RP 2002-2006

SMPTE RECOMMENDED PRACTICE

Content Specification on Solid State Media Card for DV/DV-Based Essence

Table of Contents

Approved
September 15, 2006

Foreword

SMPTE (the Society of Motion Picture and Television Engineers) is an internationally recognized standard 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 Administrative Practices.

Technology Committee W25 prepared SMPTE Recommended Practice RP 2002.

Introduction

The evolution of compression technology in the field of television has enabled new television signal processing devices. This includes new methods and applications for storing captured television signals.

Standardization efforts by the Society of Motion Picture and Television Engineers (SMPTE) has led to the development of a new file format for audio-visual applications, known as MXF (Material Exchange Format), which may be used in all stages of audio-visual signal processing.

The combination of these two technologies, signal compression and file format, are enabling the use of IT technology in network-based facilities, providing a new environment for flexible tools for Content creation, processing, storage and transfer.

The purpose of this document is to provide implementers and users with the specification of Content on a Solid State Media (SSM) Card allowing for interoperability and file interchange with other devices that usually edit or process the Content after it has been captured. This document defines the specification of Content on an SSM Card. The Content consists of Essence and Metadata, and the structure of the content as well as detailed specifications of the Essence and the Metadata are described in this document.

The Essence data consists of a video MXF file and multiple audio MXF files. The MXF file format is intended to contain not only essence (picture and sound) but associated Metadata as well. The video and audio essence data on an SSM Card are stored in separate MXF files using the OP-Atom operational pattern as specified in SMPTE 390M.

Metadata stored in the Header of the MXF files contains the MXF Structural Metadata which describes the structure or parameters of the video and audio essence and includes optional MXF Descriptive Metadata which provides additional information about the essences.

In addition, Metadata is also described using XML (eXtensible Markup Language) that is stored in a separate Clip Metadata file. Descriptions using XML are human readable, computer compatible and easily extensible. The Clip Metadata file contains the Clip Structural Metadata which is the same information as required in the MXF Structural Metadata mentioned above, describing the structure or parameters of the video and audio essence. The Clip Metadata file also contains the Clip Descriptive Metadata which describes additional information about the essences.

There is a complementary SMPTE document (SMPTE RP 2006) which describes the physical and electrical interfaces of the storage media (SSM Card) to ensure interoperability at the storage level.

Users who are interested in additional information on MXF are encouraged to begin with SMPTE Engineering Guideline EG 41 which provides the basic explanation of the MXF file format.

1 Scope

This document defines the specification of Content on an SSM (Solid State Media) Card defined in SMPTE RP 2006 for interoperability and interchange of this Content with other devices. The Content consists of Essence and Metadata and this document specifies the structure of Content as well as detailed specifications of Essence and Metadata. The Essence data consists of a video MXF file and audio MXF files each of which uses operational pattern 'OP Atom' as specified in SMPTE 390M. The video MXF file contains DV/DV-based compressed video conforming to SMPTE 383M, and multiple audio MXF files contain one individual channel of audio data conforming to SMPTE 382M. The Metadata is stored in the header metadata of the MXF files and additionally stored in a separate metadata file that is defined using XML (eXtensible Markup Language).

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 12M-1999, Television, Audio and Film — Time and Control Code

SMPTE 314M-2005, Television — Data Structure for DV-Based Audio, Data and Compressed Video — 25 and 50 Mb/s

SMPTE 330M-2004, Television — Unique Material Identifier (UMID)

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

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

SMPTE 379M-2004, Television — Material Exchange Format (MXF) — Generic Container Format

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

SMPTE 383M-2004, Television — Material Exchange Format (MXF) — Mapping DV-DIF Data to the MXF Generic Container

SMPTE 390M-2004, Television — Material Exchange Format (MXF) — Specialized Operational Pattern "Atom" (Simplified Representation of a Single Item)

SMPTE RP 2006-2006, Solid State Media (SSM) Card Specification

AES3-2003, Serial Transmission Format for two Channel Linearly Represented Digital Audio Data

IEC 61834-2 (1998-08), Recording — Helical-Scan Digital Video Cassette Recording System using 6,35 mm Magnetic Tape for Consumer Use (525-60, 625-50, 1125-60 and 1250-50 Systems) — Part 2: SD Format for 525-60 and 625-50 Systems

ISO/IEC 646:1991, Information Technology – ISO 7-Bit Coded Character Set for Information Interchange

ISO/IEC 14496-2:2004, Information Technology — Coding of Audio-Visual Objects — Part 2: Visual

ISO/IEC 14496-3:2005, Information Technology — Coding of Audio-Visual Objects — Part 3: Audio

ISO/IEC 14496-12:2005, Information Technology — Coding of Audio-Visual Objects — Part 12: ISO Base Media File Format

Extensible Markup Language (XML) 1.0 (Third Edition), W3C Recommendation, 04 February 2004

Namespaces in XML, W3C Recommendation, 14 January 1999

XML Schema Part 1: Structures Second Edition, W3C Recommendation, 28 October 2004

XML Schema Part 2: Datatypes, W3C Recommendation, 28 October 2004

3 Definitions, symbols and abbreviations

For symbol definition and abbreviations about MXF, the reader is encouraged to study the SMPTE 377M and SMPTE EG 41. This document uses same terms defined in the mentioned standards and in other referenced documents.

Content	Content comprises video or audio essence and any Metadata.
Clip Content	Clip Content indicates the Content which contains the video or audio essence and related Metadata in one Clip.
Clip Metadata	Metadata which is stored in a Clip Metadata file using XML
Clip Descriptive Metadata	Metadata which is stored in a Clip Metadata file using XML and describes additional information about the video and audio essence
Clip Structural Metadata	Metadata which is stored in a Clip Metadata file using XML and describes the structure or parameters of the video and audio essence.
MXF Descriptive Metadata	Metadata which is stored in a MXF file using the KLV coding and describes additional information about the video and audio essence.
MXF Structural Metadata	Metadata which is stored in a MXF file using the KLV coding and describes the structure or parameters of the video and audio essence.
WORD	unsigned short integer
DWORD	unsigned long integer
LONG	signed long integer

4 Data structure of Content stored on an SSM Card

4.1 Overview

This section describes the data structure of Content on an SSM Card. The SSM Card format is defined in SMPTE RP 2006. This document defines the components of the Content, the concept of the Metadata structure, the directory structure, the files within each directory, and the naming rules of the files.

4.2 Content Structure

The content on an SSM Card shall consist of separate files of Essence data and Metadata, structured as Clip Contents. The Essence data shall consist of one video MXF file and up to 16 individual audio MXF files each containing a single audio channel. The Metadata shall be stored in the Header Metadata of the MXF files, and shall be additionally stored in a Clip metadata file.

The metadata stored in the Header Metadata part of the MXF files comprises MXF Structural Metadata which describes the structure and parameters of the video and audio essence, and optional MXF Descriptive Metadata which provides additional information about the Content.

Keeping Metadata in the header of a MXF file provides great benefits in file exchange. The relationship between video/audio essence and Metadata can be always maintained by wrapping the essence data and associated Metadata together in one file. In addition, the KLV coded Metadata structure, based on the SMPTE Metadata dictionary, guarantees the interoperability of Metadata items between different systems. However, for direct editing capability using a fast random access media like a SSM Card, having Clip Metadata file outside a MXF file gives benefits for users. MXF metadata editing after capture, requires addition or modification of embedded MXF metadata items. This editing may sometimes result in the re-creation of MXF essence files. The provision of an external Clip Metadata structure provides direct editing capability of Metadata items without changes to the essence files.

For ease of editing, an external Clip Metadata file is defined as part of the Content Data Structure on a SSM Card. A portion of the Clip Metadata file contains Clip Structural Metadata which corresponds to the MXF Structural Metadata in the header portion of the MXF files. Another portion contains the Clip Descriptive Metadata which provides additional information, in human readable form, regarding the stored Clip Content.

Figure 4.1 illustrates the general structure of a Clip Content as stored on the SSM Card.

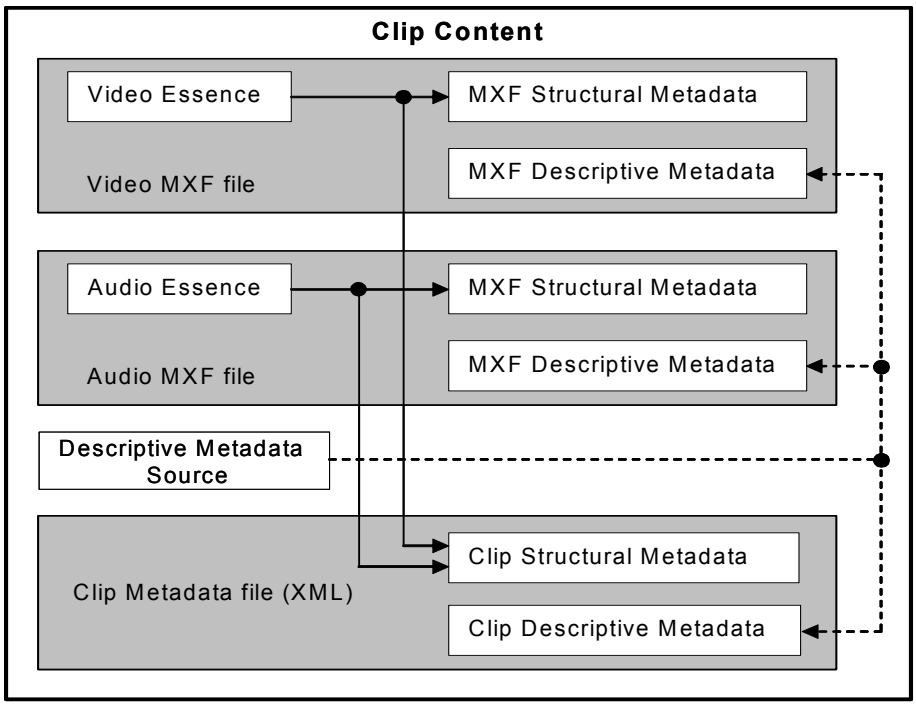


Figure 4.1 – General Structure of Content on an SSM Card

4.3 File Directory

All SSM Content files shall be organized on the SSM Card in a Directory structure that is shown in Figure 4.2. The Directory hierarchy of the Clip Content shall be "/" and this shall represent the root of all stored Content on the SSM Card.

4.3.1 Contents directory

The "CONTENTS" directory shall be located under the root directory "/" and shall contain the "CLIP" metadata directory, "VIDEO" essence directory, and "AUDIO" essence directory.

The "CONTENTS" directory may optionally contain "ICON", "VOICE" and "PROXY" directories.

4.3.2 Clip directory

The "CLIP" directory shall contain all Clip Metadata files that store the Clip Structural and Descriptive Metadata and is located under the "CONTENTS" directory.

4.3.3 Video directory

The "VIDEO" directory shall contain all video essence files including the MXF Header Metadata.

4.3.4 Audio directory

The "AUDIO" directory shall contain all audio essence files including the MXF Header Metadata.

4.3.5 File directory for optional files

The "ICON" directory shall contain all the Thumbnail files, the "VOICE" directory shall contain all the Voice Memo files and the "PROXY" directory shall contain all the Proxy files.

Definitions of the Thumbnail, Voice memo and Proxy files are included in Annex B.

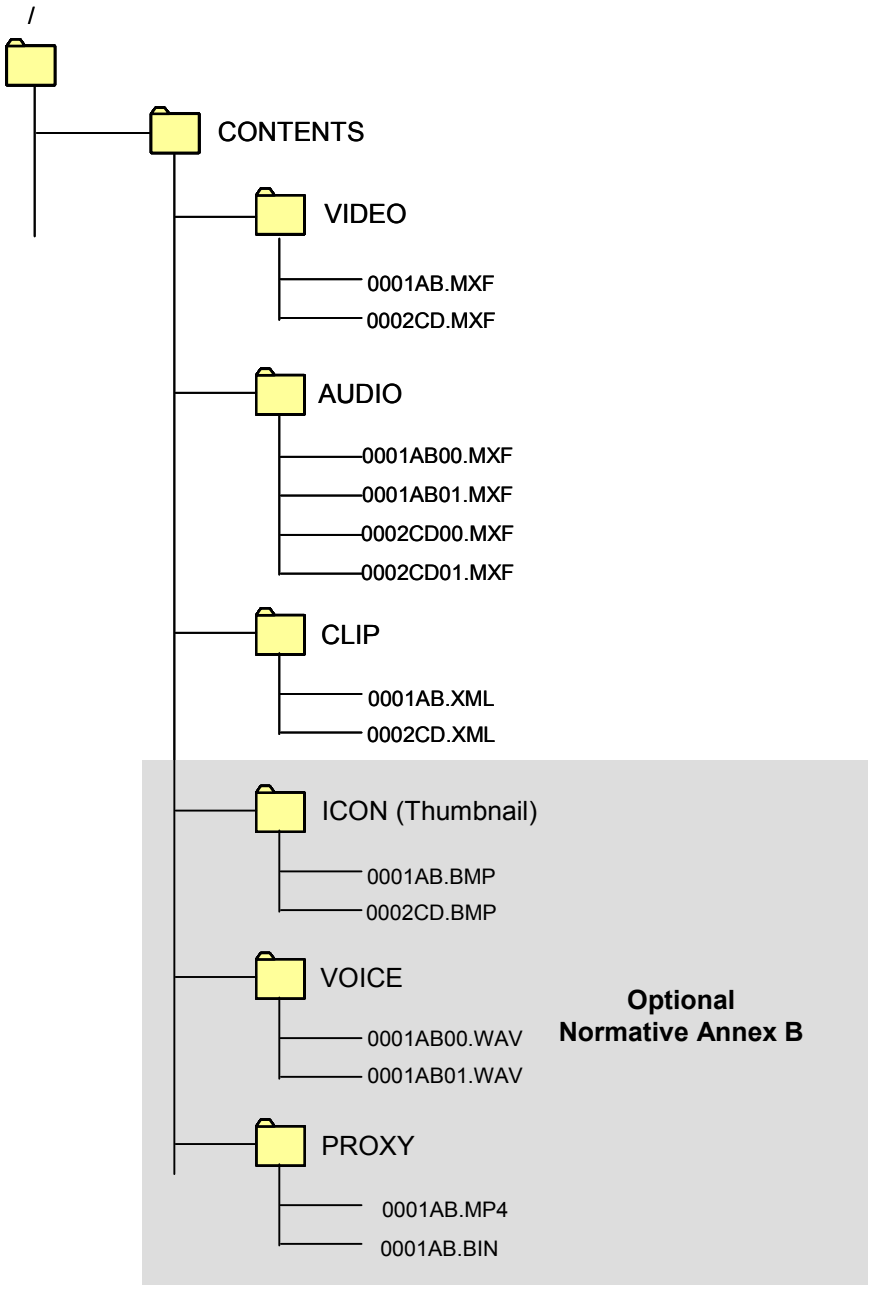


Figure 4.2 – An example of Directory Tree for Clip (0001AB) and (0002CD) with two Audio channels

4.4 Files

All Essence files in the same Clip Content shall have the same duration. The structure of MXF files is defined in section 5.

4.4.1 Video Essence file

A Video Essence file shall comply with the MXF file structure using the DV-DIF mapping defined in SMPTE 383M. The Video Essence file is defined in section 6.

4.4.2 Audio Essence file

Each Audio Essence file shall comply with the MXF file structure using the AES 3 audio mapping defined in SMPTE 382. Each Audio Essence file shall map a single channel of AES3 audio. The maximum number of the Audio Essence files per Clip shall be 16 channels. The Audio Essence file is defined in section 7.

4.4.3 Clip Metadata file

A Clip Metadata file shall consist of Clip Structural Metadata and Clip Descriptive Metadata. Both Clip Metadata types shall be described with an XML language and contained in a single file using XML Schema syntax and semantics. The Clip Structural Metadata provides information about Video and Audio essences, while the Clip Descriptive Metadata defines additional data about the Clip. The Clip Metadata file is mandatory to assure interoperability with other equipment. The Clip Metadata file is defined in section 8, and informative section 9 provides information about how this Clip Metadata can be mapped to SMPTE 380M (MXF DMS-1).

4.4.4 Optional files

The SSM Card may also include additional information about the Video and Audio Essences as application specific files. The use of this additional information (Thumbnail file, Voice-memo file and Proxy file) is optional, and descriptions of each file are included in Annex B.

4.4.5 Naming rules of files

The name shall consist of six or eight characters and an extension. Six characters shall be used for a Video Essence file and a Clip Metadata file. Eight characters shall be used for Audio Essence files. The extension shall be 3 characters. The file name and the extensions shall be case-sensitive and expressed with upper case.

- The first six characters shall represent a Clip name.

  The first four characters shall be the number such as “0001” and shall be unique within the directory.

  The next two characters have a value of “0” to “9” or “A” to “Z” such as “AB”.

  The Clip name of the files contained in a Content shall be the same.

  The Clip name shall be the same as the ClipName element value defined in section 8.3.3.

- The next two characters shall represent the channel number of the Audio Essence file and shall have a value of “00” to “15” in decimal representation.

- The extension of the Video Essence file and the Audio Essence file shall be “.MXF”.

- The extension of the Clip Metadata file shall be “.XML”.

The name of the Video Essence file consists of a Clip name (0001AB) and a file extension (.MXF) in form of “0001AB.MXF”.

The name of the Audio Essence file consists of a Clip name (0001AB), channel number, and file extension (.MXF) in form of “0001AB00.MXF”.

The name of the Clip Metadata file consists of a Clip name (0001AB) and a file extension (.XML) in the form of “0001AB.XML”.

Names and extensions of the optional files are given in Annex B.

This unique Clip name for each Clip can be used to link all the files contained in the same Clip Content.

5 Structure of MXF file for an SSM Card

Video and Audio Essence files shall be compliant with the MXF file format, the MXF Generic Container and OP-Atom as defined in SMPTE 377M, SMPTE379M and SMPTE 390M respectively.

Figure 5.1 shows the basic structure of a video or audio MXF essence file. The Header Partition shall include the Header Metadata and an Index Table. The Body Partition is required and shall include the complete Essence Container. The Footer Partition shall include an Index Table. The Index Table in the Header Partition shall be considered as a copy of that in the Footer Partition.

