PROPOSED SMPTE STANDARD

SMPTE 383M

for Television —

Material Exchange Format (MXF) — Mapping DV-DIF Data to the MXF Generic container (Standard)

Table of contents

1 Scope

This standard specifies the mapping of DV-DIF (digital interface format) data to an MXF generic container, which is suitable for the application of all variants of the DV specifications as defined by IEC61834-2, SMPTE 314M and SMPTE 370M. This standard defines a SMPTE Universal label that can be used to uniquely identify specific DV variants.

The MXF specification is written in several parts. This is one of a set of documents that define the contents of the MXF file body.

In order to achieve interoperability within any given operational pattern, restrictions may be placed on the way in which this generic container type can be implemented. The reader is advised to carefully study the appropriate operational pattern document before 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 edition of the standards indicated below.

SMPTE 314M-1999, Television — Data Structure for DV-Based Audio, Data and Compressed Video — 25 and 50 Mb/s.

SMPTE 336M-2001, Television — Data Encoding Protocol Using Key-Length-Value.

SMPTE 370M-2002, Television — Data Structure for DV-Based Audio, Data and Compressed Video at 100 Mb/s 1080/60i, 1080/50i, 720/60p

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

SMPTE 379M, Television — Material Exchange Format (MXF) — Generic Container Format

IEC61834-2 (1998-08), Recording – Helical-Scan Digital Video Cassette Recording System using 6.35mm Magnetic Tape for Consumer Use (525-60,625-50,1125-60 and 1250-50Systems), Part 2: SD format for 525-60 and 625-50 Systems

3 Glossary of acronyms, terms and data types

The full glossary of acronyms terms and data types used in the MXF specification is given in the MXF format specification. A supplementary glossary of acronyms and terms is defined in SMPTE 379M. They are not repeated here to avoid any divergence of meaning.

DV-DIF block: a 3-byte ID followed by 77 bytes of data.

DIF sequence: a specific sequence of header, subcode, VAUX, audio and video DIF blocks as defined in SMPTE 314M.

DIF channel: a number of DIF sequences as defined in SMPTE 314M.

DV-DIF data: a generic term for a number of DIF blocks.

DV-DIF frame: a generic term for all the DIF sequences which make up a picture frame.

4 Introduction

This standard will use the generic terms “DV-DIF” when referring to the basic DIF structure of both the IEC-DV and DV-based variants. There are three compression specifications which create DV-DIF data:

“IEC DV” operating at 525/60i and 625/50i as specified in IEC61834-2;
“DV-based” operating at 525/60i and 625/50i as specified in SMPTE 314M;
“DV-based” operating at 1080/60i, 1080/50i and 720/60p as specified in SMPTE 370M.

For a detailed list of differences between the various DV formats such as color sampling and setting of the flags in the DV stream, the reader is referred to annex A of SMPTE 314M. The table is not reproduced here to prevent duplication of information. Section 8 of this standard contains the numeric values required to differentiate between implementations. The terms “IEC DV” and “DV-based” will be used to highlight differences such as Y/C sampling ratio in the 625/50 (PAL) system and allowed bit rates where appropriate. The reader is referred to the normative references for the details of the specifications.

This standard defines standard definition systems operating at 525/60 and 625/50 using the IEC DV and DV-based format operating at 25 Mb/s. It also defines standard definition systems operating at 525/60 and 625/50 using the DV-based format operating at 50 Mb/s and higher definition systems operating at 100 Mb/s.

NOTE – Any 60-Hz operation system indicated in this document is a shorthand for the actual operation at a field rate of 59.94Hz.

This standard specifies methods of encapsulating DV-DIF data in a key-length-value construct as defined by SMPTE 336M.

This specification defines a unique SMPTE Universal label to identify which variant of DV-DIF data and their containment type are present in the MXF generic container.

The DV-DIF data consists of header, subcode, VAUX, audio and video DIF blocks. The data which consists of the DV-DIF block structure containing all DV-DIF data is mapped as the compound element in the MXF generic container.

4.1 DV-DIF data structure

The DV-DIF data is structured by the DIF block of 80 bytes. The data contains:

DV auxiliary data contains header, subcode and video auxiliary DIF blocks;
DV video data contains DV compressed video DIF blocks;
DV audio data contains audio DIF blocks.

The detailed specifications of each type and variant of the DV-DIF data is given in SMPTE 314M, SMPTE 370M and IEC 61834-2.

4.2 DV-DIF data distribution (informative)

SMPTE 314M, SMPTE 370M and IEC 61834-2 define the distribution of DV-DIF data in a single frame period.

For 60-Hz systems (525/59.94i, 1080/59.94i and 720/59.94p), there are 10 identical DIF sequences in each DIF channel.

For 50-Hz systems (625/50i and 1080/50i), there are 12 identical DIF sequences in each DIF channel.

Each DIF sequence is assigned to a DIF channel.

For 25-Mbps IEC DV and DV-based compression there is one DIF channel.

For 50-Mbps DV-based compression there are two DIF channels.

For 100-Mbps DV-based compression there are four DIF channels.

For DV bit-rates requiring more than one DIF channel, each DIF channel is presented in the order defined in SMPTE 314M and SMPTE 370M.

The number of DV-DIF blocks per frame for each frame-rate and bit-rate is as defined in table 1.

