more. The value in the GC shall be constrained to a maximum of 127.

- The element type value that is used in SDTI-CP to identify the kind of element present in the value field shall be defined in byte 15 of the MXF Generic Container element key. The value in the GC shall be constrained to a maximum of 127.

- The element number value that is used in SDTI-CP to give a locally unique number to each element shall be defined in byte 16 of the MXF Generic Container element key. The value in the GC shall be constrained to a maximum of 127.

7.2.2 Element lengths

MXF Generic Container length fields are coded as BER long-form. SDTI-CP compatible length fields shall be coded with a BER long-form length of 4 bytes (83h.xx.yy.zz).

SDTI-CP length fields are 4 byte UInt32 values coded as little-endian. The little-endian output value is therefore ‘zz.yy.xx.00’.

7.2.3 Element values

Each Essence Element value is byte-for-byte identical in the MXF Generic Container and SDTI-CP.

7.2.4 Element mapping details

The element data blocks of SDTI-CP have a common data structure, shown in Figure 6, where the element starts with an element type word which is the same as the element type value in byte 15 of an MXF Generic Container element key.

The element length value shall be decremented by 1 (to allow for the removal of the element number byte) and converted to 4-byte BER long form.

The element number byte is identical to that in byte 16 of an MXF Generic Container element key.

The element value is identical in the SDTI-CP compatible MXF Generic Container and SDTI-CP.

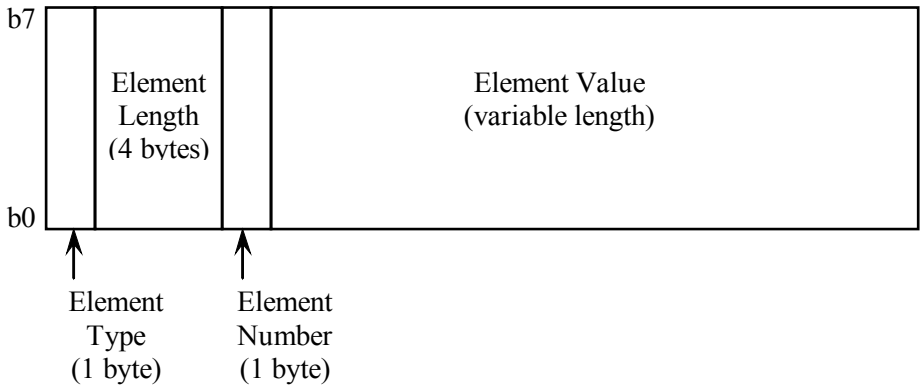

Figure 6 – Structure of SDTI-CP element data blocks

8 Length Conversions

The 4-byte length fields of SDTI variable blocks are little-endian format. For each CP item, the length field (after any adjustments) shall be converted from 4-byte little-endian form to 4-byte BER long-form coding. This format conversion can be easily achieved as follows:

SDTI length		BER length
ab.cd.ef.gh	becomes..	83.ef.cd.ab

The first byte is '83h' which indicates BER long-form encoding with 3 bytes to follow. The 3 bytes are the most significant of the SDTI length, thus the rightmost byte of the SDTI length ('gh') shall be discarded in this conversion. The remaining 3 bytes still allow a length of up to (2^24 -1) bytes.

Annex A Bibliography (Informative)

Note: All references in this document to other SMPTE documents use the current numbering style (e.g. SMPTE ST 298:2009) although, during a transitional phase, the document as published (printed or PDF) may bear an older designation (such as SMPTE 298-2009). Documents with the same root number (e.g. 298) and publication year (e.g. 2009) are functionally identical.

SMPTE ST 298:2009, Universal Labels for Unique Identification of Digital Data

SMPTE ST 305:2005, Television — Serial Data Transport Interface

SMPTE RP 204:2000, SDTI-CP MPEG Decoder Templates

SMPTE RP 210, Metadata Element Dictionary

SMPTE RP 224, SMPTE Labels Register

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

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