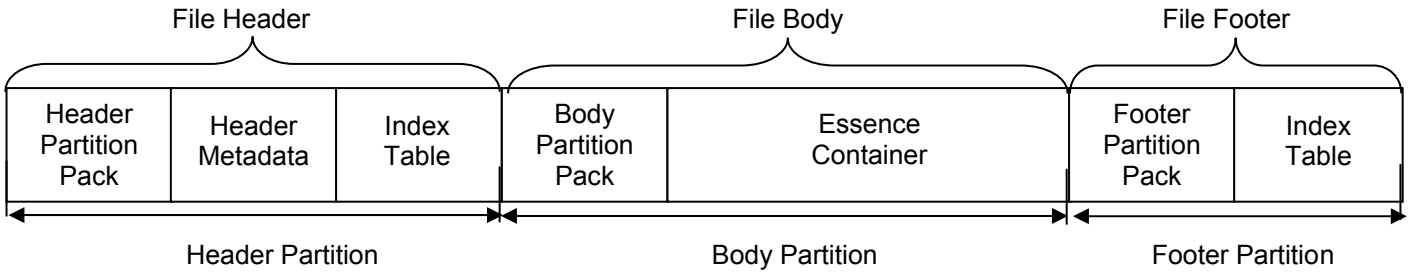


Figure 5.1 – Basic Structure of an MXF file

NOTE – The Footer Partition may optionally be followed by a Random Index Pack.

All data within an MXF file is KLV coded as defined in SMPTE 336M.

While the information provided in this section is defined in SMPTE 377M, it is provided to show the specific use of the file format as is used by this application.

5.1 Partition Packs

Partition Packs, used in a Header, Body and Footer, have many common Items regardless of where they are used; however some Item values are different such as BodySID values. These common values are duplicates of values defined in SMPTE 377M, tables 1 through 5.

5.1.1 KLV Alignment Grid requirement (KAG)

A specific KAG value is not defined. (The value is set to the default '1').

NOTE – KLV Fill items may be used to provide alignment grid when required.

5.2 Header Metadata

The Header Metadata part of the Header Partition contains the MXF Structural metadata related to the structure and capabilities of the MXF file used in the SSM Card application. The Metadata Sets for the MXF Header Metadata section are illustrated in Figure 5.2.

The Header Partition Pack shall be Closed and Complete as defined for the Operational Pattern Atom. Therefore, Header Metadata values (Required and Best Effort) shall be completed.

Tables A1 through A15, shown in informative Annex A, are provided for reader information and define the values for Required and Best Efforts properties in each metadata set.

NOTE – Optional properties, not shown in Annex A, may be present in the file.

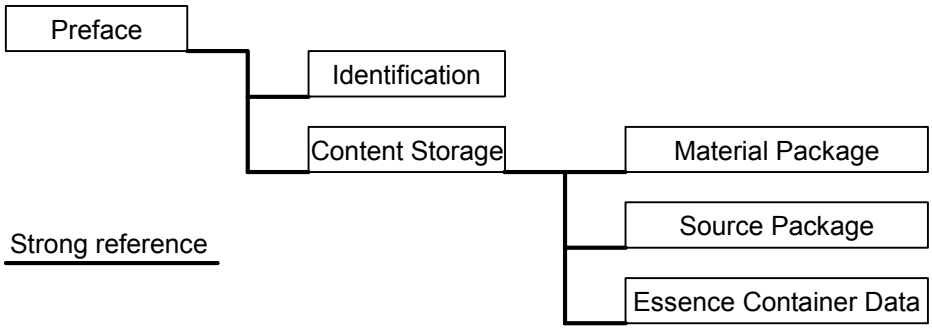


Figure 5.2 – Metadata Set for MXF Header Metadata

5.2.1 Key Values for Structural Metadata Sets

The Key values for Structural Metadata sets used in this standard are defined in SMPTE 377M.

5.2.2 Primer Pack

The Primer Pack is a look up table that serves to define the UL of any property within a local set. The definition is in SMPTE 377M.

5.2.3 Preface Set

The Preface Set defines version and contents of the MXF Structural metadata. The Required and Best Effort properties in this set are shown in Annex A, Table A.1.

5.2.4 Identification Set

The Identification Set provides information about tools that were used when the file was created and/or modified. The Required and Best Effort properties in this set are shown in Annex A, Table A.2.

5.2.5 Content Storage Set

The Content Storage Set identifies all Packages used in the file. The Required and Best Effort properties in this set are shown in Annex A, Table A.3.

5.2.6 Essence Container Data Set

The Essence Container Data Set is linked to the corresponding File Package via a common PackageID value (as a basic UMID). Each Essence Container data set contains a BodySID and an IndexSID, which identify respectively, the file partition for the essence and the file partition for the Index Table. The Required and Best Effort properties in this set are shown in Annex A, Table A.4.

5.2.7 Material Package

There shall be one Material Package as specified in SMPTE 390M. The Material Package in general represents an output time line of a MXF file stored on the SSM Card. It should contain three or more tracks: a Timecode track, a Picture track and one or more Sound tracks. The Timecode track further consists of a Timecode Track Set, a Timecode Sequence Set and a Timecode Component Set. The Picture track consists of a Picture Track Set, a Picture Sequence Set and a Picture SourceClip Set. Each Sound track consists of a Sound Track Set, a Sound Sequence Set and a Sound SourceClip Set. The Required and Best Effort properties in these sets are shown in Annex A, Table A.5 through A.14.

The Video Essence file and each of the Audio Essence files have a Material Package which contains the Timecode track Set, the Picture track Set and the Sound track Sets.

The structure of the Material Package in the Video Essence file and each Audio Essence file shall be identical.

The SourceClip of the Picture track set in the Video Essence file shall reference the Source package in the Video Essence File via the PackageID value. The SourceClip of the Sound track sets in the Video Essence file shall reference the Source packages for all the Audio Essence Files via the PackageID values.

The SourceClip of the Sound track set in the Audio Essence files shall reference the Source packages in all the Audio Essence Files via the PackageID values. The SourceClip of the Picture track set in the Audio Essence files shall reference the Source package in the Video Essence File via the PackageID value.

The Material Package in each file has an Essence Container for the embedded essence (picture or sound) and those tracks whose essence is not embedded can only be resolved by using the SourcePackageID value. The scope of the search shall be limited to the essence files within a Clip.

5.2.8 Source Package

5.2.8.1 Top-level File Package

In each file, there shall be one Top-level File Package as specified in SMPTE 390M. The Top-level File package represents storage of all essences. It is composed of a Picture Track, a Picture Sequence and a Picture SourceClip set for the Video Essence file, and a Sound Track, a Sound Sequence and a Sound SourceClip set for the Audio Essence file. In addition, the Video and Audio Essence files contain a Picture Essence Descriptor and a Sound Essence descriptor, respectively. The SourceClip in the Material package of the file directly references this set. The Top-level File Package property list is the same as the Material package as shown in Annex A, Table A.15 through A.21.

5.2.8.2 Lower-level Source Package

Lower-level Source Package may be present to provide historical annotations.

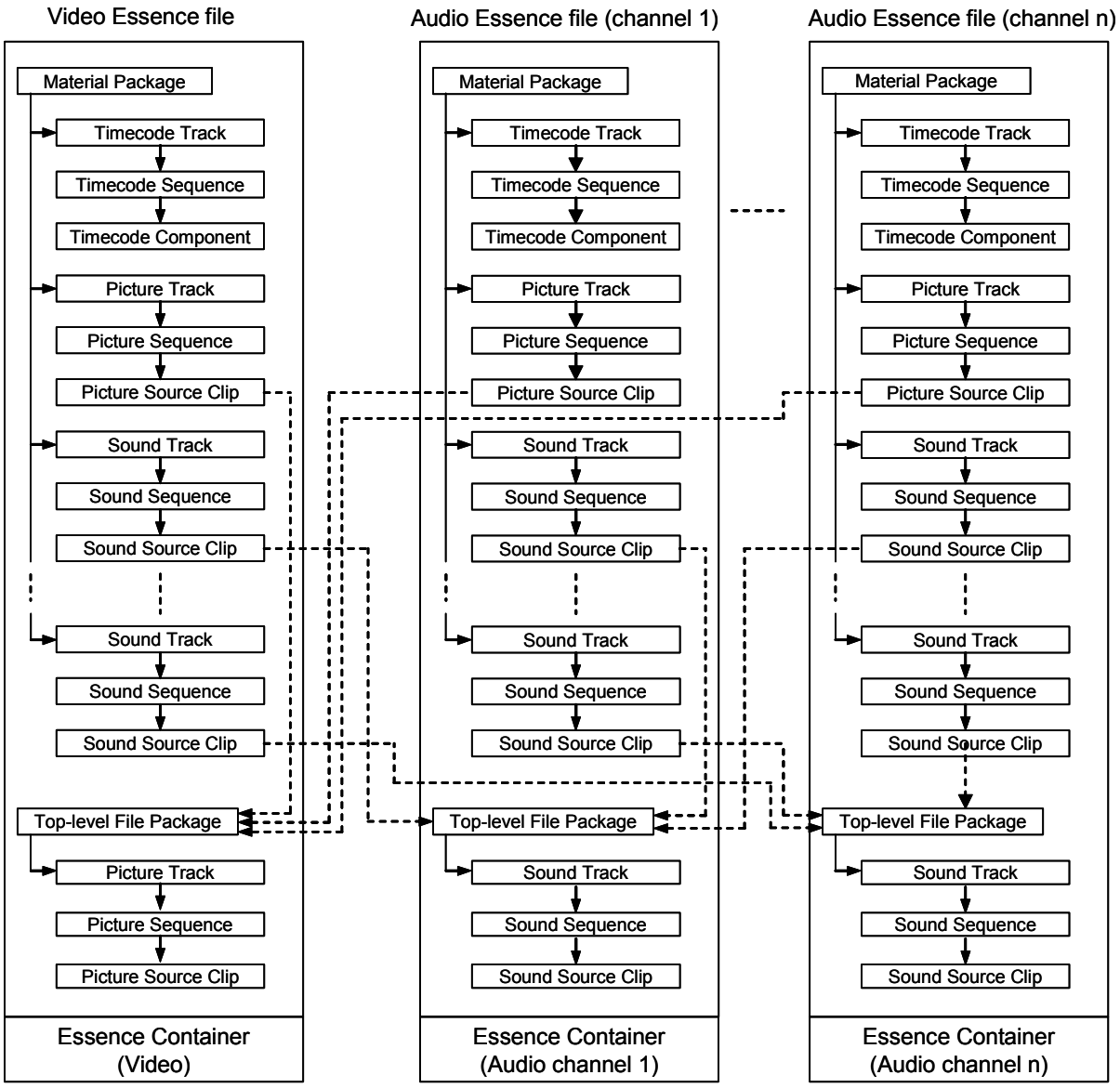


Figure 5.3 – Structure of Material Package and Top-level File Package

6 Video Essence file

6.1 Video data format

The video data format shall comply with SMPTE 314M for DV-based data, and with IEC 61834-2 for DV-compliant data. The basic format of this data is in a form of DIF frame sequences for both signal variants. The supported data rate for these signals is 25Mb/s for DV-based and DV-compliant type; 50Mb/s for DV-based type.

The mapping of the video data to the MXF Generic Container shall comply with SMPTE 383M.

A DV-DIF stream may contain audio data. If audio data is present in the DV-DIF stream, decoders shall treat this audio data as invalid. Encoders should not embed valid audio data in a DV-DIF stream.

6.1.1 Structure of the Essence Container

The structure of the DV-DIF Compound Item Essence Container in a file is shown below in Figure 6.1.

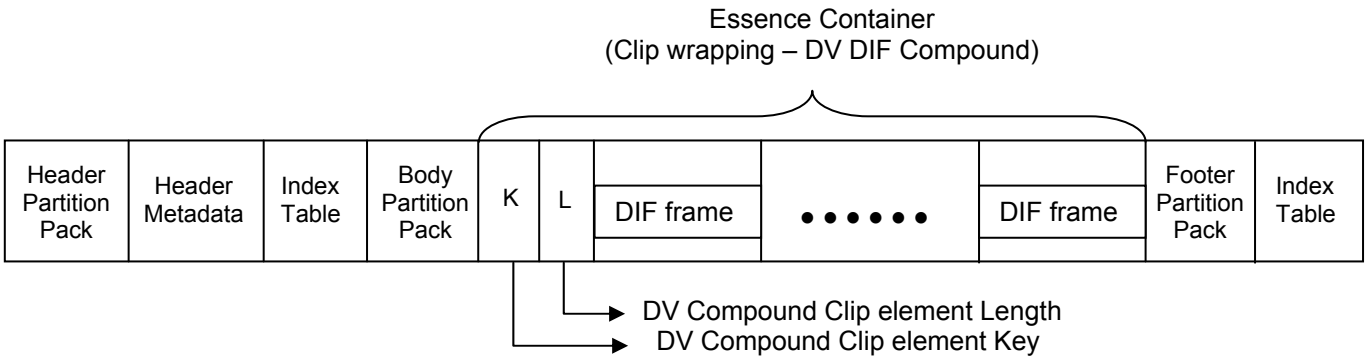


Figure 6.1 – Structure of a DV-DIF compound Essence file

6.1.2 SMPTE Essence Container identification label

The basic SMPTE label value for Essence Container identification is defined in SMPTE 379M. The specific Label for the DV-DIF Essence Container for SSM Card storage is shown below in Table 6.1 and is used in the Partition Pack, Preface Set and in the picture Essence Descriptor.

Table 6.1 – UL 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 6.2
16	Containment of the DV	02h	Clip Wrapping

Byte 14 of the Essence Container label identifies the container as DV-DIF Compound mapping into and Generic Container defined in SMPTE 379M.

Byte 15 provides information about the actual DV type signal that is carried in the container, and its value is set accordingly to Table 6.2 shown below.

Table 6.2 – Definition of byte 15 of UL Essence Container label for a DV-type signal

Byte 15 Value	DV type	Content type
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
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

Byte 16 is intended to carry basic information about the containment of the DV format and shall be set to 02h to indicate clip wrapping.

6.2 DV-DIF Compound data

6.2.1 DV-DIF Compound Element Key

The Key values for compound essence elements is defined in SMPTE 379M and for the DV-DIF data is shown in Table 6.3.

Table 6.3 – Key value for DV-DIF compound element

Byte No.	Description	Value (hex)	Meaning
1-12	Specified by the MXF Generic Container Specification SMPTE 379M		
13	Item Type Identifier	18h	Compound Item
14	Essence Element Count	01h	Count of Compound Elements in this Item
15	Essence Element Type	02h	DV-DIF Compound Clip Wrapped Element as listed in SMPTE RP224
16	Essence Element Number	01h	A number (used as an Index) of this Compound Item in this Generic Container as defined in SMPTE 379M

6.2.2 DV-DIF Compound element Length

The length of the KLV coded element field shall be 8 bytes long, encoded according to BER long-form rules defined in SMPTE 336M. The format of the length field is in form 87h.aa.bb.cc.dd.ee.ff.gg.

6.2.3 DV-DIF Compound element Value

The data of the DV-DIF compound element are Clip wrapped DV-DIF data bytes of multiple frames as defined in SMPTE 314M or IEC 61834-2.

6.2.4 CDCI (Color Difference Component Image) Picture Essence Descriptor

The use of the CDCI Picture Essence Descriptor is described in SMPTE 383M. Table 6.4 shown below provides for all Required and Best Effort Properties as indicated in the SMPTE 377M for the CDCI Picture Essence Descriptor.

NOTE – The optional properties are not shown in Table 6.4, but may be present in the file.

Table 6.4 – CDCI Picture Essence Descriptor Property list

Item Name	Type	Value	
		525/59.94i	625/50i
CDCI Essence Descriptor	Set Key	Set Key, Set Length and Instance UID values are defined in 377M	
Length	BER Length		
Instance UID	UUID		
Sample Rate	Rational	30000,1001	25,1
Container Duration	Length	Number of frames present	
Essence Container	UL	See Table 6.1	
Frame Layout	Uint8	1	
Stored Width	Uint32	720	
Stored Height	Uint32	240	288
Aspect Ratio	Rational	4,3 or 16,9	
Video Line Map	Array of Int32	23,285	23,335
Picture Essence Coding	UL	See Table 6.5	
Component Depth	UInt32	8	
Horizontal Subsampling	UInt32	IEC DV 525/59.94i, DV-based 25Mbps : 4 Others : 2	
Vertical Subsampling	UInt32	IEC DV 625/50i : 2 Others : 1	

Values of UL that are entered into the CDCI Picture Essence Descriptor for SSM Card storage application for a Picture Essence Coding for DV formats are shown in Table 6.5.

Table 6.5 – UL values for a Picture Essence Coding

UL	Content type
06.0E.2B.34.04.01.01.01.04.01.02.02.02.01.01.00	IEC DV compressed at 25Mbps, 525/59.94-I source
06.0E.2B.34.04.01.01.01.04.01.02.02.02.01.02.00	IEC DV compressed at 25Mbps, 625/50-I source
06.0E.2B.34.04.01.01.01.04.01.02.02.02.02.01.00	DV-based compressed at 25Mbps, 525/59.94-I source
06.0E.2B.34.04.01.01.01.04.01.02.02.02.02.02.00	DV-based compressed at 25Mbps, 625/50-I source
06.0E.2B.34.04.01.01.01.04.01.02.02.02.02.03.00	DV-based compressed at 50Mbps, 525/59.94-I source
06.0E.2B.34.04.01.01.01.04.01.02.02.02.02.04.00	DV-based compressed at 50Mbps, 625/50-I source

6.2.5 Index table for Video Essence file

An Index Table is placed in the Header and Footer Partition as required by Operational Pattern Atom. The Index Table in the Essence file for the SSM Card indexes the first byte of each frame of the DV-DIF Compound Element.

NOTE – In Clip-wrapping, the index table indexes from the start of the essence Value and skips the Key and Length values.

A 16-byte SMPTE Universal Label identifies an Index Table Segment and acts as a Key for the Index Table Segment Set. The value of the Index Table Segment Key is shown below in Table 6.6.

Table 6.6 – Index Table Segment Key

Byte No.	Description	Value (hex)	Meaning
1	Object Identifier	06h	
2	Label size	0Eh	
3	Designator	2Bh	ISO
4	Designator	34h	SMPTE
5	Registry Category Designator	02h	Sets & Packs
6	Registry Designator	53h	Local Sets (2 byte tags, 2 byte lengths)
7	Structure Designator	01h	Set/Pack registry
8	Version Number	01h	Registry Version 1
9	Item Designator	0Dh	Organizationally registered
10	Organization	01h	AAF Association
11	Application	02h	MXF File Structure
12	Structure Version	01h	Version 1
13	Structure Kind	01h	File Structure sets & packs
14	Set/Pack Kind	10h	Index Table Segment
15	Version	01h	Index Table Specification version
16	Reserved	00h	

The Index Table Segment Set and values for Video are shown in Table 6.7. Delta Entry Array table is not required for this application in which the Essence Container consists only of a single item as it is in SSM Card storage case. As defined in SMPTE 383M, the DV-DIF Compound Essence has a static structure and therefore specific Table Segmentation is not required.