Table 1 – DV-DIF block counts over a frame duration

DV-DIF BlockType	525/60		625/50		HD 100 Mbit/s	
	25 Mbit/s	50 Mbit/s	25 Mbit/s	50 Mbit/s	60i or 60p	50i
Header (H)	10	20	12	24	40	48
Subcode (SC)	20	40	24	48	80	96
VAUX	30	60	36	72	120	144
Audio	90	180	108	216	360	432
Video	1350	2700	1620	3240	5400	6480
Total	1500	3000	1800	3600	6000	7200

The DV-DIF data shall be presented in the order defined in SMPTE 314M, SMPTE 370M and IEC 61834-2 for the purpose of mapping to the MXF generic container.

5 Mapping the DV-DIF data to the MXF generic container

DV-DIF data shall be KLV coded as defined in SMPTE 336M. The auxiliary, audio and video DIF data for a single frame are shown pictorially in figure 1. When DV-DIF data is mapped into the generic container, a number of options are available for mapping into the various GC item types.

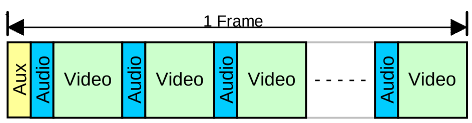

Figure 1 – Simple representation of a DIF frame

5.1 Frame and clip wrapping

There are two methods of mapping DV-DIF data into the MXF generic container as defined by SMPTE 379M. These are "frame wrapping" (figure 2) and "clip wrapping" (figure 3).

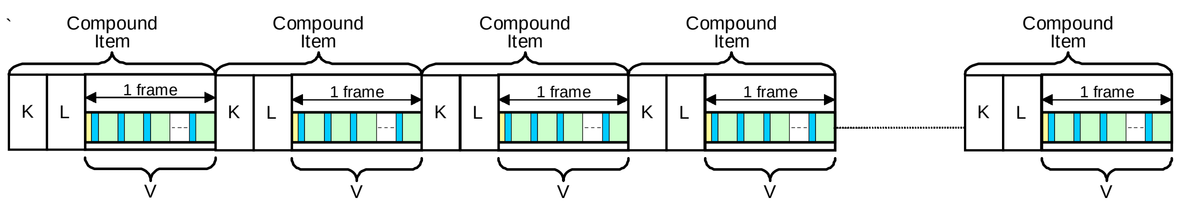

Figure 2 – Simple representation of frame wrapping

The frame wrapping method is intended to enable frame by frame access by MXF applications which process at the KLV level. This can be particularly useful for applications which support multiple generic container mapping types. Sufficient Information is provided to allow individual frames to be identified at the KLV level without an MXF decoder having to parse or decode the essence data. Each frame of DV-DIF data shall be KLV wrapped using a compound element key.

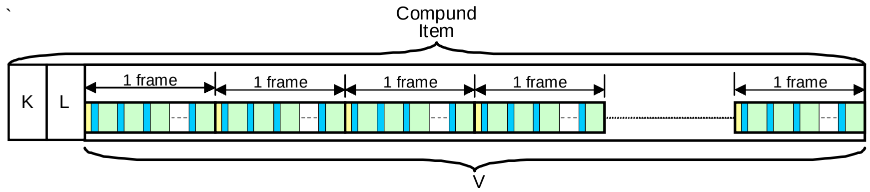

Figure 3 – Simple representation of clip wrapping

The clip wrapping method is intended for applications which carry the DV-DIF data as a single large entity. This can be very useful applications such as store and forward servers which process whole files and also in applications where it is desired to use the rich metadata structures of MXF as an annotation to DV data. The clip of DV-DIF data shall be KLV wrapped using a compound element key. When DV is clip wrapped, there shall be only one clip per generic container body. Multiple clips can be concatenated and edited using the operational pattern mechanism detailed in the MXF format document.

6 KLV coding of DV-DIF data

6.1 Compound item mapping

The MXF-GC compound element for DV-DIF data contains an interleave of audio, video and auxiliary DV-DIF as either a single DV-DIF frame (frame wrapping) or as a sequence of one or more DV-DIF frames (clip wrapping).

6.1.1 DV compound element key

The values of the last four bytes of the essence element key are given below:

Table 2 – Key value for the DV-DIF compound element

Byte No.	Description	Value (hex)	Meaning
1-12	Specified by the MXF Generic container Specification SMPTE379M		
13	Item Type Identifier	18h	Compound Item
14	Essence Element Count	kkh	Count of Compound Elements in this Item
15	Essence Element Type	01h 02h	DV-DIF Frame Wrapped Element DV-DIF Clip Wrapped Element as listed in SMPTE RP224
16	Essence Element Number	nnh	A number (used as an Index) of this Compound Item in this Generic container as defined in SMPTE379M

The key shall be unique within the partition in which it is used.

6.1.2 Essence element count – Byte 14

This is a count of the number of elements in each compound item in the generic container. The BodySID mechanism detailed in the MXF format specification shall be used to identify the correct essence container.

6.1.3 Essence element type – Byte 15

For frame wrapped DV-DIF, this shall be 01h. For clip wrapped DV-DIF, this shall be 02h.

6.1.4 Essence element number – Byte 16

This is a number used as an index to identify this instance of the element type within the item. It shall have a value between 1 and kkh. It shall not change within any instance of a generic container.