Table 6.7 – Index table Segment set

Item Name	Type	Value
Index Table Segment	Set Key	Set Key, Set Length and Instance UID values are defined in 377M
Length	BER Length	
Instance ID	UUID	
Index Edit Rate	Rational	Set to the same value as the Edit Rate defined in the Picture track in the Packages
Index Start Position	Position	Set to zero
Index Duration	Length	Set to zero to indicate that Index table Segment is applicable to the entire Essence Container identified by the BodySID
Edit Unit Byte Count	UInt32	Set to the number of bytes in a DV-DIF frame (e.g. 120kByte for 25Mbps, 525-I)
IndexSID	UInt32	Set to the same value as the IndexSID in the Header and Footer Partition Pack
BodySID	UInt32	Set to the same value as the BodySID in the Body Partition Pack

7 Audio Essence file

7.1 Audio data format

The audio data format carried in the Audio Essence Container shall be either 16-bit or 24-bit PCM audio sampled at 48 kHz. The signal format shall comply with the AES3-2003 (R1997) standard.

7.1.1 Structure of the Essence Container

The Essence Container for AES3 Audio Essence files shall be as defined in SMPTE 382M. The audio data mapped into the Essence Container shall use clip wrapping to form an Audio Essence file clip that is shown in Figure 7.1 below. Each audio channel of the AES3 audio stereo pair shall be wrapped into a separate Essence container and forms an independent file.

NOTES

1 When the frame rate is 30000/1001 Hz, the duration of each Audio Essence shall be the same as that of the video essence over a repeating 5-frame period starting with the first frame of the clip.

2 All audio channels shall be sample aligned starting from the first sample of each audio clip.

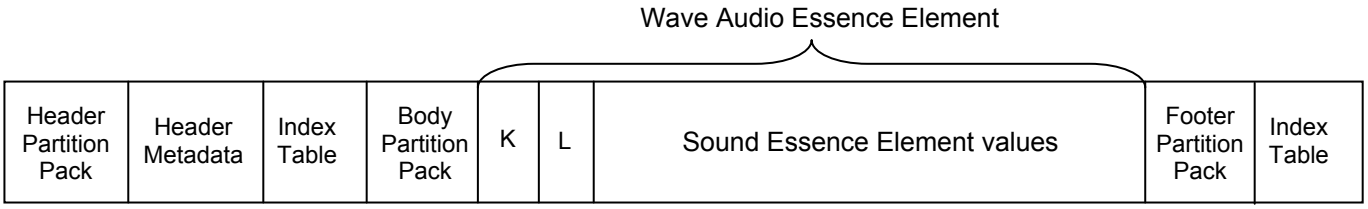


Figure 7.1 – Structure of a single Audio Essence file Clip

7.1.2 SMPTE Essence Container Identification label

The SMPTE label for Essence Container identification is defined in SMPTE 379M. The specific Label for the Sound Essence element for SSM Card audio files shown below in Table 7.1 is used in the Partition Pack, Preface Set and in the Sound Essence Descriptor which is a part of the file Header Partition.

Table 7.1 – UL 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	06h	AES3-BWF as listed in SMPTE RP224
15	Content Kind	04h	AES Clip Wrapped Element
16	Reserved	00h	

7.2 Sound Essence data

7.2.1 Audio Essence element Key

The Key value for the audio Essence element is defined in SMPTE 379M where the last four bytes (byte 13 thru 16) are defined in SMPTE 382M and the value for SSM Card use is shown below in Table 7.2.

Table 7.2 – Key value for Sound Essence element

Byte No.	Description	Value (hex)	Meaning
1-12	Specified by the MXF Generic Container Specification SMPTE379M		
13	Item Type Identifier	16h	GC Sound item
14	Essence Element Count	01h	One Essence Element present
15	Essence Element Type	04h	AES Clip Wrapped Element
16	Essence Element Number	00h	Unique Number

7.2.2 Sound Essence element Length

The length of the KLV coded element field shall be 8 bytes long, encoded according to BER long-form rules defined in SMPTE 336M. An example for a Clip wrapping, the format of the length field is 87h.aa.bb.cc.dd.ee.ff.gg.

7.2.3 Sound Essence element Value

The Clip wrapped data of audio samples are stored with either 16-bit or 24-bit resolution. Each audio sample is formatted in such a way that the minimum number of bytes is needed for each audio sample and is stored as an integer. As a result, the 16-bit audio sample is formatted into 2 bytes, and the 24-bit audio sample is formatted into 3 bytes with the least significant byte first.

7.2.4 Sound Essence Descriptor

The Sound Essence descriptor shall be the AES3 Audio Essence Descriptor (subclass of the Wave Audio descriptor) defined in SMPTE 382M. The Table 7.3 shown below shows all of the Required and Best Effort Properties as indicated in SMPTE 382 for the Sound Essence Descriptor.

NOTE – The optional properties are not shown in Table 7.3, but may be present in the file.

Table 7.3 – AES3 Sound Essence descriptor Item list

Item Name	Type	Value
AES3 Audio Essence Descriptor	Set Key	Set Key, Set Length and Instance UID values are defined in 377M
Length	BER Length	
Instance UID	UUID	
SampleRate	Rational	48000,1 (48 kHz)
Container Duration	Length	Number of samples present
Essence Container	UL	See Table 7.1
Audio sampling rate	Rational	48000,1 (48 kHz)
Locked/Unlocked	Boolean	1 (Locked)
ChannelCount	Uint32	1
Quantization bits	Uint32	16 bits : 16, 24 bits : 24
BlockAlign	Uint16	16 bits : 2, 24 bits : 3
AvgBps	Uint32	16 bits : 96000, 24 bits : 144000

7.2.5 Index table for an Audio Essence file

An index table is placed into the Header and Footer Partition as specified in SMPTE 390M. The Index table Segment Key (Table 6.6) and Index table Segment Set (Table 6.7) are identical as for a Video Essence file (Section 6.2.5). In general, there is no specific requirement for Index table Segment set in the SMPTE 382M for AES3 and BWF mapping format and Index table segment are set to the values as shown in Table 7.4.

Table 7.4 – Index table Segment set

Item Name	Type	Value
Index Table Segment	Set Key	Set Key, Set Length and Instance UID values are defined in 377M
Length	BER Length	
Instance ID	UUID	
Index Edit Rate	Rational	Set to the same value as the Edit Rate defined in the Sound track in the Packages
Index Start Position	Position	Set to zero
Index Duration	Length	Set to zero to indicate that Index table Segment is applicable to the entire Essence Container identified by the BodySID
Edit Unit Byte Count	UInt32	Set to the number of bytes in an audio sample for a Clip wrapping (e.g. 2Bytes for 16-bit audio, 3Bytes for 24-bit audio sample)
IndexSID	UInt32	Set to the same value as the IndexSID indicated in the Header and Footer Partition Pack
BodySID	UInt32	Set to the same as in the Body Partition Pack

8 Clip Metadata file (XML)

The following section defines the Clip Structural and Descriptive Metadata as an XML file.

This information enables users of other applications to read and process the Clip Metadata file (XML) associated with the MXF essence files (video and sound) stored on an SSM Card.

Figure 8.1 and Figure 8.2 show the overview of the Clip Structural Metadata elements and the Clip Descriptive Metadata elements.

Any Clip Metadata elements that duplicate a provision in the Header Metadata shall contain the same value. If the MXF Structural Metadata is updated in any one of the MXF files, then the corresponding Clip Structural Metadata in a Clip metadata file shall be updated and vice versa. The relationship of the Descriptive metadata between MXF and XML files is described in section 9.

NOTES

1 If there is a difference in the Descriptive Metadata elements, the value in the Clip Metadata should take precedence.

2 Since the name of the Card "SSM" was not decided yet as the SMPTE standard before starting implementations, the expression “P2 (Professional Plug-in)” is applied instead of the expression “SSM” in the tables and the XML schemas below.

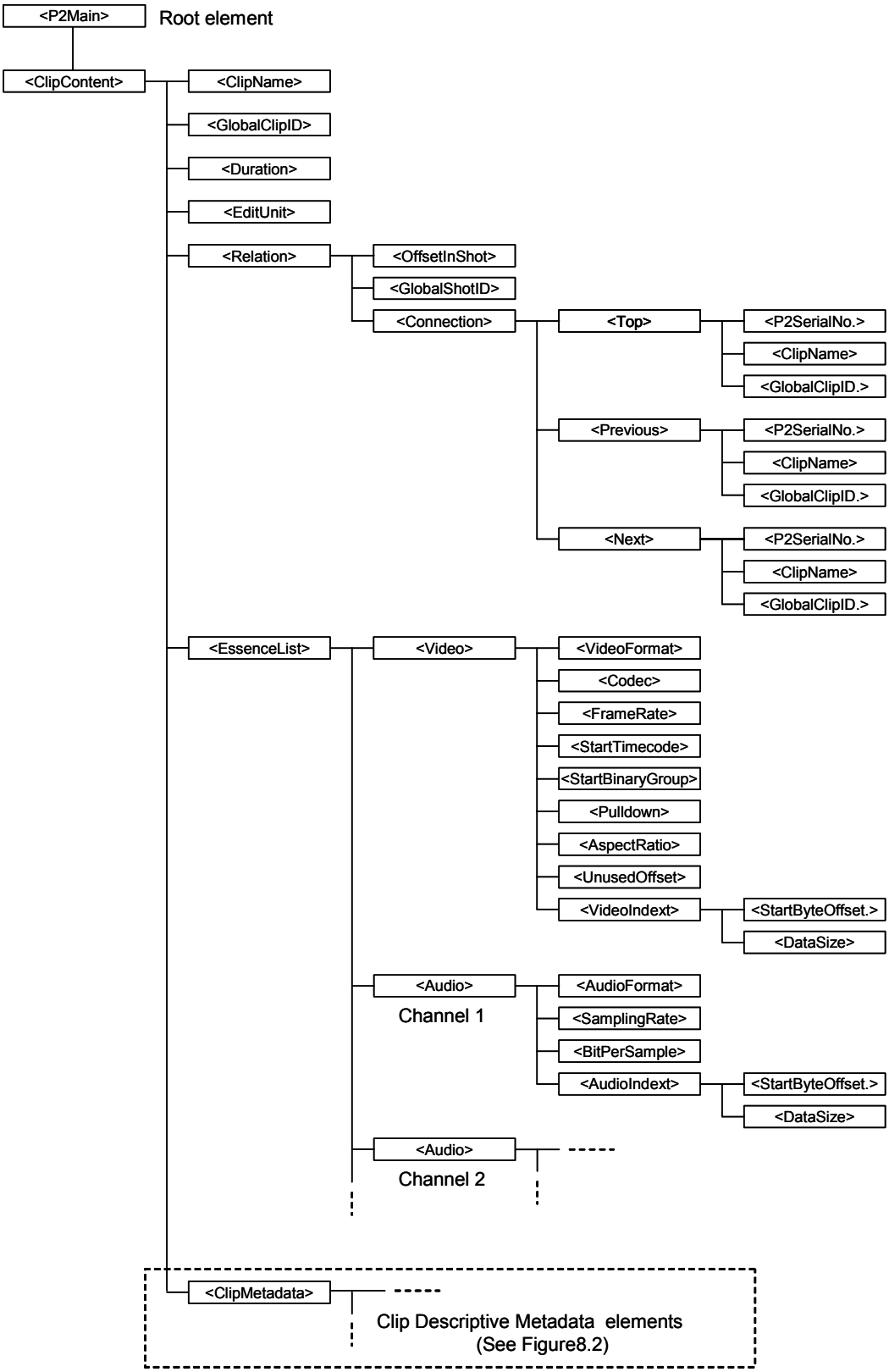


Figure 8.1 – Overview of Clip Structural Metadata elements

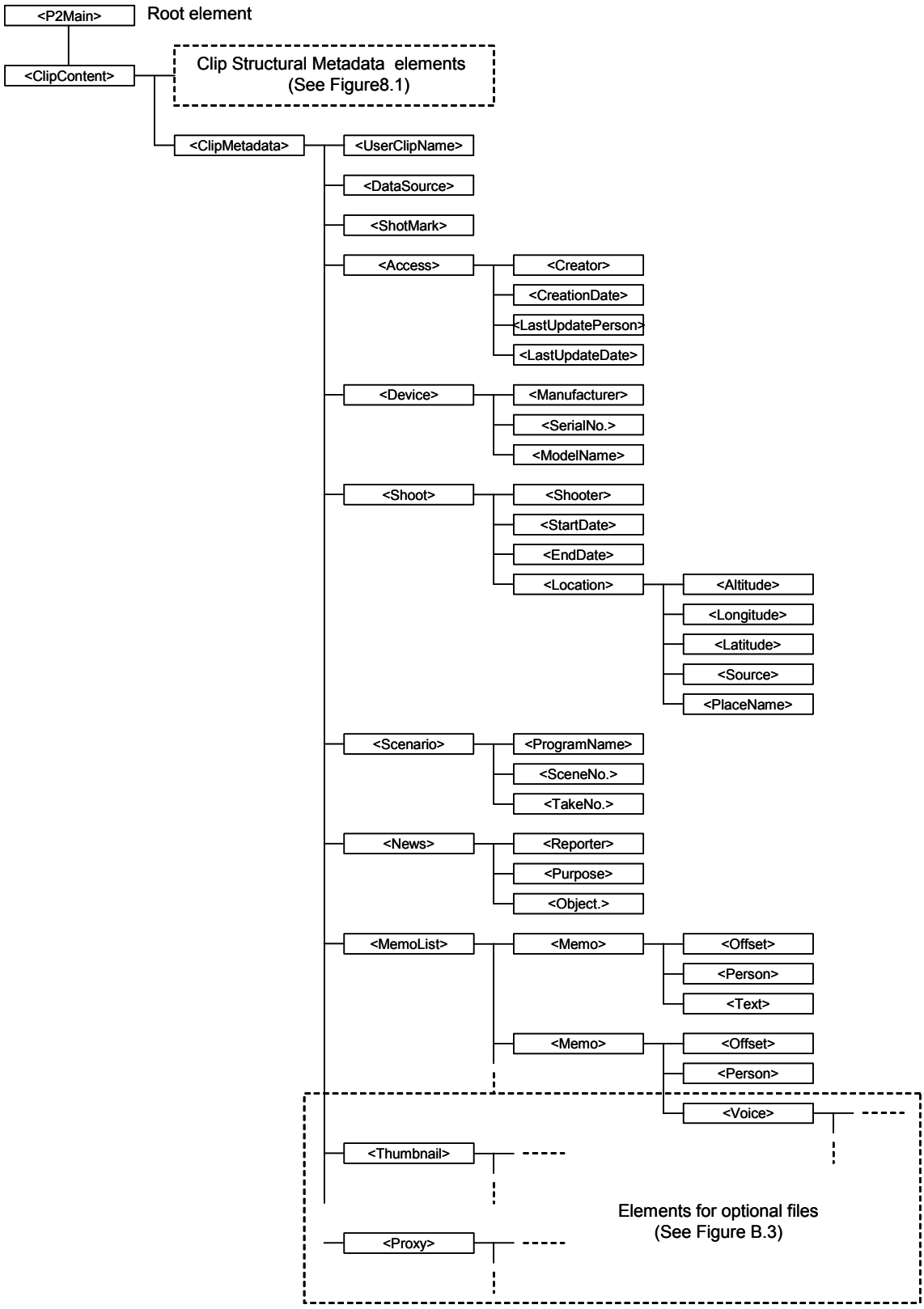


Figure 8.2 – Overview of Clip Descriptive Metadata elements

8.1 XML version and document encoding information

A valid Clip Metadata document specifies the XML version and the character encoding used in the document. The XML version shall be 1.0 and the character encoding shall be UTF-8. Names for character encoding shall be case-insensitive, so for example "UTF-8", "Utf-8", and "utf-8" expressions are equivalent.

8.2 Namespace

The namespace of the Clip Metadata description schemas specified in this document is defined as below.

urn:schemas-Professional-Plug-in:P2:ClipMetadata:v2.0

The Clip Metadata description schemas specified in this document include the description schemas that are defined in the XML Schema recommended by W3C.

8.3 Clip Structural metadata

8.3.1 P2Main element

The P2Main element shall be the root element of the Clip Metadata document.

- Child Elements

Name	Definition
ClipContent	Required. 1 sibling only.

- Schema

```
<xsd:element name="P2Main">
  <xsd:complexType>
    <xsd:choice>
      <xsd:element name="ClipContent" type="ClipContentType"/>
    </xsd:choice>
  </xsd:complexType>
</xsd:element>
```

8.3.2 ClipContent element

The ClipContent element describes a Clip.

- Child Elements

Name	Definition
ClipName	Required. 1 sibling only.
GlobalClipID	Required. 1 sibling only.
Duration	Required. 1 sibling only.
EditUnit	Required. 1 sibling only.
Relation	Optional. 0 or 1 sibling only.
EssenceList	Required. 1 sibling only.
ClipMetadata	Required. 1 sibling only.

- Schema

```
<xsd:complexType name="ClipContentType">
  <xsd:sequence>
    <xsd:element name="ClipName" type="ClipNameType"/>
    <xsd:element name="GlobalClipID" type="GlobalClipIDType"/>
    <xsd:element name="Duration" type="DurationType"/>
    <xsd:element name="EditUnit" type="EditUnitType"/>
    <xsd:element name="Relation" type="RelationType" minOccurs="0"/>
    <xsd:element name="EssenceList" type="EssenceListType"/>
    <xsd:element name="ClipMetadata" type="ClipMetadataType"/>
  </xsd:sequence>
</xsd:complexType>
```

8.3.3 ClipName element

The ClipName element specifies the name of this Clip, specified as "XXXXYY". The "XXXX" part shall be specified as a four digit decimal integer in the range “0000” to "9999”. The "YY" part shall be specified in a two-digit value generated from the Material Number of UMID for the GlobalClipID element defined in section 8.3.4.

The two-digit value shall be generated by the following process:

1. Separate material number of UMID into eight 16-bit (unsigned short) numbers. The 16-bit numbers are arranged as little endian. For example, when the first byte is 34h and the second 12h, the 16-bit number will be 1234h.
2. Calculate the exclusive OR (XOR) of every bit of each 16-bit number.
3. Calculate the remainder given by dividing the result of step 2 (XOR of every bits of each 16-bit number) by 1291 in decimal representation.
4. Express the remainder in base of 36. If the value is more than 9, it is expressed as A-Z or a-z.

For example, when the material number is ‘BEF8467C533005CC0040001004200024’, the two-digit value is generated as follows:

1. Material number ‘BEF8467C533005CC0040001004200024’ is divided into eight 16-bit numbers below. F8BEh, 7C46h, 3053h, CC05h, 4000h, 1000h, 2004h, 2400h
2. Calculate F8BEh XOR 7C46h XOR 3053h XOR CC05h XOR 4000h XOR 1000h XOR 2004h XOR 2400h, and the result is 2CAAh.
3. 2CAAh is expressed as 11434 in decimal. The remainder is found by modulo dividing 11434 by 1291 giving the result of 1106.
4. 1106 in decimal is expressed as ’30.26’ to base 36. ‘30’ is expressed as ‘U’ and ‘26’ is expressed as ‘Q’.

- Value

Value	Definition
XXXX	Integer from 0001 to 9999.
YY	Two digit value Each digit must have a value from 0 to 9 or from A to Z

- Schema

```
<xsd:simpleType name="ClipNameType">
  <xsd:restriction base="xsd:normalizedString">
    <xsd:pattern value="((000[1-9])|(00[1-9][0-9])|(0[1-9]([0-9]{2}))|([1-9]([0-
9]{3})))([0-9A-Z]{2})"/>
  </xsd:restriction>
</xsd:simpleType>
```

8.3.4 GlobalClipID element

The GlobalClipID element specifies the global unique ID (UMID) of this Clip. This element is specified as xsd:normalizedString.

Since the UMID contains byte values, the GlobalClipID shall be represented as a simple string of the hexadecimal representation of each byte of a UMID as follows: 060E2B3401 …… etc.

NOTE – GlobalClipID value depends on the implementation and may be a unique value or may be a replication of the PackageID defined in the MXF Header Metadata.

- Schema

```
<xsd:simpleType name="GlobalClipIDType">
  <xsd:restriction base="xsd:normalizedString"/>
</xsd:simpleType>
```

8.3.5 Duration element

The Duration element specifies the duration of this Clip in numbers of Edit Units (See Section 8.3.6). This element is specified as xsd:unsignedInt. This element value shall never be zero.

- Schema

```
<xsd:simpleType name="DurationType">
  <xsd:restriction base="xsd:unsignedInt">
    <xsd:minInclusive value="1"/>
  </xsd:restriction>
</xsd:simpleType>
```

8.3.6 EditUnit element

The EditUnit element specifies the Edit Unit of this Clip as a period of time. The values of the EditUnit element shall be expressed with a fractional number such as 1001/30000 for 1/29.97Hz.

- Value

Value	Definition
1001/30000	Video or Audio & Video mixed Clip of 59.94i system
1/25	Video or Audio & Video mixed Clip of 50i system
1/48000	48K samples/sec Audio Clip.

NOTE – In SMPTE 377M, the Edit Rate property is defined as a rational number that specifies the number of Editable Units per second while the EditUnit element in the Clip Metadata is defined as a period of time equal to 1/(Edit Rate). Therefore, the relationship of the values defined in the Clip Structural Metadata and the MXF Structural Metadata are given below.

Clip Metadata value	Edit Rate in MXF Metadata
1001/30000	{ 30000, 1001}
1/25	{25,1}
1/48000	{48000, 1}

- Schema

```
<xsd:simpleType name="EditUnitType">
  <xsd:restriction base="xsd:normalizedString">
    <xsd:enumeration value="1001/30000"/>
    <xsd:enumeration value="1/25"/>
    <xsd:enumeration value="1/48000"/>
  </xsd:restriction>
</xsd:simpleType>
```

8.3.7 Relation element

The Relation element describes the relations between Clips which are stored in multiple SSM Cards. This element is optional but shall be specified when a Shot, which is a Content unit for sequential recording or playback, is structured with multiple Clips and the Clips are stored in the multiple SSM Cards. When a Shot is structured with one Clip, this element is omitted.

- Child Elements

Name	Definition
OffsetInShot	Required. 1 sibling only.
GlobalShotID	Required. 1 sibling only.
Connection	Required. 1 sibling only.

- Schema

```
<xsd:complexType name="RelationType">
  <xsd:sequence>
    <xsd:element name="OffsetInShot" type="OffsetInShotType"/>
    <xsd:element name="GlobalShotID" type="GlobalShotIDType"/>
    <xsd:element name="Connection" type="ConnectionType"/>
  </xsd:sequence>
</xsd:complexType>
```

8.3.8 OffsetInShot element

The OffsetInShot element specifies the in-point of this Clip from the start of the Shot in numbers of Edit Units (See Section 8.3.6). This element is specified as xsd:unsignedInt.

- Schema

```
<xsd:simpleType name="OffsetInShotType">
  <xsd:restriction base="xsd:unsignedInt"/>
</xsd:simpleType>
```

8.3.9 GlobalShotID element

The GlobalShotID specifies the global unique ID (UMID) of the related Shot. This element is specified as xsd:normalizedString.

Since the UMID contains byte values, the GlobalShotID shall be represented as a simple string of the hexadecimal representation of each byte of a UMID as follows: 060E2B3401 …… etc.

NOTE – GlobalShotID value depends on the implementation and may be a unique value or may be a replication of the Package UID defined in the MXF Header Metadata.

- Schema

```
<xsd:simpleType name="GlobalShotIDType">
  <xsd:restriction base="xsd:normalizedString"/>
</xsd:simpleType>
```

8.3.10 Connection element

The Connection element describes a list of related SSM Cards.

- Child Elements

Name	Definition
Top	Required. 1 sibling only.
Previous	Optional. 0 or 1 sibling only.
Next	Optional. 0 or 1 sibling only.

- Schema

```
<xsd:complexType name="ConnectionType">
  <xsd:sequence>
    <xsd:element name="Top" type="TopType"/>
    <xsd:element name="Previous" type="PreviousType" minOccurs="0"/>
    <xsd:element name="Next" type="NextType" minOccurs="0"/>
  </xsd:sequence>
</xsd:complexType>
```

8.3.11 Top element

The Top element describes the information about the clip which is the start of a shot and is located on a different SSM Card.

- Child Elements

Name	Definition
ClipName	Required. 1 sibling only.
GlobalClipID	Required. 1 sibling only.
P2SerialNo.	Required. 1 sibling only.

- Schema

```
<xsd:complexType name="TopType">
  <xsd:sequence>
    <xsd:element name="ClipName" type="ClipNameType"/>
    <xsd:element name="GlobalClipID" type="GlobalClipIDType"/>
    <xsd:element name="P2SerialNo." type="P2SerialNo.Type"/>
  </xsd:sequence>
</xsd:complexType>
```

8.3.12 P2SerialNo. element

The P2SerialNo.element specifies the serial number of the related SSM card. This element is specified as xsd:token. The maximum length shall be 20 characters.

- Schema

```
<xsd:simpleType name="P2SerialNo.Type">
  <xsd:restriction base="xsd:token">
    <xsd:maxLength value="20"/>
  </xsd:restriction>
</xsd:simpleType>
```

8.3.13 Previous element

The Previous element describes the information about the clip which precedes this clip and is located on a different SSM Card. This element is optional and is only used if this clip is at least the third clip in a shot which spans multiple SSM Cards.

- Child Elements

Name	Definition
ClipName	Required. 1 sibling only.
GlobalClipID	Required. 1 sibling only.
P2SerialNo.	Required. 1 sibling only.

- Schema

```
<xsd:complexType name="PreviousType">
  <xsd:sequence>
    <xsd:element name="ClipName" type="ClipNameType"/>
    <xsd:element name="GlobalClipID" type="GlobalClipIDType"/>
    <xsd:element name="P2SerialNo." type="P2SerialNo.Type"/>
  </xsd:sequence>
</xsd:complexType>
```

8.3.14 Next element

The Next element describes the information about the clip which follows this clip and is located on a different SSM Card. This element is optional.

- Child Elements

Name	Definition
ClipName	Required. 1 sibling only.
GlobalClipID	Required. 1 sibling only.
P2SerialNo.	Required. 1 sibling only.

- Schema

```
<xsd:complexType name="NextType">
  <xsd:sequence>
    <xsd:element name="ClipName" type="ClipNameType"/>
    <xsd:element name="GlobalClipID" type="GlobalClipIDType"/>
    <xsd:element name="P2SerialNo." type="P2SerialNo.Type"/>
  </xsd:sequence>
</xsd:complexType>
```

8.3.15 EssenceList element

The EssenceList element describes a list of Essences.

- Child Elements

Name	Definition
Video	Optional. 0 or 1 sibling only. Either one Video or Audio must exist.
Audio	Optional. 0 to 16 siblings. Either one Video or Audio must exist.

- Schema

```
<xsd:complexType name="EssenceListType">
  <xsd:choice>
    <xsd:sequence>
      <xsd:element name="Video" type="VideoType"/>
      <xsd:element name="Audio" type="AudioType" minOccurs="0" maxOccurs="16"/>
    </xsd:sequence>
    <xsd:element name="Audio" type="AudioType" maxOccurs="16"/>
  </xsd:choice>
</xsd:complexType>
```

8.3.16 Video element

The Video element describes a Video Essence. This element is optional, but shall be written where a video essence file exists.

- Child Elements

Name	Definition
VideoFormat	Required. 1 sibling only.
Codec	Required. 1 sibling only.
FrameRate	Required. 1 sibling only.
StartTimecode	Required. 1 sibling only.
StartBinaryGroup	Optional. 0 or 1 sibling only.
Pulldown	Optional. 0 or 1 sibling only.
AspectRatio	Optional. 0 or 1 sibling only.
UnusedOffset	Optional. 0 or 1 sibling only.
VideoIndex	Required. 1 sibling only.

- Schema

```
<xsd:complexType name="VideoType">
  <xsd:choice>
    <xsd:sequence>
      <xsd:element name="VideoFormat" type="VideoFormatType"/>
      <xsd:element name="Codec" type="CodecType"/>
      <xsd:element name="FrameRate" type="FrameRateType"/>
      <xsd:element name="StartTimecode" type="StartTimecodeType"/>
      <xsd:element name="StartBinaryGroup" type="StartBinaryGroupType"
minOccurs="0"/>
      <xsd:element name="Pulldown" type="PulldownType" minOccurs="0"/>
      <xsd:element name="AspectRatio" type="AspectRatioType" minOccurs="0"/>
      <xsd:element name="UnusedOffset" type="UnusedOffsetType" minOccurs="0"/>
      <xsd:element name="VideoIndex" type="VideoIndexType"/>
    </xsd:sequence>
    <xsd:sequence>
      <xsd:element name="FrameRate" type="FrameRateType"/>
      <xsd:element name="StartTimecode" type="StartTimecodeType"/>
      <xsd:element name="StartBinaryGroup" type="StartBinaryGroupType"
minOccurs="0"/>
    </xsd:sequence>
  </xsd:choice>
</xsd:complexType>
```

8.3.17 VideoFormat element

The VideoFormat element specifies the file format of this Video Essence.

- Value

Value	Definition
MXF	Material Exchange Format.

- Schema

```
<xsd:simpleType name="VideoFormatType">
  <xsd:restriction base="xsd:NMTOKEN">
    <xsd:enumeration value="MXF"/>
  </xsd:restriction>
</xsd:simpleType>
```

8.3.18 Codec element

The Codec element specifies the codec type of this Video Essence.

- Value

Value	Definition
DV25_411	IEC DV and DV-based compression, 25Mbps, YCbCr at 4:1:1.
DV25_420	IEC DV compression, 25Mbps, YCbCr at 4:2:0.
DV50_422	DV-based compression, 50Mbps, YCbCr at 4:2:2.

NOTE – In SMPTE 377M, the Content type is defined in the Essence Container label shown in Table 6.2 and the Picture Essence Coding property shown in Table 6.5 to specify the Picture Compression scheme. The relationship between values in the Clip Structural Metadata and Content types in the MXF Structural Metadata are given below. In the case of DV25_411, the Content type in the MXF Structural Metadata should be used to distinguish between the IEC DV and the DV-based Essence.

Clip Metadata value	Content type in MXF Metadata
DV25_411	IEC DV compressed at 25Mbps, 525/59.94-I source DV-based compressed at 25Mbps, 525/59.94-I source DV-based compressed at 25Mbps, 625/50-I source
DV25_420	IEC DV compressed at 25Mbps, 625/50-I source
DV50_422	DV-based compressed at 50Mbps, 525/59.94-I source DV-based compressed at 50Mbps, 625/50-I source

- Schema

```
<xsd:simpleType name="CodecType">
  <xsd:restriction base="xsd:NMTOKEN">
    <xsd:enumeration value="DV25_411"/>
    <xsd:enumeration value="DV25_420"/>
    <xsd:enumeration value="DV50_422"/>
  </xsd:restriction>
</xsd:simpleType>
```

8.3.19 FrameRate element

The FrameRate element specifies the frame rate while playing a clip.

- Value

Value	Definition
50i	25 fps interlaced.
59.94i	29.97 fps interlaced.

NOTE – In SMPTE 377M, the Sample Rate property is defined as a rational number that specifies the field or frame rate of an Essence Container while the FrameRate element in the Clip Metadata is expressed with character strings. Therefore, the relationship of the values defined in the Clip Structural Metadata and the MXF Structural Metadata are given below.

Clip Metadata value	Sample Rate in MXF Metadata
50i	{ 25,1 }
59.94i	{ 30000, 1001}

- Attributes

Name	Definition
DropFrameFlag	A boolean attribute that specifies whether the Timecode for this clip is recorded as drop-frame or non drop-frame as needed. Specified as xsd:boolean. Optional. Only specified when the FrameRate is "59.94i".

- Schema

```
<xsd:complexType name="FrameRateType">
  <xsd:simpleContent>
    <xsd:extension base="FrameRateBaseType">
      <xsd:attribute name="DropFrameFlag" type="xsd:boolean" use="optional"/>
    </xsd:extension>
  </xsd:simpleContent>
</xsd:complexType>
<xsd:simpleType name="FrameRateBaseType">
  <xsd:restriction base="xsd:NMTOKEN">
    <xsd:enumeration value="50i"/>
    <xsd:enumeration value="59.94i"/>
  </xsd:restriction>
</xsd:simpleType>
```

8.3.20 StartTimecode element

The StartTimecode element specifies the timecode of the in-point of this Clip. This element is specified as "HH:MM:SS:FF".

- Value

Value	Definition
HH	Hour. From 00 to 23.
MM	Minute. From 00 to 59.
SS	Second. From 00 to 59.
FF	Frame number. From 00 to 29.

- Schema

```
<xsd:simpleType name="StartTimecodeType">
  <xsd:restriction base="xsd:string">
    <xsd:pattern value="([0-1][0-9]|2[0-3]):[0-5][0-9]:[0-5][0-9]:[0-2][0-9]"/>
  </xsd:restriction>
</xsd:simpleType>
```

8.3.21 StartBinaryGroup element

The StartBinaryGroup element specifies the Binary Group (BG) data of the in-point of this Clip. (Detailed information is given in SMPTE12M.) This element is specified as 8 hexadecimal digits. This element is optional.

- Schema

```
<xsd:simpleType name="StartBinaryGroupType">
  <xsd:restriction base="xsd:string">
    <xsd:pattern value="[0-9A-F]{8}"/>
  </xsd:restriction>
</xsd:simpleType>
```

8.3.22 Pulldown element

The Pulldown element specifies the pull-down type. This element is optional.

- Value

Value	Definition
2:2	2:2 pull-down
2:3	2:3 pull-down
2:3:3:2	2:3:3:2 pull-down

- Schema

```
<xsd:simpleType name="PulldownType">
  <xsd:restriction base="xsd:NMTOKEN">
    <xsd:enumeration value="2:2"/>
    <xsd:enumeration value="2:3"/>
    <xsd:enumeration value="2:3:3:2"/>
  </xsd:restriction>
</xsd:simpleType>
```

8.3.23 AspectRatio element

The AspectRatio element specifies the aspect ratio that conveys the picture with no geometric distortion (this may include black bars to give letterbox or pillarbox pictures). This element is optional.

- Value

Value	Definition
4:3	Width:Height=4:3.
16:9	Width:Height=16:9.

- Schema

```
<xsd:simpleType name="AspectRatioType">
  <xsd:restriction base="xsd:NMTOKEN">
    <xsd:enumeration value="4:3"/>
    <xsd:enumeration value="16:9"/>
  </xsd:restriction>
</xsd:simpleType>
```

8.3.24 UnusedOffset element

The UnusedOffset element specifies the offset of unused data. The offset is described as the number of frames of unused duration from the start of the Essence element value. This element is specified as xsd:unsignedInt except in the case of zero, where this element shall not be used. This element is optional.

- Schema

```
<xsd:simpleType name="UnusedOffsetType">
  <xsd:restriction base="xsd:unsignedInt">
    <xsd:minInclusive value="1"/>
  </xsd:restriction>
</xsd:simpleType>
```

8.3.25 VideoIndex element

The VideoIndex element specifies the index as a byte offset to the start of the Video Essence element value and its duration as a byte count.

- Child Elements

Name	Definition
StartByteOffset	Required. 1 sibling only.
DataSize	Required. 1 sibling only.

- Schema

```
<xsd:complexType name="VideoIndexType">
  <xsd:sequence>
    <xsd:element name="StartByteOffset" type="StartByteOffsetType"/>
    <xsd:element name="DataSize" type="DataSizeType"/>
  </xsd:sequence>
</xsd:complexType>
```

8.3.26 StartByteOffset element

The StartByteOffset element specifies the displacement, the number of bytes, from the first byte of the essence file to the first byte of the essence element value. This element is specified as xsd:unsignedInt. This element value shall never be zero.

- Schema

```
<xsd:simpleType name="StartByteOffsetType">
  <xsd:restriction base="xsd:unsignedInt">
    <xsd:minInclusive value="1"/>
  </xsd:restriction>
</xsd:simpleType>
```

8.3.27 DataSize element

The DataSize element specifies the size, in number of bytes, of the essence element value. This element is specified as xsd:unsignedLong. This element value shall never be zero.