6.1.5 DV compound element length

The length field of the KLV coded element shall be 4 bytes BER long-form encoded (i.e., 83h.xx.yy.zz) for frame wrapping. The length field of the KLV coded element shall be 8 bytes BER long-form encoded (i.e., 87h.aa.bb.cc.dd.ee.ff.gg) for clip wrapping.

6.1.6 DV compound element value

This shall be the appropriately wrapped DV-DIF data bytes.

7 System item mapping

No system item is required.

8 SMPTE label for essence container identification

The Essence Container UL is defined in SMPTE 379M.

The DV specific values for the essence container UL are given in table 3.

Table 3 – Specification of the essence container label

Byte No.	Description	Value (hex)	Meaning
1-12	Specified by the MXF Generic container Specification SMPTE379M		
13	Essence Container Kind	02h	MXF Generic container
14	Mapping Kind	02h	DV-DIF as listed in SMPTE RP224
15	Locally defined	yyh	See **Table 4**
16	Locally defined	yyh	See Table 5

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

Byte 14 identifies the container as the DV-DIF mapping into the generic container.

Byte 15 carries information about the DV implementation and the value shall be set according to table 4.

For IEC DV mappings, the default value of byte 15 of the key shall be '3Fh' which shall be used to indicate no defined method of IEC-DV coding and hence no implied quality level.

For DV-based mappings, the default value of byte 15 shall be '7Fh' which shall be used to indicate no defined method of DV-based coding and hence no implied quality level.

Table 4 – Specification of byte 15 of DV container label

Byte 15 Value	DV type	Content type
00	Not used	Not used
01	IEC DV, 25Mbps 525-I	IEC DV compressed at 25Mbps, 525/59.94-I source
02	IEC DV, 25Mbps 625-I	IEC DV compressed at 25Mbps, 625/50-I source
03	IEC DV, 25Mbps 525-lines, Registered Quality: SMPTE 322M Compatible	IEC DV compressed at 25Mbps, 525/59.94-I source
04	IEC DV, 25Mbps 625-lines, Registered Quality: SMPTE 322M Compatible	IEC DV compressed at 25Mbps, 625/50-I source
3F	IEC DV, undefined	Undefined IEC DV compressed
40	DV-based, 25Mbps 525-I	DV-based compressed at 25Mbps, 525/59.94-I source
41	DV-based, 25Mpbs 625-I	DV-based compressed at 25Mbps, 625/50-I source
50	DV-based, 50Mbps 525-I	DV-based compressed at 50Mbps, 525/59.94-I source
51	DV-based, 50Mpbs 625-I	DV-based compressed at 50Mbps, 625/50-I source
60	DV-based, 100Mpbs 1080/59.94-I	DV-based compressed at 100Mbps, 1080/59.94-I source
61	DV-based, 100Mpbs 1080/50-I	DV-based compressed at 100Mbps, 1080/50-I source
62	DV-based, 100Mpbs 720/59.94-P	DV-based compressed at 100Mbps, 720/59.94-P
63	DV-based, 100Mpbs 720/50-P (placeholder)	DV-based compressed at 100Mbps, 720/50-P (placeholder)
7F	DV-based, undefined	Undefined DV-based compressed

Byte 16 is intended to carry basic information about the containment of the DV. It is set according to table 5.

Table 5 – **Specification of DV container label, byte 16**

Byte 16	Meaning
00h	not used
01h	Frame Wrapping
02h	Clip Wrapping
03h-7Fh	Reserved

9 Use of DV-DIF in the generic container (informative)

In addition to the overall frame wrapping strategy, it is often necessary to allow an MXF application to access the generic container sound and auxiliary data by KLV manipulation. There are several mechanisms by which this can be done and some of these are described below.

9.1 Accessing the picture within the compound item

The MXF header metadata contains structural metadata which describes the content. Figure 4 helps to explain the linking mechanism between the header metadata and the DV-DIF generic container essence element used to wrap the DV-DIF structure. Note that in higher operational patterns (e.g., 2x, 3x), there may be several file packages, each of which links to a generic container element.