- Schema

```
<xsd:simpleType name="DataSizeType">
  <xsd:restriction base="xsd:unsignedLong">
    <xsd:minInclusive value="1"/>
  </xsd:restriction>
</xsd:simpleType>
```

8.3.28 Audio element

The Audio element describes an Audio Essence, which may include reserved audio channels without audio data. If a Clip Content contains multiple Audio Essences, the Audio elements are treated as their channel numbers are 0, 1, 2, and so on, in the order of their appearance in the XML document. For example, the channel number of the first Audio element will be "0", the second "1", the third "2", and so on. When this element stands for a reserved audio channel, no child elements should be specified. This element is optional.

- Child Elements

Name	Definition
AudioFormat	Required. 1 sibling only.
SamplingRate	Required. 1 sibling only.
BitsPerSample	Required. 1 sibling only.
AudioIndex	Required. 1 sibling only.

- Schema

```
<xsd:complexType name="AudioType">
  <xsd:sequence minOccurs="0">
    <xsd:element name="AudioFormat" type="AudioFormatType"/>
    <xsd:element name="SamplingRate" type="SamplingRateType"/>
    <xsd:element name="BitsPerSample" type="BitsPerSampleType"/>
    <xsd:element name="AudioIndex" type="AudioIndexType"/>
  </xsd:sequence>
</xsd:complexType>
```

8.3.29 AudioFormat element

The AudioFormat element specifies the file format of this Audio Essence.

- Value

Value	Definition
MXF	Material Exchange Format.

- Schema

```
<xsd:simpleType name="AudioFormatType">
  <xsd:restriction base="xsd:NMTOKEN">
    <xsd:enumeration value="MXF"/>
  </xsd:restriction>
</xsd:simpleType>
```

8.3.30 SamplingRate element

The SamplingRate element specifies the sampling rate of this Audio Essence.

- Value

Value	Definition
48000	48000 samples/sec.

NOTE – In SMPTE 377M, the Sample Rate or the Audio sampling Rate properties are defined as a rational number that specifies the sampling rate of the audio essence while the SamplingRate element in the Clip Metadata is expressed with an integer number. Therefore, the relationship of the values defined in the Clip Structural Metadata and the MXF Structural Metadata are given below.

Clip Metadata value	Sample Rate or Audio sampling Rate in MXF Metadata
48000	{ 48000, 1}

- Schema

```
<xsd:simpleType name="SamplingRateType">
  <xsd:restriction base="xsd:positiveInteger">
    <xsd:enumeration value="48000"/>
  </xsd:restriction>
</xsd:simpleType>
```

8.3.31 BitsPerSample element

The BitsPerSample element specifies the number of bits per sample for this Audio Essence.

- Value

Value	Definition
16	16 bits/sample
24	24 bits/sample

- Schema

```
<xsd:simpleType name="BitsPerSampleType">
  <xsd:restriction base="xsd:positiveInteger">
    <xsd:enumeration value="16"/>
    <xsd:enumeration value="24"/>
  </xsd:restriction>
</xsd:simpleType>
```

8.3.32 AudioIndex element

The AudioIndex element specifies the index as a byte offset to the start of the Audio Essence element value and its duration as a byte count.

- Child Elements

Name	Definition
StartByteOffset	Required. 1 sibling only.
DataSize	Required. 1 sibling only.

- Schema

```
<xsd:complexType name="AudioIndexType">
  <xsd:sequence>
    <xsd:element name="StartByteOffset" type="StartByteOffsetType"/>
    <xsd:element name="DataSize" type="DataSizeType"/>
  </xsd:sequence>
</xsd:complexType>
```

8.4 Clip Descriptive metadata

8.4.1 ClipMetadata element

The ClipMetadata element contains additional information describing this Clip.

- Child Elements

Name	Definition
UserClipName	Optional. 0 or 1 sibling only.
DataSource	Required. 1 sibling only.
ShotMark	Optional. 0 or 1 sibling only.
Access	Required. 1 sibling only.
Device	Optional. 0 or 1 sibling only.
Shoot	Optional. 0 or 1 sibling only. Required when DataSource is "SHOOTING".
Scenario	Optional. 0 or 1 sibling only.
News	Optional. 0 or 1 sibling only.
MemoList	Optional. 0 or 1 sibling only.
Thumbnail	Optional. 0 or 1 sibling only (1).
Proxy	Optional. 0 or 1 sibling only (2).

NOTES

1 The Thumbnail element is optional, but shall be written where the Thumbnail file exists. The Thumbnail element and its child elements are described in Annex B.

2 The Proxy element is optional, but shall be written where the Proxy AV file exists. The Proxy element and its child elements are described in Annex B.

- Schema

```
<xsd:complexType name="ClipMetadataType">
  <xsd:sequence>
    <xsd:element name="UserClipName" type="UserClipNameType" minOccurs="0"/>
    <xsd:element name="DataSource" type="DataSourceType"/>
    <xsd:element name="ShotMark" type="ShotMarkType" minOccurs="0"/>
    <xsd:element name="Access" type="AccessType"/>
    <xsd:element name="Device" type="DeviceType" minOccurs="0"/>
    <xsd:element name="Shoot" type="ShootType" minOccurs="0"/>
    <xsd:element name="Scenario" type="ScenarioType" minOccurs="0"/>
    <xsd:element name="News" type="NewsType" minOccurs="0"/>
    <xsd:element name="MemoList" type="MemoListType" minOccurs="0">
      <xsd:unique name="MemoIDunderMemoList">
        <xsd:selector xpath="Memo"/>
        <xsd:field xpath="@MemoID"/>
      </xsd:unique>
    </xsd:element>
    <xsd:element name="Thumbnail" type="ThumbnailType" minOccurs="0"/>
    <xsd:element name="Proxy" type="ProxyType" minOccurs="0"/>
  </xsd:sequence>
</xsd:complexType>
```

8.4.2 UserClipName element

The UserClipName element specifies the user-defined name of this Clip. This element is specified as xsd:normalizedString. The maximum length of this element shall be 100 characters. This element is optional.

- Schema

```
<xsd:simpleType name="UserClipNameType">
  <xsd:restriction base="xsd:normalizedString">
    <xsd:maxLength value="100"/>
  </xsd:restriction>
</xsd:simpleType>
```

8.4.3 DataSource element

The DataSource element specifies how this clip was generated.

- Value

Value	Definition
SHOOTING	By shooting
RENDERING	By rendering

- Schema

```
<xsd:simpleType name="DataSourceType">
  <xsd:restriction base="xsd:NMTOKEN">
    <xsd:enumeration value="SHOOTING"/>
    <xsd:enumeration value="RENDERING"/>
  </xsd:restriction>
</xsd:simpleType>
```

8.4.4 ShotMark element

The ShotMark element specifies the quality of this Clip. This element is specified as xsd:boolean. The meaning of the boolean value is user defined. This element is optional.

- Schema

```
<xsd:simpleType name="ShotMarkType">
  <xsd:restriction base="xsd:boolean"/>
</xsd:simpleType>
```

8.4.5 Access element

The Access element describes the person and the date for the Clip creation as well as the last Clip update.

- Child Elements

Name	Definition
Creator	Optional. 0 or 1 sibling only.
CreationDate	Required. 1 sibling only.
LastUpdatePerson	Optional. 0 or 1 sibling only.
LastUpdateDate	Required. 1 sibling only.

- Schema

```
<xsd:complexType name="AccessType">
  <xsd:sequence>
    <xsd:element name="Creator" type="CreatorType" minOccurs="0"/>
    <xsd:element name="CreationDate" type="CreationDateType"/>
    <xsd:element name="LastUpdatePerson" type="LastUpdatePersonType"
minOccurs="0"/>
    <xsd:element name="LastUpdateDate" type="LastUpdateDateType"/>
  </xsd:sequence>
</xsd:complexType>
```

8.4.6 Creator element

The Creator element specifies the name of the person who created this Clip. This element is specified as xsd:normalizedString. The maximum length shall be 30 characters. This element is optional.

- Schema

```
<xsd:simpleType name="CreatorType">
  <xsd:restriction base="xsd:normalizedString">
    <xsd:maxLength value="30"/>
  </xsd:restriction>
</xsd:simpleType>
```

8.4.7 CreationDate element

The CreationDate element specifies the date and time of creation. This element is specified as xsd:dateTime.

- Schema

```
<xsd:simpleType name="CreationDateType">
  <xsd:restriction base="xsd:dateTime">
<xsd:pattern value="[0-9]{4}\-(0[0-9]|1[0-2])\-([0-2][0-9]|3[0-1])T([0-1][0-9]|2[0-
3]):[0-5][0-9]:[0-5][0-9](((\-(0[0-9]|1[0-1]):[0-5][0-9])|\-12:00)|((\+(0[0-9]|1[0-
3]):[0-5][0-9]))|\+14:00)"/>
  </xsd:restriction>
</xsd:simpleType>
```

8.4.8 LastUpdatePerson element

The LastUpdatePerson element specifies the name of the person who updated this clip last. This element is specified as xsd:normalizedString. The maximum length shall be 30 characters. This element is optional.

- Schema

```
<xsd:simpleType name="LastUpdatePersonType">
  <xsd:restriction base="xsd:normalizedString">
    <xsd:maxLength value="30"/>
  </xsd:restriction>
</xsd:simpleType>
```

8.4.9 LastUpdateDate element

The LastUpdateDate element specifies the date and time when this clip was last updated. This element is specified as xsd:dateTime.

- Schema

```
<xsd:simpleType name="LastUpdateDateType">
  <xsd:restriction base="xsd:dateTime">
    <xsd:pattern value="[0-9]{4}\-(0[0-9]|1[0-2])\-([0-2][0-9]|3[0-1])T([0-1][0-
9]|2[0-3]):[0-5][0-9]:[0-5][0-9](((\-(0[0-9]|1[0-1]):[0-5][0-9])|\-12:00)|((\+(0[0-
9]|1[0-3]):[0-5][0-9]))|\+14:00)"/>
  </xsd:restriction>
</xsd:simpleType>
```

8.4.10 Device element

The Device element describes the information of the device creating this Clip. If the Device element is used, at least one of its three Child Elements shall exist. This element is optional.

- Child Elements

Name	Definition
Manufacturer	Optional. 0 or 1 sibling only.
SerialNo.	Optional. 0 or 1 sibling only.
ModelName	Optional. 0 or 1 sibling only.

- Schema

```
<xsd:complexType name="DeviceType">
  <xsd:choice>
    <xsd:sequence>
      <xsd:element name="Manufacturer" type="ManufacturerType"/>
      <xsd:element name="SerialNo." type="SerialNo.Type" minOccurs="0"/>
      <xsd:element name="ModelName" type="ModelNameType" minOccurs="0"/>
    </xsd:sequence>
    <xsd:sequence>
      <xsd:element name="SerialNo." type="SerialNo.Type"/>
      <xsd:element name="ModelName" type="ModelNameType" minOccurs="0"/>
    </xsd:sequence>
    <xsd:sequence>
      <xsd:element name="ModelName" type="ModelNameType"/>
    </xsd:sequence>
  </xsd:choice>
</xsd:complexType>
```

8.4.11 Manufacturer element

The Manufacturer element specifies the manufacturer of the device creating this Clip. This element is specified as xsd:normalizedString. The maximum length shall be 30 characters. This element is optional.

- Schema

```
<xsd:simpleType name="ManufacturerType">
  <xsd:restriction base="xsd:normalizedString">
    <xsd:maxLength value="30"/>
  </xsd:restriction>
</xsd:simpleType>
```

8.4.12 SerialNo. element

The SerialNo. element specifies the Serial number of the device creating this Clip. This element is specified as xsd:token. The maximum length shall be 20 characters. This element is optional.

- Schema

```
<xsd:simpleType name="SerialNo.Type">
  <xsd:restriction base="xsd:token">
    <xsd:maxLength value="20"/>
  </xsd:restriction>
</xsd:simpleType>
```

8.4.13 ModelName element

The ModelName element specifies the model name of the device creating this Clip. This element is specified as xsd:normalizedString. The maximum length shall be 30 characters. This element is optional.

- Schema

```
<xsd:simpleType name="ModelNameType">
  <xsd:restriction base="xsd:normalizedString">
    <xsd:maxLength value="30"/>
  </xsd:restriction>
</xsd:simpleType>
```

8.4.14 Shoot element

The Shoot element describes information about a shoot. This element is optional.

- Child Elements

Name	Definition
Shooter	Optional. 0 or 1 sibling only.
StartDate	Required. 1 sibling only.
EndDate	Required. 1 sibling only.
Location	Optional. 0 or 1 sibling only.

- Schema

```
<xsd:complexType name="ShootType">
  <xsd:sequence>
    <xsd:element name="Shooter" type="ShooterType" minOccurs="0"/>
    <xsd:element name="StartDate" type="StartDateType"/>
    <xsd:element name="EndDate" type="EndDateType"/>
    <xsd:element name="Location" type="LocationType" minOccurs="0"/>
  </xsd:sequence>
</xsd:complexType>
```

8.4.15 Shooter element

The Shooter element specifies the name of the person who created this Clip. This element is specified as xsd:normalizedString. The maximum length shall be 30 characters. This element is optional.

- Schema

```
<xsd:simpleType name="ShooterType">
  <xsd:restriction base="xsd:normalizedString">
    <xsd:maxLength value="30"/>
  </xsd:restriction>
</xsd:simpleType>
```

8.4.16 StartDate element

The StartDate element specifies the date and time of the start of the Clip creation. This element is specified as xsd:dateTime.

- Schema

```
<xsd:simpleType name="StartDateType">
  <xsd:restriction base="xsd:dateTime">
    <xsd:pattern value="[0-9]{4}\-(0[0-9]|1[0-2])\-([0-2][0-9]|3[0-1])T([0-1][0-
9]|2[0-3]):[0-5][0-9]:[0-5][0-9](((\-(0[0-9]|1[0-1]):[0-5][0-9])|\-12:00)|((\+(0[0-
9]|1[0-3]):[0-5][0-9]))|\+14:00)"/>
  </xsd:restriction>
</xsd:simpleType>
```

8.4.17 EndDate element

The EndDate element specifies the date and time of the end of the Clip creation. This element is specified as xsd:dateTime.

- Schema

```
<xsd:simpleType name="EndDateType">
  <xsd:restriction base="xsd:dateTime">
    <xsd:pattern value="[0-9]{4}\-(0[0-9]|1[0-2])\-([0-2][0-9]|3[0-1])T([0-1][0-
9]|2[0-3]):[0-5][0-9]:[0-5][0-9](((\-(0[0-9]|1[0-1]):[0-5][0-9])|\-12:00)|((\+(0[0-
9]|1[0-3]):[0-5][0-9]))|\+14:00)"/>
  </xsd:restriction>
</xsd:simpleType>
```

8.4.18 Location element

The Location element specifies the location of the Clip creation. This element is optional. The reference World Geodetic System for the spatial co-ordinate elements shall be the WGS84.

- Child Elements

Name	Definition
Altitude	Optional. 0 or 1 sibling only.
Longitude	Required when PlaceName doesn't exist. 1 sibling only.
Latitude	Required when PlaceName doesn't exist. 1 sibling only.
Source	Required when PlaceName doesn't exist. 1 sibling only.
PlaceName	Optional. 0 or 1 sibling only.

- Schema

```
<xsd:complexType name="LocationType">
  <xsd:choice>
    <xsd:sequence>
      <xsd:element name="Altitude" type="AltitudeType" minOccurs="0"/>
      <xsd:element name="Longitude" type="LongitudeType"/>
      <xsd:element name="Latitude" type="LatitudeType"/>
      <xsd:element name="Source" type="SourceType"/>
      <xsd:element name="PlaceName" type="PlaceNameType" minOccurs="0"/>
    </xsd:sequence>
    <xsd:element name="PlaceName" type="PlaceNameType"/>
  </xsd:choice>
</xsd:complexType>
```

8.4.19 Altitude element

The Altitude element specifies the altitude of the location of the Clip creation. The altitude is expressed as a 1-to-6-digit signed value relative to the sea level of the local geoid. This element is optional.

NOTE – The accuracy of the value may be subject to errors depending on the method by which the GPS coordinate was created.

- Schema

```
<xsd:simpleType name="AltitudeType">
  <xsd:restriction base="xsd:normalizedString">
    <xsd:pattern value="(\+|\-)([0-9]|[1-9][0-9]{1,5})"/>
  </xsd:restriction>
</xsd:simpleType>
```

8.4.20 Longitude element

The Longitude element specifies the longitude of the location of the Clip creation. The Longitude element is expressed as an xsd:normalizedString. For example, “135 degrees east longitude” is expressed as "E135.00000".

NOTE – The accuracy of the value may be subject to errors depending on the method by which the GPS coordinate was created.

- Schema

```
<xsd:simpleType name="LongitudeType">
  <xsd:restriction base="xsd:normalizedString">
    <xsd:pattern value="(W|E)([0-9]|[1-9][0-9]|1[0-7][0-9]|180).[0-9]{5}"/>
  </xsd:restriction>
</xsd:simpleType>
```

8.4.21 Latitude element

The Latitude element specifies the latitude of the location of the Clip creation. The Latitude element is expressed as an xsd:normalizedString. For example, “35 degrees north latitude” is expressed as "N35.00000".

NOTE – The accuracy of the value may be subject to errors depending on the method by which the GPS coordinate was created.

- Schema

```
<xsd:simpleType name="LatitudeType">
  <xsd:restriction base="xsd:normalizedString">
    <xsd:pattern value="(N|S)([0-9]|[1-8][0-9]|90).[0-9]{5}"/>
  </xsd:restriction>
</xsd:simpleType>
```

8.4.22 Source element

The Source element specifies the source of the Clip creation location information.

- Value

Value	Definition
CAPTURED	Received from GPS system on capture device.
CREATED	Manually generated after shooting.
DIVERSED	GPS data generated separately from capture device.

- Schema

```
<xsd:simpleType name="SourceType">
  <xsd:restriction base="xsd:NMTOKEN">
    <xsd:enumeration value="CAPTURED"/>
    <xsd:enumeration value="CREATED"/>
    <xsd:enumeration value="DIVERSED"/>
  </xsd:restriction>
</xsd:simpleType>
```

8.4.23 PlaceName element

The PlaceName element specifies the user-defined name of the location of the Clip creation. This element is specified as xsd:string. The maximum length shall be 100 characters. This element is optional.