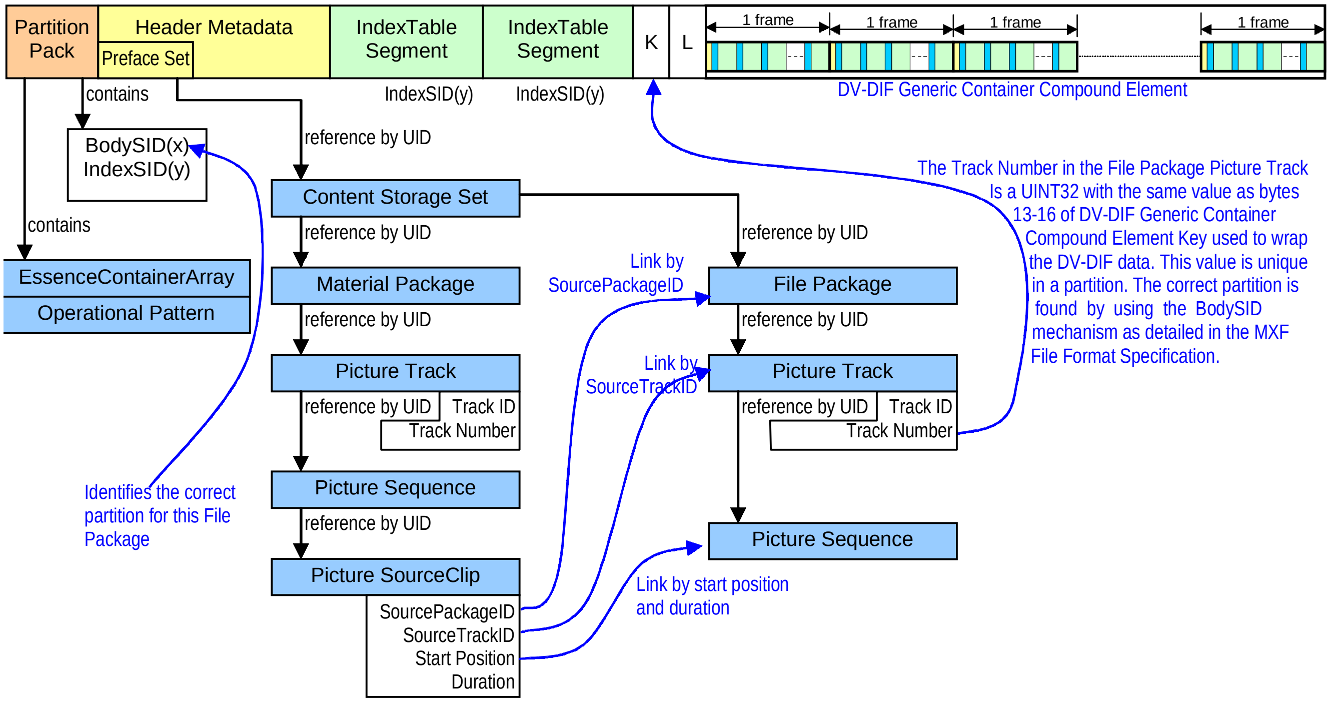

Figure 4 – Linking of the DV-DIF generic container compound element key to the track number item

The sets and structure are fully explained in the MXF format specification. The intent of figure 4 is to indicate how the track number item in the picture track of the file package has the same values as bytes 13-16 of the key used to KLV wrap the DV-DIF data. The same mechanism is used for both frame and clip wrapping types. To identify the correct partition in which to find the content, the BodySID mechanism shall be used as detailed in the MXF format specification.

9.2 Accessing Sound and Data within the Compound Item

If a sound track or a data track exists in the header metadata, then the same linking mechanism shown above shall be used to link to the essence container. This results in several tracks all having the same track number value.

9.3 Extracting sound and data information as separate elements

There is a requirement in some applications to extract sound and data information from the DV-DIF blocks and insert it into the generic container as separate sound and data elements. A generic container mapping document which specifies the format of the resulting sound element or data element is required for this.

An example of this operation may be to extract and process the sound data to create one or more broadcast wave generic container sound elements. The format of elements created by extracting and processing DV-DIF blocks into other formats shall be fully specified in generic container mapping documents. Another example could be the extraction of data buried in the VAUX DIF blocks and creating one or more data elements. The resulting data elements would have to be specified by a separate generic container mapping document. The operations necessary for extracting the data and creating the new generic container elements are fully covered by:

- this specification;
- the underlying DV-DIF specifications;
- the target Generic container elements mapping document;
- any underlying specifications on which the data depends.

Full details are therefore beyond the scope of this standard.

These examples are shown pictorially in figure 5 and figure 6.

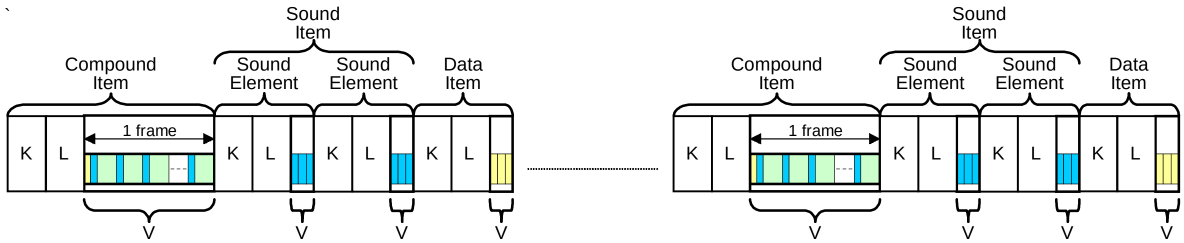

Figure 5 – Frame wrapping example of additional sound and data elements in the generic container

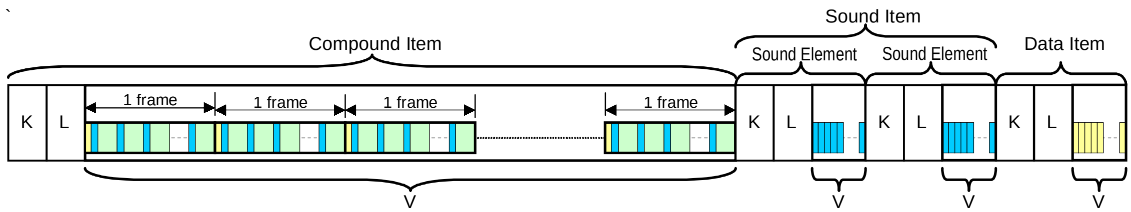

Figure 6 – Clip wrapping example of additional sound and data elements in the generic container

When this technique is used, the DV-DIF auxiliary data and the DV-DIF audio data in the compound item is not overwritten within the DV-DIF compound item.

9.4 Use of KAG

There are no specific KAG requirements for the DV GC mapping. MXF encoders and decoders shall comply with the KAG rules in the MXF format document.

10 File descriptors for DV-DIF (informative)

10.1 CDCI (picture) descriptor

The table below can become very complex if every combination of image characteristics is entered for every possible DV-DIF block format. In many cases the reader is referred to SMPTE 314M and SMPTE 370M in order to obtain the correct numerical value for the property to be completed.

The status of all the items in the descriptor shall be the same as indicated in the MXF format specification.

Item Name	Meaning	Use
CDCI Essence Descriptor	Defines the CDCI Essence Descriptor Set (a collection of parametric metadata)	See MXF Format specification
Length	Set length	See MXF Format specification
Instance UID	Unique ID of this instance	See MXF Format specification
Generation UID	Generation Identifier	See MXF Format specification

Item Name	Meaning	Use
Linked Track ID	Value of the Track ID in this Package to which the descriptor applies	See MXF Format specification
Sample Rate	The frame rate of Essence Container (not the video clock rate)	Refer to SMPTE 314M, 370M,IEC-61834
Container Duration	Duration of Essence Container measured in Edit Units	Calculate - each header should contain the best information available at the time of writing that header. If it cannot be calculated then omit the parameter in Open Headers
Codec Label	UL to identify a codec compatible with the Essence Container. Values are defined in SMPTE RP224	Any registered codec label supporting this specification Refer to RP224 for examples
Essence Container	The UL identifying the Essence Container used to contain this Essence. Values are defined in SMPTE RP224	See **Table 4** The UL defined in this document
Picture Essence Coding	Picture Essence coding, Values are defined in SMPTE RP224	UL identifying the Picture Compression Scheme [**RP210** Specifies the Compression scheme used]
Frame layout	Interlace or progressive layout (See SMPTE 377M for details)	Refer to SMPTE 314M, 370M,IEC-61834
Stored Width	Horizontal Size of active picture	Refer to SMPTE 314M, 370M,IEC-61834
Stored Height	Vertical Field Size of active picture in video lines	Refer to SMPTE 314M, 370M,IEC-61834
SoredF2Offset	Topness adjustment	Refer to SMPTE 314M, 370M,IEC-61834
Sampled Width	Sampled width supplied to codec	Refer to SMPTE 314M, 370M,IEC-61834
Sampled Height	Sampled height supplied to codec	Refer to SMPTE 314M, 370M,IEC-61834
Sampled X-Offset	Offset from sampled to stored width	Refer to SMPTE 314M, 370M,IEC-61834
Sampled Y-Offset	Offset from sampled to stored height	Refer to SMPTE 314M, 370M,IEC-61834
Display Height	Displayed Height placed in Production Aperture	Refer to SMPTE 314M, 370M,IEC-61834
Display Width	Displayed Width placed in Production Aperture	Refer to SMPTE 314M, 370M,IEC-61834
Display X-Offset	Offset from Sampled to Display Width	Refer to SMPTE 314M, 370M,IEC-61834
Display Y-Offset	Offset from Sampled to Display Height	Refer to SMPTE 314M, 370M,IEC-61834
DisplayF2 Offset	Topness adjustment	Refer to SMPTE 314M, 370M,IEC-61834
Aspect Ratio	Specifies the intended aspect ratio of the whole image including any black	Refer to SMPTE 314M, 370M,IEC-61834

Item Name	Meaning	Use
	borders	
Active Format Descriptor	Specifies the intended Framing of the image	Depends on how the material was shot.
Video Line Map	Specifies the first active video line in each field	Refer to SMPTE 314M, 370M,IEC-61834
Alpha Transparency	Is Alpha Inverted? When true, an alpha value is considered trasparent	
Capture Gamma	Defined in SMPTE RP224	Refer to SMPTE 314M, 370M and SMPTE RP224
Image Alignment Offset	Size for Image Alignment	Refer to SMPTE 314M, 370M,IEC-61834
Image Start Offset	Unused bytes before start of stored data	Omit the property or use the value 0
Image End Offset	Unused bytes after end of stored data	Omit the property or use the value 0
FieldDominance	Number of the field which is considered to come first	
Component Depth	Number of active bits per sample (8 or 10)	Refer to SMPTE 314M, 370M,IEC-61834
Horizontal Subsampling	Specifies the H colour subsampling	Refer to SMPTE 314M, 370M,IEC-61834
Vertical Subsampling	Specifies the V colour subsampling	Refer to SMPTE 314M, 370M,IEC-61834
Color Siting	Enumerated value that defines how to compute and place subsampled chrominance values. See SMPTE 377M for details	Refer to SMPTE 314M, 370M,IEC-61834
ReversedByteOrder	False == Same byte order as Rec 601	Refer to SMPTE 314M, 370M,IEC-61834
Padding Bits	Number of bits to round up each pixel to stored size	Omit the property or Refer to SMPTE 314M, 370M,IEC-61834
Alpha Sample Depth	Number of bits per alpha sample	Omit the property or Refer to SMPTE 314M, 370M,IEC-61834
Black Ref Level	16 or 64 (8 or 10-bits)	Refer to SMPTE 314M, 370M,IEC-61834
White Ref level	235 or 943 (8 or 10 bits)	Refer to SMPTE 314M, 370M,IEC-61834
Color Range	225 or 897 (8 or 10 bits)	Refer to SMPTE 314M, 370M,IEC-61834
Container Locators	Ordered array of strong references to Essence Container Locator sets If present, essence may be located external to the file. If there is more than one locator set an MXF Decoder shall use them in the order specified.	Complete according to MXF Format specification

10.2 Sound Essence Descriptor

This descriptor is used when sound tracks reference the sound data within the DV-DIFgGeneric container compound element.