- Schema

```
<xsd:simpleType name="PlaceNameType">
  <xsd:restriction base="xsd:string">
    <xsd:maxLength value="100"/>
  </xsd:restriction>
</xsd:simpleType>
```

8.4.24 Scenario element

The Scenario element describes the information of scenario. This element is optional.

- Child Elements

Name	Definition
ProgramName	Required. 1 sibling only.
SceneNo.	Optional. 0 or 1 sibling only.
TakeNo.	Optional. 0 or 1 sibling only.

- Schema

```
<xsd:complexType name="ScenarioType">
  <xsd:sequence>
    <xsd:element name="ProgramName" type="ProgramNameType"/>
    <xsd:element name="SceneNo." type="SceneNo.Type" minOccurs="0"/>
    <xsd:element name="TakeNo." type="TakeNo.Type" minOccurs="0"/>
  </xsd:sequence>
</xsd:complexType>
```

8.4.25 ProgramName element

The ProgramName element specifies the program name. This element is specified as xsd:string. The maximum length shall be 100 characters.

- Schema

```
<xsd:simpleType name="ProgramNameType">
  <xsd:restriction base="xsd:string">
    <xsd:maxLength value="100"/>
  </xsd:restriction>
</xsd:simpleType>
```

8.4.26 SceneNo. element

The SceneNo. element specifies the scene number. This element is specified as xsd:unsignedInt. This element value shall never be zero. This element is optional.

- Schema

```
<xsd:simpleType name="SceneNo.Type">
  <xsd:restriction base="xsd:unsignedInt">
    <xsd:minInclusive value="1"/>
  </xsd:restriction>
</xsd:simpleType>
```

8.4.27 TakeNo. element

The TakeNo. element specifies the take number. This element is specified as xsd:unsignedInt. This element value shall never be zero. This element is optional.

- Schema

```
<xsd:simpleType name="TakeNo.Type">
  <xsd:restriction base="xsd:unsignedInt">
    <xsd:minInclusive value="1"/>
  </xsd:restriction>
</xsd:simpleType>
```

8.4.28 News element

The News element describes the information of this news program. If the News element is used, at least one of its three Child Elements shall exist. This element is optional.

- Child Elements

Name	Definition
Reporter	Optional. 0 or 1 sibling only.
Purpose	Optional. 0 or 1 sibling only.
Object	Optional. 0 or 1 sibling only.

- Schema

```
<xsd:complexType name="NewsType">
  <xsd:choice>
    <xsd:sequence>
      <xsd:element name="Reporter" type="ReporterType"/>
      <xsd:element name="Purpose" type="PurposeType" minOccurs="0"/>
      <xsd:element name="Object" type="ObjectType" minOccurs="0"/>
    </xsd:sequence>
    <xsd:sequence>
      <xsd:element name="Purpose" type="PurposeType"/>
      <xsd:element name="Object" type="ObjectType" minOccurs="0"/>
    </xsd:sequence>
    <xsd:sequence>
      <xsd:element name="Object" type="ObjectType"/>
    </xsd:sequence>
  </xsd:choice>
</xsd:complexType>
```

8.4.29 Reporter element

The Reporter element specifies the reporter's name. This element is specified as xsd:normalizedString. The maximum length shall be 30 characters. This element is optional.

- Schema

```
<xsd:simpleType name="ReporterType">
  <xsd:restriction base="xsd:normalizedString">
    <xsd:maxLength value="30"/>
  </xsd:restriction>
</xsd:simpleType>
```

8.4.30 Purpose element

The Purpose element specifies the purpose of this news. This element is specified as xsd:string. The maximum length shall be 100 characters. This element is optional.

- Schema

```
<xsd:simpleType name="PurposeType">
  <xsd:restriction base="xsd:string">
    <xsd:maxLength value="100"/>
  </xsd:restriction>
</xsd:simpleType>
```

8.4.31 Object element

The Object element specifies the object of this news. This element is specified as xsd:string. The maximum length shall be 100 characters. This element is optional.

- Schema

```
<xsd:simpleType name="ObjectType">
  <xsd:restriction base="xsd:string">
    <xsd:maxLength value="100"/>
  </xsd:restriction>
</xsd:simpleType>
```

8.4.32 MemoList element

The MemoList element describes the list of memos. This element is optional.

- Child Elements

Name	Definition
Memo	Required. 1 to 100 siblings.

- Schema

```
<xsd:complexType name="MemoListType">
  <xsd:sequence>
    <xsd:element name="Memo" type="MemoType" maxOccurs="100"/>
  </xsd:sequence>
</xsd:complexType>
```

8.4.33 Memo element

The Memo element describes a memo.

- Child Elements

Name	Definition
Offset	Optional. 0 or 1 sibling only.
Person	Optional. 0 or 1 sibling only.
Text	Required. 1 sibling only.
Voice	Required. 1 sibling only.

NOTE 1 – The Voice element is optional, but shall be written where the Voice memo file exists. The Voice element and its child elements are defined in Annex B.

NOTE 2 – Either the Text element or the Voice element shall exist as the child element of the Memo element, but never both simultaneously.

- Attributes

Name	Definition
MemoID	An attribute that specifies the ID of this memo. Required. Specified as MemoIDType (integer of 0 to 99). Must be unique among all MemoIDs under MemoList.

- Schema

```
<xsd:complexType name="MemoType">
  <xsd:sequence>
    <xsd:element name="Offset" type="OffsetType" minOccurs="0"/>
    <xsd:element name="Person" type="PersonType"/>
    <xsd:choice>
      <xsd:element name="Text" type="TextType"/>
      <xsd:element name="Voice" type="VoiceType"/>
    </xsd:choice>
  </xsd:sequence>
  <xsd:attribute name="MemoID" type="MemoIDType" use="required"/>
</xsd:complexType>
<xsd:simpleType name="MemoIDType">
  <xsd:restriction base="xsd:nonNegativeInteger">
    <xsd:maxInclusive value="99"/>
  </xsd:restriction>
</xsd:simpleType>
```

8.4.34 Offset element

The Offset element specifies the offset of the frame, which this memo is related, in number of Edit Units (See Section 8.3.6). This element is specified as xsd:unsignedInt. This element is optional.

- Schema

```
<xsd:simpleType name="OffsetType">
  <xsd:restriction base="xsd:unsignedInt"/>
</xsd:simpleType>
```

8.4.35 Person element

The Person element specifies the creator of this memo. This element is specified as xsd:normalizedString. The maximum length shall be 30 characters. This element is optional.

- Schema

```
<xsd:simpleType name="PersonType">
  <xsd:restriction base="xsd:normalizedString">
    <xsd:maxLength value="30"/>
  </xsd:restriction>
</xsd:simpleType>
```

8.4.36 Text element

The Text element specifies the textual memo. This element is specified as xsd:string. The maximum length shall be 1000 characters.

- Schema

```
<xsd:simpleType name="TextType">
  <xsd:restriction base="xsd:string">
    <xsd:maxLength value="1000"/>
  </xsd:restriction>
</xsd:simpleType>
```

9 MXF Descriptive metadata implementation (informative)

This section describes the relationship between the Clip Descriptive Metadata elements stored in an XML Clip Metadata file and the MXF Descriptive Metadata Scheme-1 (DMS-1) properties defined in SMPTE 380M. The MXF DMS-1 properties should be implemented in MXF files according to the guideline shown in this section. Detailed information required for implementation of DMS-1 is described in SMPTE 380M and EG 42.

9.1 DM Framework

A DM framework is defined as a group of related Descriptive Metadata properties and sets which describe the contents of an MXF file body. The Clip Descriptive Metadata provides capture and creation information about an audio and a video essence in the file body. Therefore, the Clip Framework is created in a DM track and the Clip Descriptive Metadata elements described in section 8 are mapped to the corresponding Metadata properties contained in the Clip Framework.

In addition, the Clip Descriptive Metadata contains information about production and scene in the Scenario element. The Metadata elements are mapped to the Metadata properties in the Production Framework and the Scene Framework where these elements exist.

9.2 DM Tracks, DM Sequence, DM Segments

Figure 9.1 illustrates one of the structures of the DM framework contained in the Header Metadata.

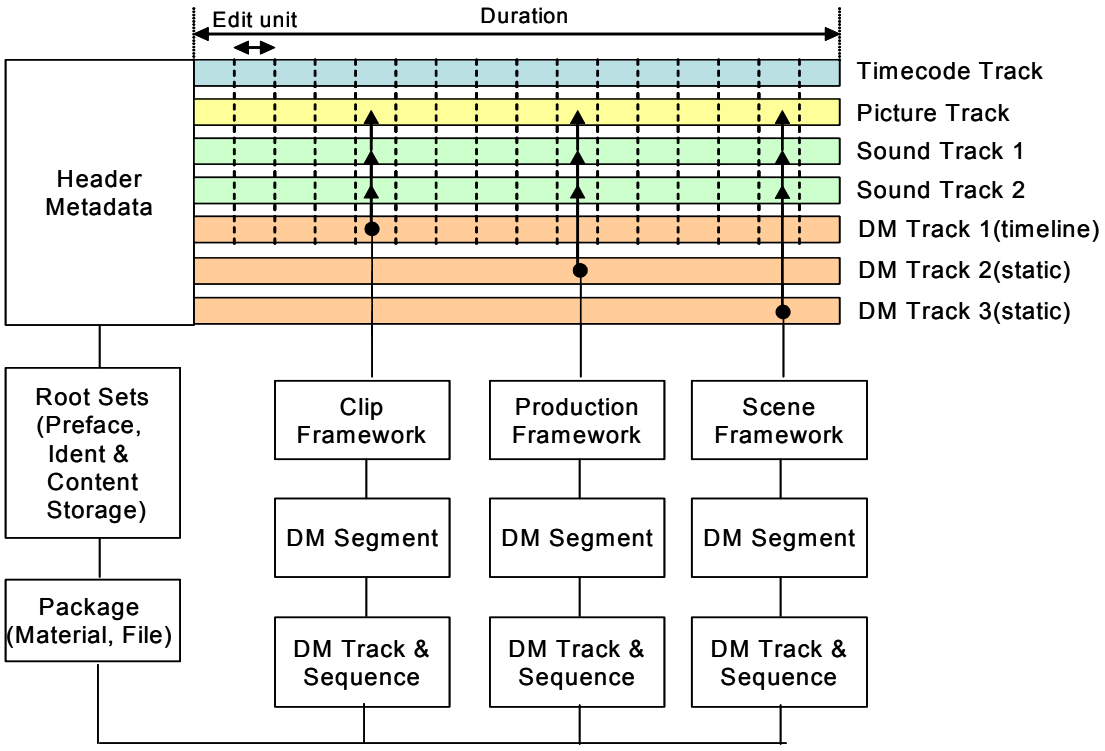


Figure 9.1 – Example of structure of the DM framework in the Header Metadata

9.2.1 Clip Framework

A DM Track is set as a static track or a timeline track in the Material Package or the Top-level File Package and has a Sequence set.

The Sequence set has one DM Segment which connects to the Clip Framework.

The duration set in the DM Segment is the same as that in the DM Track.

NOTE – The track kind should be decided depending on implementation of Metadata properties.

9.2.2 Production Framework

A DM Track is set as a static track in the Material Package or the Top-level File Package and has a Sequence set.

The Sequence set has one DM Segment which connects to the Production Framework.

9.2.3 Scene Framework

A DM Track is set as a static track in the Material Package or the Top-level File Package and has a Sequence set.

The Sequence set has one DM Segment which connects to the Scene Framework.

9.3 Universal Label for Descriptive Metadata Scheme

Universal Label for Descriptive Metadata Scheme-1 is defined in SMPTE 380M. Byte 14 of the Universal label is set depending on the Framework.

9.4 Clip Metadata mapping to DMS-1

This sub section describes the mapping from the Clip Descriptive Metadata elements to the DMS-1 properties. The Clip Descriptive Metadata elements using an XML are converted to the KLV coding and are mapped to the DMS-1 properties.

Figure 9.2 illustrates a data model of the Framework, sets and properties for the Clip Descriptive Metadata.

As metadata description, Metadata properties defined in the DMS-1 are expressed with (Framework)/(Set)/(Name). For example, ClipFramework/Titles/MainTitle indicates the Main Title property in the Title Set in the Clip Framework.

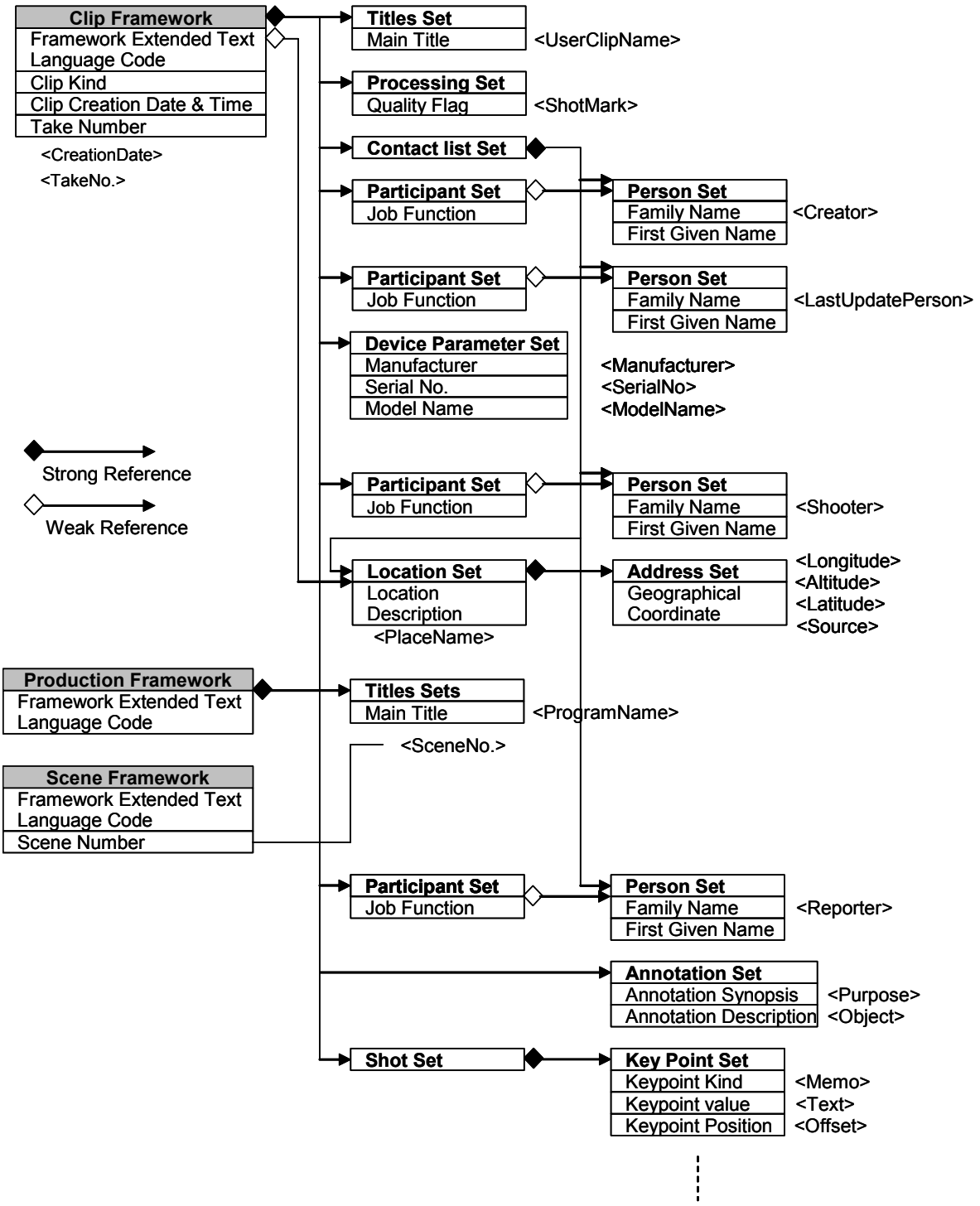


Figure 9.2 – Model of Framework, sets and properties for the Clip Descriptive Metadata

9.4.1 ClipMetadata element

This element is not mapped to a DMS-1 property.

9.4.2 UserClipName element

The UserClipName element is mapped to the ClipFramework/Titles/MainTitle.

9.4.3 DataSource element

There is no Metadata property corresponding to the DataSource element. Therefore, this element should be mapped to a custom element in the Clip Framework.

9.4.4 ShotMark element

The ShotMark element is mapped to the ClipFramework/Processing/QualityFlag.

NOTE – The meaning of the QualityFlag value is defined in DMS-1 while the meaning of the ShotMark value can be defined depending on users. Therefore, the ShotMark value must be converted to the QualityFlag value so there is no contradiction between them.

9.4.5 Access element

This element is not mapped to a DMS-1 property.

9.4.6 Creator element

The Creator element and its value are mapped to the Clip Framework/Participant/(Job Function or Role) and the connect to the Clip Framework/Person Set which may contain Family Name and First Given Name. Value of the Clip Framework/Participant/Job Function is set to “Creator”.

NOTE – The Person Set is the strong reference from the Contact List Set.

9.4.7 CreationDate element

The CreationDate element is mapped to the ClipFramework/ClipCreationDateAndTime.

9.4.8 LastUpdatePerson element

The LastUpdatePerson element and its value are mapped to the Clip Framework/Participant/(Job Function or Role) and the connect to the Clip Framework/Person Set which may contain Family Name and First Given Name. Value of the Clip Framework/Participant/Job Function is set to “Last Update Person”.

NOTE – The Person Set is the strong reference from the Contact List Set.

9.4.9 LastUpdateDate element

The LastUpdateDate element is not mapped to any DMS-1 properties. However, the value exists in the Last Modified Date property in the Preface Set and the Modification Date property in the Identification Set.

9.4.10 Device element

This element is not mapped to a DMS-1 property.

9.4.11 Manufacturer element

The Manufacturer element is mapped to the Clip Framework/Device Parameters/Manufacturer.

9.4.12 SerialNo. element

The SerialNo. element is mapped to the Clip Framework/Device Parameters/Serial Number.

9.4.13 ModelName element

The ModelName element is mapped to the Clip Framework/Device Parameters/Device Model.

9.4.14 Shoot element

This element is not mapped to a DMS-1 property.

9.4.15 Shooter element

The Shooter element and its value are mapped to the Clip Framework/Participant/(Job Function or Role) and the connect to the Clip Framework/Person Set which may contain Family Name and First Given Name. Value of the Clip Framework/Participant/Job Function is set to “Shooter”.

NOTE – The Person Set is the strong reference from the Contact List Set.