None or one as required

Item Name	Meaning	Use
Sound Essence Descriptor	Defines the Sound Essence Descriptor set (a collection of parametric metadata)	See MXF Format specification
Length	Set length	See MXF Format specification
Instance UID	Unique ID of this instance	See MXF Format specification
Generation UID	Generation Identifier	See MXF Format specification
Linked Track ID	Value of the Track ID in this Package to which the descriptor applies	See MXF Format specification
Sample Rate	The frame rate of Essence Container (not the audio clock rate)	Refer to SMPTE 314M, 370M,IEC-61834
Container Duration	Duration of the Essence Container measured in Edit Units	Calculate - each header should contain the best information available at the time of writing that header. If it cannot be calculated then omit the parameter in Open Headers
Codec Label	UL to identify a codec compatible with the Essence Container. Values are defined in SMPTE RP224	Any registered codec label supporting this specification Refer to RP224 for examples
Essence Container	The UL identifying the Essence Container used to contain this Essence. Values are defined in SMPTE RP224	See Table 4 The UL defined in this document
Sound Essence Compression	Sound Essence coding. Values are defined in SMPTE RP224	UL identifying the Sound Compression Scheme [RP210 Specifies the Compression scheme used]
Audio sampling rate	Sampling rate of the audio essence	Refer to SMPTE 314M, 370M,IEC-61834
Locked/Unlocked	Number of samples per frame is locked or unlocked Locked = non-zero	01h
Audio Ref Level	Audio reference level which gives the number of dBm for 0VU.	Refer to SMPTE 314M, 370M,IEC-61834
Electro-Spatial Formulation	E.g. mono, dual mono, stereo, A,B etc	Refer to SMPTE 314M, 370M,IEC-61834
ChannelCount	Number of Sound Channels Distinguished Value = 0	Refer to SMPTE 314M, 370M,IEC-61834
Quantization bits	Number of quantization bits Distinguished Value = 0	Refer to SMPTE 314M, 370M,IEC-61834
Dial Norm	Gain to be applied to normalize perceived loudness of the clip, (1dB per step)	Refer to SMPTE 314M, 370M,IEC-61834
Container Locators	Ordered array of strong references to Essence Container Locator sets If present, the Essence Container may be located external to the file. If there is more than one locator set an MXF Decoder shall use them in the order specified.	Complete according to MXF Format specification

10.3 Data essence descriptor

Any data essence descriptor should be filled in according to the MXF format specification, SMPTE 314M and SMPTE 370M and any associated data essence descriptions.

11 Index tables for DV-DIF

The DV-DIF compound Item has a static structure and this makes index table generation very simple in most cases. No specific index table segmentation strategy is required by the DV-DIF GC mapping. Index tables are optional, but it is strongly recommended that they be included in MXF files containing DV-DIV compound Items.

11.1 Frame wrapping with no other elements in the generic container

This is a simple case where the index table points to the first byte of the DV-DIF compound element generic container key. There are no other generic container items and any sound or data information is embedded within the DV-DIF container. There is therefore only pieces of information in the index table segment are the start position, duration and (fixed) size of each DV-DIF compound item KLV triplet.

Table 6 – Frame wrapped index table segment set

Item Name	Req ?	Meaning	Use
Index Table Segment	Req	An Index Table Segment set	See MXF Format Specification
Length	Req	Set Length	See MXF Format Specification
Instance ID	Req	Unique ID of this instance	See MXF Format Specification
Edit Rate	Req	Edit Rate copied from the tracks of the Essence Container	See MXF Format Specification
Start Position	Req	The first editable unit indexed by this Index Table segment measured in File Package Edit Units	Calculate
Duration	Req	Time duration of this table segment measured in Edit Units of the referenced Package	Calculate
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	Set to the number of bytes in every KLV, including the length of the Key and Length. The Index Table can be used to find the first byte of the KLV of every DV-DIF frame
IndexSID	D/Req	Stream Identifier (SID) of Index Table	See MXF Format Specification
BodySID	Req	Stream Identifier (SID) of the indexed Essence Container	See MXF Format Specification

11.2 Clip wrapping with no other elements in the generic container

This is an even simpler case where the index table points to the first byte of each DV DIF frame of the DV-DIF compound element KLV triplet. There are no other generic container items and any sound or data information is embedded within the DV-DIF container. There is only one KLV Item.

Table 7 – Clip wrapped index table segment set

Item Name	Req ?	Meaning	Use
Index Table Segment	Req	An Index Table Segment set	See MXF Format Specification
Length	Req	Set Length	See MXF Format Specification
Instance ID	Req	Unique ID of this instance	See MXF Format Specification
Edit Rate	Req	Edit Rate copied from the tracks of the Essence Container	See MXF Format Specification
Start Position	Req	The first editable unit indexed by this Index Table segment measured in File Package Edit Units	Calculate
Duration	Req	Time duration of this table segment measured in Edit Units of the referenced Package	Calculate
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	Set to the number of bytes in the blocks of a DV-DIF frame. The Index Table can be used to find the first byte of the first DIF block every DV-DIF frame
IndexSID	D/Req	Stream Identifier (SID) of Index Table	See MXF Format Specification
BodySID	Req	Stream Identifier (SID) of the indexed Essence Container	See MXF Format Specification

11.3 Frame wrapping with extra items in the generic container

This is a more complex case where the index table points to the first byte of the DV-DIF compound element generic container key. The other generic container elements should be indexed by correct use of the delta entries and index entries.

Table 8 – Frame wrapped index table segment set example for figure 5

Item Name	Req ?	Meaning	Use
Index Table Segment	Req	An Index Table Segment set	See MXF Format Specification
Length	Req	Set Length	See MXF Format Specification
Instance ID	Req	Unique ID of this instance	See MXF Format Specification
Edit Rate	Req	Edit Rate copied from the tracks of the Essence Container	See MXF Format Specification
Start Position	Req	The first editable unit indexed by this Index Table segment measured in File Package Edit Units	Calculate
Duration	Req	Time duration of this table segment measured in Edit Units of the referenced Package	Calculate
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	0 unless the total length of all the GC Elements is of constant size. In this example we assume the Data Elements are VBR so the value is 0.
IndexSID	D/Req	Stream Identifier (SID) of Index Table	See MXF Format Specification
BodySID	Req	Stream Identifier (SID) of the indexed Essence Container	See MXF Format Specification
Slice Count	D/Req	Number of slices minus 1 (NSL)	0
PosTableCount	Opt	Number of PosTable Entries minus 1	0
Delta Entry Array	Opt	Map Elements onto Slices	Table 9
Index Entry Array	D/Req	Index from Edit Unit number to stream offset	Table 10

The delta entry array shall contain an entry for every indexed element in the generic container. The order of the elements in the delta entry array shall match the order of the elements in the generic container. The example below is a delta entry array designed to match the example in figure 5. Implementations should contruct a delta entry array according to the properties of the actual essence in the file.

In frame wrapping mode, the element delta values shall include the lengths of the "KL" for each element. The result is that each element delta shall point to the first byte of the key in the KLV which wraps an element. 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. In this example, we will assume that the data element in figure 5 is of variable length so that an index entry array is required.

Table 9 – Frame wrapped delta entry array example for figure 5

	Field Name	Type	Meaning	Use
	NDE	UInt32	Number of delta entries	4
	Length	UInt32	Length of each delta entry	6
Compund Delta Entry	PosTableIndex	Int8	0 = no reordering or offsets	0
	Slice	UInt8	Slice number in IndexEntry	0
	Element Delta	UInt32	Delta from start of slice to this Element	0
Sound Delta Entry	PosTableIndex	Int8	0 = no reordering or offsets	0
	Slice	UInt8	Slice number in IndexEntry	0
	Element Delta	UInt32	Delta from start of slice to this Element	sizeof(KL) + sizeof(DV-DIF frame)
Sound Delta Entry	PosTableIndex	Boolean	0 = no reordering or offsets	0
	Slice	UInt8	Slice number in IndexEntry	0
	Element Delta	UInt32	Delta from start of slice to this Element	sizeof(KL) + sizeof(DV-DIF frame) + sizeof(Sound Element 1)
Data Delta Entry	PosTableIndex	Int8	0 = no reordering or offsets	0
	Slice	UInt8	Slice number in IndexEntry	0
	Element Delta	UInt32	Delta from start of slice to this Element	sizeof(KL) + sizeof(DV-DIF frame) + sizeof(Sound Element 1) + sizeof(Sound Element 2)

Table 10 – Frame wrapped index entry array example for figure 5

	N	Field Name	Type	Meaning	Use
	1	NIE	UInt32	Number of index entries	=number of frames
	1	Length	UInt32	Length of each index array entry	calculate
One Index Entry for every frame	NIE	Temporal Offset	Int8	Offset in edit units from Display Order to Coded Order	0
		Anchor Offset	Int8	Offset in edit units to previous Anchor Frame. The value is zero if this is an anchor frame.	0
		Flags	EditUnitFlag	Flags for this Edit Unit Bit 7: Random Access Bit 6: Sequence Header Bit 5: P flag Bit 4: B flag 00== I frame 10== P frame 01== B frame 11== not used Bits 0-3: reserved	80h
		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 Compound Item in this frame as shown in Figure 5
		SliceOffset	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 it would not be present because there is only 1 slice
		PosTable	NPE *Rational	The fractional position offset from the start of the content package to the synchronized sample in the Content Package	Optional depending on the complexity of the synchronization required. In this case it would not be present because PosTableCount is 0

11.4 Clip wrapping with extra Items in the generic container

This is the most complex case where the index table points to the first byte of the payload of each element in the generic container key. The other generic container elements should be indexed by correct use of the delta entries and index entries. The delta entry values are limited to a UINT32 number range and will therefore be incapable of specifying the offset from the compound item of frame 1 to the sound item of frame 1. To accommodate this, the index table segment is sliced so that there is, in effect, a 64-bit variable offset from the DV-DIF frame to the corresponding sound item can be created. in clip wrapping where an index table is required, but 64-bit offsets are insufficient, the clip wrapped generic container shall be split into smaller clips.