9.4.16 StartDate element

The StartDate element is mapped to the Clip Framework/Clip Creation Date & Time.

9.4.17 EndDate element

There is no Metadata property corresponding to the EndDate element. Therefore, this element should be mapped to the custom element in the Clip Framework.

9.4.18 Location element

This element is not mapped to a DMS-1 property.

9.4.19 Altitude, the Longitude, the Latitude and the Source elements

The Altitude, the Longitude and the Latitude elements are converted to the 12-byte spatial coordinate defined in UMID and mapped to the Clip Framework/Location/Address/Geographical Coordinate. The confidence level of the coordinate information is set in the 12-byte spatial coordinate in the UMID. Therefore, the <Source> element will be converted to confidence level information within the UMID.

9.4.20 PlaceName element

The PlaceName element is mapped to the Clip Framework/Location/Location Description.

NOTE – The Location Set is the strong reference from the Contact List Set.

9.4.21 Scenario element

This element is not mapped to a DMS-1 property.

9.4.22 ProgramName element

The ProgrameName element indicates the title of a parent Program. Therefore, this element should be mapped to the Production Framework/Titles/Main Title.

9.4.23 SceneNo. element

The SceneNo. element relates a part of a Program. Therefore, this element should be mapped to the Scene Framework/Scene Number.

9.4.24 TakeNo. element

The TakeNo. element is mapped to the Clip Framework/Take Number.

9.4.25 News element

This element is not mapped to a DMS-1 property.

9.4.26 Reporter element

The Reporter element and its value are mapped to the Clip Framework/Participant/(Job Function or Role) and the connect to the Clip Framework/Person Set which may contain Family Name and First Given Name. Value of the Clip Framework/Participant/Job Function is set to “Reporter”.

NOTE – The Person set is the strong reference from the Contact List Set.

9.4.27 Purpose element

The Purpose element is mapped to the Clip Framework/Annotation/Annotation Synopsis.

9.4.28 Object element

The Object element is mapped to the Clip Framework/Annotation/Annotation Description.

9.4.29 MemoList element

If the MemoList element exists, the Shot set which connects to the Key Point Set each of which corresponds to each memo is created.

9.4.30 Memo element

The Memo element is mapped to the Clip Framework/Shot/Key Point/Key Point Kind and the value of the Key Point Kind is set to "Memo" or "Text memo".

9.4.31 Offset element

The Offset element is mapped to the Clip Framework/Shot/Key Point/Key Point Position.

9.4.32 Person element

The Person element is not mapped to a property in the Key Point Set. However the value may be described to the head of the Keypoint Value property shown in the next.

9.4.33 Text element

The Text element is mapped to the Clip Framework/Key Point/Keypoint Value.

Annex A (Informative)
Header Metadata Set for MXF file

A.1 Preface Set

Table A.1 – Preface Set Item list

Item Name	Type	Value
Preface	Set Key	Set Key, Set Length and Instance UID values are defined in 377M
Length	BER Length	
Instance UID	UUID	
Last Modified Date	Timestamp	Created by the application
Version	Version Type	258
Primary Package	WeakRef (Package)	UID of Top level File Package
Identifications	StrongRefArray (Identification)	UIDs of Identification Sets
Content Storage	StrongRef (ContentStorage)	UID of Content Storage Set
Operational Pattern	UL	OP-Atom
Essence Containers	ULBatch (Essence Containers)	See 6.1.2 and 7.1.2
DM Schemes	ULBatch (DMSchemes)	ULs of the DM Frameworks

A.2 Identification Set

Table A.2 – Identification Set Item List

Item Name	Type	Value
Identification	Set Key	Set Key, Set Length and Instance UID values are defined in 377M
Length	BER Length	
Instance UID	UUID	
This Generation UID	UUID	16 byte UUID value created at the time this set was created
Company Name	UTF-16 string	Created by the application
Product Name	UTF-16 string	Created by the application
Version String	UTF-16 string	Created by the application
Product UID	UUID	Created by the application
Modification Date	Timestamp	Created by the application

A.3 Content Storage Set

Table A.3 – Content Storage Set Item list

Item Name	Type	Value
Content Storage	Set Key	Set Key, Set Length and Instance UID values are defined in 377M
Length	BER Length	
Instance UID	UUID	
Packages	StrongRef Batch (Packages)	UID of Material Package and Top level File Package
EssenceContainerData	StrongRef Batch (Container Data)	UID of Essence Container Set

A.4 Essence Container Data Set

Table A.4 – Essence Container Data Set

Item Name	Type	Value
Essence Container Data	Set Key	Set Key, Set Length and Instance UID values are defined in 377M
Length	BER Length	
Instance UID	UUID	
Linked Package UID	UMID	UMID of Top level File package
IndexSID	UInt32	Set to the same value as the IndexSID in the Header Partition Pack
BodySID	UInt32	Set to the same value as the BodySID in the Body Partition Pack

A.5 Material Package

Table A.5 – Material package Item list

Item Name	Type	Value
Material Package	Set Key	Set Key, Set Length and Instance UID values are defined in 377M
Length	BER Length	
Instance UID	UUID	
Package UID	UMID	UMID of this Material Package
Package Creation Date	Timestamp	Created by the application
Package Modified Date	Timestamp	Created by the application
Tracks	StrongRefArray (Tracks)	UID of Timecode Track Set, Picture Track Set and Sound Track Set

A.5.1 Timecode Track

Table A.6 –Timecode Track set Item list

Item Name	Type	Value
Track	Set Key	Set Key, Set Length and Instance UID values are defined in 377M
Length	BER Length	
Instance UID	UUID	
Track ID	UInt32	Created by the application
Track Number	UInt32	1
Edit Rate	Rational	{30000,1001}, {25,1}
Origin	Position	0 (Default)
Sequence	StrongRef (Sequence)	UID of Timecode Sequence set

Table A.7 – Timecode Sequence set Item list

Item Name	Type	Value
Sequence	Set Key	Set Key, Set Length and Instance UID values are defined in 377M
Length	BER Length	
Instance UID	UUID	
Data Definition	UL	{06.0E.2B.34.04.01.01.01.01.03.02.01.01.00.00.00} as defined in SMPTE RP224
Duration	Length	Number of frames present
Structural Components	StrongRefArray (Structural Component)	UID of Timecode Component Set

Table A.8 – Timecode Component set Item list

Item Name	Type	Value
Timecode Component	Set Key	Set Key, Set Length and Instance UID values are defined in 377M
Length	BER Length	
Instance UID	UUID	
Data Definition	UL	{06.0E.2B.34.04.01.01.01.01.03.02.01.01.00.00.00} as defined in SMPTE RP224
Duration	Length	Number of frames present
Rounded Timecode Base	Uint16	30,25
Start Timecode	Position	Converted to integer frame count from 00:00:00:00
Drop Frame	Boolean	0: Non-drop frame, 1 Drop frame

A.5.2 Picture track

Table A.9 – Picture Track set Item list

Item Name	Type	Value
Track	Set Key	Set Key, Set Length and Instance UID values are defined in 377M
Length	BER Length	
Instance UID	UUID	
Track ID	UInt32	Created by the application
Track Number	UInt32	1
Edit Rate	Rational	{30000,1001}, {25,1}
Origin	Position	0 (Default)
Sequence	StrongRef (Sequence)	UID of Picture Sequence Set

Table A.10 – Picture Sequence set Item list

Item Name	Type	Value
Sequence	Set Key	Set Key, Set Length and Instance UID values are defined in 377M
Length	BER Length	
Instance UID	UUID	
Data Definition	UL	{06.0E.2B.34.04.01.01.01.01.03.02.02.01.00.00.00} as defined in SMPTE RP224
Duration	Length	Number of frames present
Structural Components	StrongRefArray (Structural Component)	UID of Picture SourceClip Set

Table A.11 – Picture SourceClip set Item list

Item Name	Type	Value
SourceClip	Set Key	Set Key, Set Length and Instance UID values are defined in 377M
Length	BER Length	
Instance UID	UUID	
Data Definition	UL	{06.0E.2B.34.04.01.01.01.01.03.02.02.01.00.00.00} as defined in SMPTE RP224
Start Position	Position	0 (Default)
Duration	Length	Number of frames present
SourcePackageID	Package ID	UMID of Top-level File Package in a Video Essence file
SourceTrackID	Track ID (UInt32)	Track ID of Picture track in Top level File Package

A.5.3 Sound Track

Table A.12 – Sound Track set Item list

Item Name	Type	Value
Track	Set Key	Set Key, Set Length and Instance UID values are defined in 377M
Length	BER Length	
Instance UID	UUID	
Track ID	UInt32	Created by the application
Track Number	UInt32	1: channel 1, 2: channel 2, ….
Edit Rate	Rational	{48000,1}
Origin	Position	0 (Default)
Sequence	StrongRef (Sequence)	UID of Sound Sequence Set

Table A.13 – Sound Sequence set Item list

Item Name	Type	Value
Sequence	Set Key	Set Key, Set Length and Instance UID values are defined in 377M
Length	BER Length	
Instance UID	UUID	
Data Definition	UL	{06.0E.2B.34.04.01.01.01.01.03.02.02.02.00.00.00} as defined in SMPTE RP224
Duration	Length	Number of samples present
Structural Components	StrongRefArray (Structural Components)	UID of Sound SourceClip Set

Table A.14 – Sound SourceClip Item list

Item Name	Type	Value
SourceClip	Set Key	Set Key, Set Length and Instance UID values are defined in 377M
Length	BER Length	
Instance UID	UUID	
Data Definition	UL	{06.0E.2B.34.04.01.01.01.01.03.02.02.02.00.00.00} as defined in SMPTE RP224
Start Position	Position	0 (Default)
Duration	Length	Number of samples present
SourcePackageID	Package ID	UMID of Top level File Package in an Audio Essence file
SourceTrackID	Track ID (UInt32)	Track ID of Sound track in Top level File Package

A.6 Top level File Package

Table A.15 – Top level File package Item list

Item Name	Type	Value
File Package	Set Key	Set Key, Set Length and Instance UID values are defined in 377M
Length	BER Length	
Instance UID	UUID	
Package UID	UMID	UMID of this Top level File Package
Package Creation Date	Timestamp	Created by the application
Package Modified Date	Timestamp	Created by the application
Tracks	StrongRefArray (Tracks)	Video Essence file : UID of Picture Track Set Audio Essence file : UID of Sound Track Set

A.6.1 Picture track

Table A.16 – Picture Track set Item list

Item Name	Type	Value
Track	Set Key	Set Key, Set Length and Instance UID values are defined in 377M
Length	BER Length	
Instance UID	UUID	
Track ID	UInt32	Created by the application
Track Number	UInt32	Bytes 13 to 16 of the DV-DIF Compound Element Key
Edit Rate	Rational	{30000,1001}, {25,1}
Origin	Position	0 (Default)
Sequence	StrongRef (Sequence)	UID of Picture Sequence Set

Table A.17 – Picture Sequence set Item list

Item Name	Type	Value
Sequence	Set Key	Set Key, Set Length and Instance UID values are defined in 377M
Length	BER Length	
Instance UID	UUID	
Data Definition	UL	{06.0E.2B.34.04.01.01.01.01.03.02.02.01.00.00.00} as defined in SMPTE RP224
Duration	Length	Number of frames present
Structural Components	StrongRefArray (Structural Component)	UID of Picture SourceClip Set

Table A.18 – Picture SourceClip set Item list

Item Name	Type	Value
SourceClip	Set Key	Set Key, Set Length and Instance UID values are defined in 377M
Length	BER Length	
Instance UID	UUID	
Data Definition	UL	{06.0E.2B.34.04.01.01.01.01.03.02.02.01.00.00.00} as defined in SMPTE RP224
Start Position	Position	0 (Default)
Duration	Length	Number of frames present
SourcePackageID	Package ID	0 (Default)
SourceTrackID	Track ID (UInt32)	0 (Default)

A.6.2 Sound Track

Table A.19 – Sound Track set Item list

Item Name	Type	Value
Track	Set Key	Set Key, Set Length and Instance UID values are defined in 377M
Length	BER Length	
Instance UID	UUID	
Track ID	UInt32	Created by the application
Track Number	UInt32	Bytes 13 to 16 of the Audio Essence Element Key
Edit Rate	Rational	{48000,1}
Origin	Position	0 (Default)
Sequence	StrongRef (Sequence)	UID of Sound Sequence Set

Table A.20 – Sound Sequence set Item list

Item Name	Type	Value
Sequence	Set Key	Set Key, Set Length and Instance UID values are defined in 377M
Length	BER Length	
Instance UID	UUID	
Data Definition	UL	{06.0E.2B.34.04.01.01.01.01.03.02.02.02.00.00.00} as defined in SMPTE RP224
Duration	Length	Number of samples present
Structural Components	StrongRefArray (Structural Components)	UID of Sound SourceClip Set

Table A.21 – Sound SourceClip Item list

Item Name	Type	Value
SourceClip	Set Key	Set Key, Set Length and Instance UID values are defined in 377M
Length	BER Length	
Instance UID	UUID	
Data Definition	UL	{06.0E.2B.34.04.01.01.01.01.03.02.02.02.00.00.00} as defined in SMPTE RP224
Start Position	Position	0 (Default)
Duration	Length	Number of samples present
SourcePackageID	Package ID	0 (Default)
SourceTrackID	Track ID (UInt32)	0 (Default)

Annex B (Normative)
Structures of application specific files

The structures of the optional files, specifically Thumbnail, Voice and Proxy, are defined in this Annex.

B.1 Naming rules of optional files

The name shall consist of six or eight characters and an extension. Six characters shall be used for a Thumbnail file, a Proxy file and a Real time metadata file. Eight characters shall be used for a Voice memo. The extension shall be 3 characters. The file name and the extensions shall be case-sensitive and expressed with upper case.

- The first six characters shall represent a Clip name as defined in section 4.4.5.
- The next two characters shall represent the memo ID of the Voice memo file and shall have a value of “00” to “99” in decimal representation.

These files shall use the same Clip name as the Essence files in the Clip.

B.2 Thumbnail file

A Thumbnail file shall be an 80 x 60 pixel bitmap file used as an icon that marks a section of the recorded/stored image.

The Thumbnail file shall be stored in the “ICON” directory which is located under the “CONTENTS” directory. The name of a Thumbnail file shall consist of a Clip name and an extension ‘.BMP’. An example of such a name file is “0001AB.BMP. The detailed specification of the bitmap file is described in Annex D.

B.3 Voice memo file

A Voice memo file shall be a single channel WAVE format file that contains audio sampled at 12 kHz sampling rate and 16 bits/sample. The maximum number of the Voice memo files shall be 100.

The Voice memo file shall be stored in the “VOICE” directory which is located under the “CONTENTS” directory. The name of the Voice memo file shall consist of a Clip name, Voice memo ID and an extension ‘.WAV’. An example of such a name file is “0001AB00.WAV”. The detailed specification of the wave file is described in Annex E.

B.4 Proxy AV file

A Proxy AV file is a file that stores low bit rate compressed video and audio data.

The name of the Proxy AV file shall consist of the related Clip name and extension ‘.MP4’, such as "0001AB.MP4" where "0001AB" is the related Clip name and ".MP4" is the extension.

B.4.1 Structure of Proxy AV File

A Proxy AV file shall be compliant with the MP4 file format as defined in ISO/IEC 14496-12. The Proxy AV file is structured with a series of objects called ‘boxes’. The Proxy AV file shall start with ‘ftyp’ (File type box) and be followed by Media Data and Metadata as specified in Figure B.1. The Media Data contains compressed video and audio data. The Metadata contains information about the structure and the descriptions of the Media Data. The Media Data shall be stored in ‘mdat’ (media data box) and the Metadata shall be stored in ‘moov’ (movie box) and ‘skip’ (skip box).

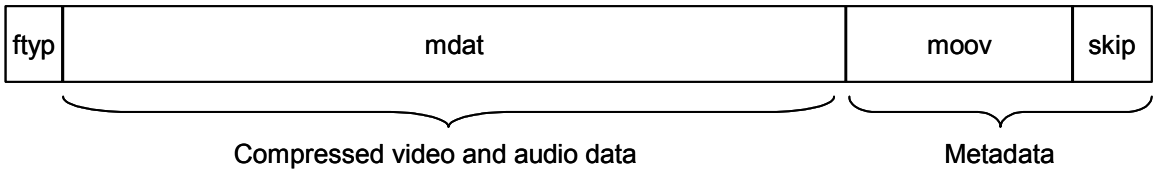


Figure B.1 – Structure of Proxy AV File

B.4.2 Box Definitions

Table B.1 shows the box list which are supported in the Proxy AV file.

NOTE 1 – Box type in Table B.1 is represented with hierarchy from the left column respectively.

NOTE 2 – Box sizes are defined in ISO/IEC 14496-12.

Table B.1 – Box type and definition

box type						Definition
ftyp						File type and compatibility
mdat						Media data container
moov						Container for all the meta-data
	mvhd					Movie header, overall declarations
	track					Container for an individual track or stream
		tkhd				Track header, overall information about the track
		tref				Track reference container
		media				Container for the media information in a track
			mdhd			Media header, overall information about the media
			hdlr			Handler, declares the media (handler) type
			minf			Media information container
				vmhd		Video media header, overall information (video track only)
				smhd		Sound media header, overall information (sound track only)
				dinf		Data information box, container
					dref	Data reference box, declares source(s) of media data in track
				stbl		Sample table box, container for the time/space map
					stsd	Sample descriptions (codec types, initialization etc.)
					Stts	(Decoding) time-to-sample
					Ctts	(Composition) time to sample
					stsc	Sample- to–chunk, partial data-offset information
					stsz	Sample sizes (framing)
					stco	Chunk offset, partial data-offset information
					stss	Sync sample table (random access points)
skip						See Table B.2

Specific Metadata for a Proxy AV file shall be stored in ‘skip’ (skip box). ‘skip’ shall be located at the end of the file and shall contain Clip name, Global Clip ID, Time code and SSM Card serial number as shown in Table B.2.

Table B.2 – Definition of skip box

box type				Data Size (Number of byte)	Definition
skip					Free space
	udta				User data
		P2dt			Metadata
			clpn	6	Clip name
			umid	32	Basic UMID which is the same as Global Clip ID in the Clip Metadata.(see 8.2.4)
			sttc	8	Time code and binary group associated with the start frame of the video data stored in ‘mdat’. The time code and the binary group are specified in SMPTE 12M. The first 4 bytes indicates the time code and the last 4 bytes indicates the binary group.
			P2sn	20	SSM Card serial number Where the SSM Card serial number is less than 20 bytes, the active part of the number shall occupy the first part of the 20 bytes and the remainder shall be filled with the NULL code (00h).