Table 11 – Clip wrapped index table segment set example for figure 6

Item Name	Req ?	Meaning	Use
Index Table Segment	Req	An Index Table Segment set	See MXF Format Specification
Length	Req	Set Length	See MXF Format Specification
Instance ID	Req	Unique ID of this instance	See MXF Format Specification
Edit Rate	Req	Edit Rate copied from the tracks of the Essence Container	See MXF Format Specification
Start Position	Req	The first editable unit indexed by this Index Table segment measured in File Package Edit Units	Calculate
Duration	Req	Time duration of this table segment measured in Edit Units of the referenced Package	Calculate
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	0 unless the total length of all the GC Elements is of constant size. In this example we assume the Data Elements are VBR so the value is 0.
IndexSID	D/Req	Stream Identifier (SID) of Index Table	See MXF Format Specification
BodySID	Req	Stream Identifier (SID) of the indexed Essence Container	See MXF Format Specification
Slice Count	D/Req	Number of slices minus 1 (NSL)	3
PosTableCount	Opt	Number of PosTable Entries minus 1	0
Delta Entry Array	Opt	Map Elements onto Slices	Table 12
Index Entry Array	D/Req	Index from Edit Unit number to stream offset	Table 13

The delta entry array shall contain an entry for every indexed element in the generic container. The order of the elements in the delta entry array shall match the order of the elements in the generic container. The example below is a delta entry array designed to match the example in figure 6. Implementations should construct a eelta entry array according to the properties of the actual essence in the file.

In clip wrapping mode, the element delta values shall **not** include the lengths of the "KL" for each element. The result is that each element delta shall point to the first byte of the value in the KLV which wraps an element.

Table 12 – Clip wrapped delta entry array example for figure 6

	Field Name	Type	Meaning	Use
	NDE	UInt32	Number of delta entries	4
	Length	UInt32	Length of each delta entry	6
Compund Delta Entry	PosTableIndex	Int8	0 = no reordering or offsets	0
	Slice	UInt8	Slice number in IndexEntry	0
	Element Delta	UInt32	Delta from start of slice to this Element	0
Sound Delta Entry	PosTableIndex	Int8	0 = no reordering or offsets	0
	Slice	UInt8	Slice number in IndexEntry	1
	Element Delta	UInt32	Delta from start of slice to this Element	0
Sound Delta Entry	PosTableIndex	Int8	0 = no reordering or offsets	0
	Slice	UInt8	Slice number in IndexEntry	2
	Element Delta	UInt32	Delta from start of slice to this Element	0
Data Delta Entry	PosTableIndex	Int8	0 = no reordering or offsets	0
	Slice	UInt8	Slice number in IndexEntry	3
	Element Delta	UInt32	Delta from start of slice to this Element	0

Table 13 – Clip wrapped index entry array example for figure 6

	N	Field Name	Type	Meaning	Use
	1	NIE	UInt32	Number of index entries	=number of frames
	1	Length	UInt32	Length of each index array entry	calculate
One Index Entry for every frame	NIE	Temporal Offset	Int8	Offset in edit units from Display Order to Coded Order	0
		Anchor Offset	Int8	Offset in edit units to previous Anchor Frame. The value is zero if this is an anchor frame.	0
		Flags	EditUnitFlag	Flags for this Edit Unit Bit 7: Random Access Bit 6: Sequence Header Bit 5: P flag Bit 4: B flag 00== I frame 10== P frame 01== B frame 11== not used Bits 0-3: reserved	80h
		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 value of the DV-DIF KLV for the first frame to the first byte of the first block this DV-DIF frame in Figure 6
		SliceOffset	UInt32	The offset in bytes from the Stream Offset to the start of slice 1.	Offset from the first byte of the first block this DV-DIF frame to the first byte of the corresponding audio sample in the first Sound Element in Figure 6
		SliceOffset	UInt32	The offset in bytes from the Stream Offset to the start of slice 2	Offset from the first byte of the first block this DV-DIF frame to the first byte of the corresponding audio sample in the second Sound Element in Figure 6
		SliceOffset	UInt32	The offset in bytes from the Stream Offset to the start of slice 3	Offset from the first byte of the first block this DV-DIF frame to the first byte of the corresponding audio sample in the Data Element in Figure 6
		PosTable	NPE *Rational	The fractional position offset from the start of the content package to the synchronized sample in the Content Package	Optional depending on the complexity of the synchronization required. In this case it would not be present because PosTableCount is 0

Annex A (informative)
Bibliography

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

SMPTE 322M-1999 , Television — Format for Transmission of DV Compressed Video, Audio and Data Over a Serial Data Transport Interface.

SMPTE 382M, Television — Material Exchange Format (MXF) — Mapping AES3 and Broadcast Wave Audio into the MXF Generic Container

SMPTE 385M, Television — Material Exchange Format (MXF) — Mapping SDTI-CP Essence and Metadata into the MXF Generic container.

SMPTE EG 41, Television — Material Exchange Format (MXF) — MXF Engineering Guideline

SMPTE RP 224, SMPTE Labels Registry