B.4.3 Video Compression

Compression for video data shall be compliant with MPEG-4 visual as defined in ISO/IEC 14496-2.

Profile :	Advanced Real Time Simple Profile Level 4
Video resolution / Frame rate:	(SIF) 352x240 / 29.97Hz , 29.97/2Hz
	(SIF) 352x288 / 25Hz , 25/2Hz
	(QVGA) 320x240 / 29.97Hz , 29.97/2Hz , 25Hz , 25/2Hz
Bit rate after compression :	384 kbps – 2 Mbps for 29.97Hz and 25Hz,
	192 kbps – 383 kbps for 29.97/2Hz and 25/2Hz

B.4.4 Audio Compression

Compression for audio data shall be compliant with MPEG-4 audio as defined in ISO/IEC 14496-3.

Profile :	AAC
Sampling rate / Quantization:	24 kHz/16 bit
Bit rate after compression :	32 kbps, 64 kbps
Number of channels:	Maximum 2 channels

B.5 Real time Metadata file

A Real time Metadata file is a file that stores metadata which synchronize with video and audio data stored in the Proxy AV file. The name of the Real Time Metadata file shall consist of related Clip name and extension 'BIN', such as "0001AB.BIN" where "0001AB" is the related Clip name and "BIN" is the extension.

B.5.1 Structure of Real Time Metadata file

The Real Time Metadata file shall consist of the Header Block and a sequence of Real Time Metadata items for each frame. The Real Time Metadata file shall start with the Header Block and be followed by the Metadata items. The Basic structure of the Real Time Metadata file is shown in Figure B.2.

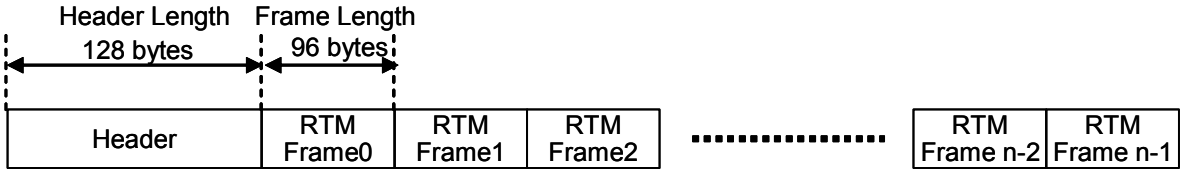


Figure B.2 – Structure of Real Time Metadata File

B.5.2 Header Block

The Header Block shall be placed in the start of the Real time Metadata file and contains information about the structure of the Real time Metadata and a Global clip ID to link to the Clip Metadata. The total size of the Header Block shall be 128 bytes. The items in the Header Block shall be as given in Table B.3.

Table B.3 – Header Block item

Item Name	Number of byte	Definition	Value (hex)
ID	4	ID of this Header Block	"RTMH" in ISO/IEC 646
Header Length	4	Count of bytes of the Header Block	128
Frame Length	4	Count of bytes of Real time Metadata items in each frame	96
Version No.	4	Version number of Real time Metadata Version number is represented with a pair of 2 bytes values. The first 2 bytes is the Major version number and the second 2 bytes is the Minor version number.	var
Global Clip ID	32	Basic UMID which is the same as a Global Clip ID in the Clip Metadata (See 8.3.4)	var
RTM Frame Rate	8	Frame Rate of the Real Time Metadata Frame rate in Hz is represented with a pair of 4 bytes values. The first 4 bytes is the numerator and the second 4 bytes is the denominator. (E.g. for 29.97Hz, Frame Rate is represented as 30000/1001 in decimal format).	var

<table>
<tr><th>Item Name</th><th>Number of byte</th><th>Definition</th><th>Value (hex)</th></tr>
<tr><td>Meta Descriptor</td><td>2</td><td>Validity of Real Time Metadata items defined in Table B.4
1: Metadata item is valid
0: Metadata item is invalid
<table><tr><th>Bit position</th><th>Real Time Metadata item</th></tr><tr><td>0(LSB)</td><td>LTC</td></tr><tr><td>1</td><td>LTC UB</td></tr><tr><td>2</td><td>VITC</td></tr><tr><td>3</td><td>VITC UB</td></tr><tr><td>4</td><td>UMID</td></tr><tr><td>5-15(MSB)</td><td>Reserved for future definition and shall be set to 0</td></tr></table></td><td>var</td></tr>
<tr><td>Reserved</td><td>70</td><td>Reserved for future definition and shall be set to 0</td><td></td></tr>
</table>

NOTE 1 – Byte order of Length, Version No., RTM Frame Rate and Meta descriptor items is big endian (Most significant byte first).

NOTE 2 – The GlobalClipID defined as a text string in the Clip Metadata is converted to a binary UMID as the Global Clip ID for the Real time metadata.

B.6 Real Time Metadata item

Real time Metadata items shall be placed after the Header Block and contain Metadata which is synchronized with video and audio data in a Proxy AV file. The total size of the Real Time Metadata items for a frame shall be fixed to 96 bytes. The items in the Real time Metadata for each frame shall be as given in Table B.4. In the case that there is no valid data associated with a Proxy AV file, the value of the corresponding Meta descriptor given in Table B.3 shall be set to invalid.

Table B.4 – Real Time Metadata item

Item Name	Number of byte	Definition
LTC	4	LTC value associated with a frame
LTC UB	4	User bit value in LTC associated with a frame
VITC	4	VITC value associated with a frame
VITC UB	4	User bit value in VITC associated with a frame
UMID	52	Extended UMID which is set in the File Body in a MXF video file. The value of the UMID does not include 12 byte Universal Label, which shall have the same value as that for the UMID (Global Clip ID)stored in the Header Block. (see Table B.3)
Reserved	28	Reserved for future definition and shall be set to 0

B.7 Clip Descriptive metadata element for Voice memo, Thumbnail and Proxy

B.7.1 Namespace

The Schemas for the elements described in this Annex shall be under the same namespace value as in the main body (section 8.2).

B.7.2 Metadata elements

Figure B.3 shows the overview of the elements for the optional files.

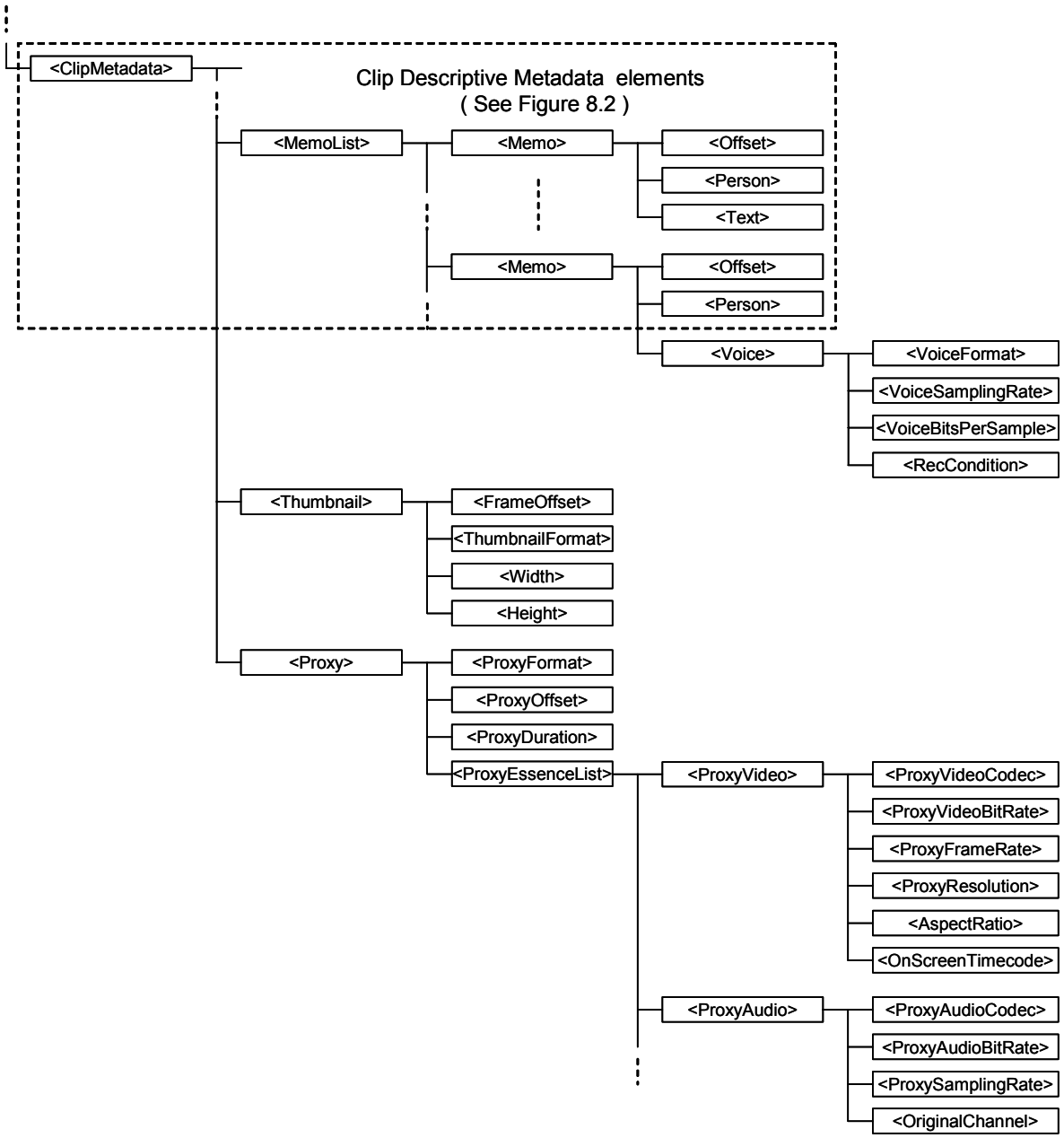


Figure B.3 – Overview of Clip Metadata elements for optional files

B.7.2.1 Voice element

The Voice element describes a Voice memo. This element is optional.

- Child Elements

Name	Definition
VoiceFormat	Required. 1 sibling only.
VoiceSamplingRate	Required. 1 sibling only.
VoiceBitsPerSample	Required. 1 sibling only.
RecCondition	Optional. 0 or 1 sibling only.

- Schema

```
<xsd:complexType name="VoiceType">
  <xsd:sequence>
    <xsd:element name="VoiceFormat" type="VoiceFormatType"/>
    <xsd:element name="VoiceSamplingRate" type="VoiceSamplingRateType"/>
    <xsd:element name="VoiceBitsPerSample" type="VoiceBitsPerSampleType"/>
    <xsd:element name="RecCondition" type="RecConditionType" minOccurs="0"/>
  </xsd:sequence>
</xsd:complexType>
```

B.7.2.2 VoiceFormat element

The VoiceFormat element specifies the Voice file format.

- Value

Value	Definition
WAV	WAVE file.

- Schema

```
<xsd:simpleType name="VoiceFormatType">
  <xsd:restriction base="xsd:NMTOKEN">
    <xsd:enumeration value="WAV"/>
  </xsd:restriction>
</xsd:simpleType>
```

B.7.2.3 VoiceSamplingRate element

The VoiceSamplingRate element specifies the sampling rate of this Voice memo.

- Value

Value	Definition
12000	12000 samples/sec.

- Schema

```
<xsd:simpleType name="VoiceSamplingRateType">
  <xsd:restriction base="xsd:positiveInteger">
    <xsd:enumeration value="12000"/>
  </xsd:restriction>
</xsd:simpleType>
```

B.7.2.4 VoiceBitsPerSample element

The VoiceBitsPerSample element specifies the number of bits per sample of this Voice memo.

- Value

Value	Definition
16	16 bits/sample.

- Schema

```
<xsd:simpleType name="VoiceBitsPerSampleType">
  <xsd:restriction base="xsd:positiveInteger">
    <xsd:enumeration value="16"/>
  </xsd:restriction>
</xsd:simpleType>
```

B.7.2.5 RecCondition element

The RecCondition element specifies the recording condition of this Voice memo. This element is optional.

- Value

Value	Definition
PLAY	Recorded while playing this Clip.
STILL	Recorded while not playing this Clip.

- Schema

```
<xsd:simpleType name="RecConditionType">
  <xsd:restriction base="xsd:NMTOKEN">
    <xsd:enumeration value="PLAY"/>
    <xsd:enumeration value="STILL"/>
  </xsd:restriction>
</xsd:simpleType>
```

B.7.2.6 Thumbnail element

The Thumbnail element describes the Thumbnail. This element is optional.

- Child Elements

Name	Definition
FrameOffset	Required. 1 sibling only.
ThumbnailFormat	Required. 1 sibling only.
Width	Required. 1 sibling only.
Height	Required. 1 sibling only.

- Schema

```
<xsd:complexType name="ThumbnailType">
  <xsd:sequence>
    <xsd:element name="FrameOffset" type="FrameOffsetType"/>
    <xsd:element name="ThumbnailFormat" type="ThumbnailFormatType"/>
    <xsd:element name="Width" type="WidthType"/>
    <xsd:element name="Height" type="HeightType"/>
  </xsd:sequence>
</xsd:complexType>
```

B.7.2.7 FrameOffset element

The FrameOffset element specifies the offset of the frame, in number of frames, to which this Thumbnail is related. This element is specified as xsd:unsignedInt. This element is optional.

- Schema

```
<xsd:simpleType name="FrameOffsetType">
  <xsd:restriction base="xsd:unsignedInt"/>
</xsd:simpleType>
```

B.7.2.8 ThumbnailFormat element

The ThumbnailFormat element specifies the Thumbnail image format.

- Value

Value	Definition
BMP	Bitmap file.

- Schema

```
<xsd:simpleType name="ThumbnailFormatType">
  <xsd:restriction base="xsd:NMTOKEN">
    <xsd:enumeration value="BMP"/>
  </xsd:restriction>
</xsd:simpleType>
```

B.7.2.9 Width element

The Width element specifies the number of horizontal active pixels. This element is specified as xsd:unsignedInt and the value shall never be zero. This element is optional.

- Schema

```
<xsd:simpleType name="WidthType">
  <xsd:restriction base="xsd:unsignedInt">
    <xsd:pattern value="80"/>
  </xsd:restriction>
</xsd:simpleType>
```

B.7.2.10 Height element

The Height element specifies the number of vertical active pixels. This element is specified as xsd:unsignedInt and the value shall never be zero. This element is optional.

- Schema

```
<xsd:simpleType name="HeightType">
  <xsd:restriction base="xsd:unsignedInt">
    <xsd:pattern value="60"/>
  </xsd:restriction>
</xsd:simpleType>
```

B.7.2.11 Proxy element

The Proxy element describes a Proxy. This element is optional.

- Child Elements

Name	Definition
ProxyFormat	Required. 1 sibling only.
ProxyOffset	Optional. 0 or 1 sibling only.
ProxyDuration	Optional. 0 or 1 sibling only.
ProxyElementList	Required. 1 sibling only.

- Schema

```
<xsd:complexType name="ProxyType">
  <xsd:sequence>
    <xsd:element name="ProxyFormat" type="ProxyFormatType"/>
    <xsd:element name="ProxyOffset" type="ProxyOffsetType" minOccurs="0"/>
    <xsd:element name="ProxyDuration" type="ProxyDurationType" minOccurs="0"/>
    <xsd:element name="ProxyElementList" type="ProxyElementListType"/>
  </xsd:sequence>
</xsd:complexType>
```

B.7.2.12 ProxyFormat element

The ProxyFormat element specifies the file format of the Proxy.

- Value

Value	Definition
MP4	MPEG-4 file format (ISO/IEC 14496).

- Schema

```
<xsd:simpleType name="ProxyFormatType">
  <xsd:restriction base="xsd:NMTOKEN">
    <xsd:enumeration value="MP4"/>
  </xsd:restriction>
</xsd:simpleType>
```

B.7.2.13 ProxyOffset element

The ProxyOffset element specifies the difference between the start of this Proxy and the start of the original Clip in numbers of Edit Units (See Section 8.3.6). This element is specified as xsd:int. This element is optional and only specified when this difference exists. The difference should be specified in a positive value when the Proxy starts after the original Clip and in a negative value when the Proxy starts before the original Clip.

- Schema

```
<xsd:simpleType name="ProxyOffsetType">
  <xsd:restriction base="xsd:int"/>
</xsd:simpleType>
```

B.7.2.14 ProxyDuration element

The ProxyDuration element specifies the duration of this Proxy in numbers of Edit Units (See Section 8.3.6). This element is specified as xsd:unsignedInt and the value shall never be zero. This element is optional and is only specified when the duration of this Proxy is different from the original Clip.

- Schema

```
<xsd:simpleType name="ProxyDurationType">
  <xsd:restriction base="xsd:unsignedInt">
    <xsd:minInclusive value="1"/>
  </xsd:restriction>
</xsd:simpleType>
```

B.7.2.15 ProxyElementList element

The ProxyElementList element describes a list of elements that constitute this Proxy.

- Child Elements

Name	Definition
ProxyVideo	Required. 1 sibling only.
ProxyAudio	Optional. 0 to 2 siblings only.

- Schema

```
<xsd:complexType name="ProxyElementListType">
  <xsd:sequence>
    <xsd:element name="ProxyVideo" type="ProxyVideoType"/>
    <xsd:element name="ProxyAudio" type="ProxyAudioType" minOccurs="0"
maxOccurs="2"/>
  </xsd:sequence>
</xsd:complexType>
```

B.7.2.16 ProxyVideo element

The ProxyVideo element describes a Video element of this Proxy.

- Child Elements

Name	Definition
ProxyVideoCodec	Required. 1 sibling only.
ProxyVideoBitRate	Required. 1 sibling only.
ProxyFrameRate	Required. 1 sibling only.
ProxyResolution	Required. 1 sibling only.
AspectRatio	Optional. 0 or 1 sibling only.
OnScreenTimecode	Required. 1 sibling only.

- Schema

```
<xsd:complexType name="ProxyVideoType">
  <xsd:sequence>
    <xsd:element name="ProxyVideoCodec" type="ProxyVideoCodecType"/>
    <xsd:element name="ProxyVideoBitRate" type="ProxyVideoBitRateType"/>
    <xsd:element name="ProxyFrameRate" type="ProxyFrameRateType"/>
    <xsd:element name="ProxyResolution" type="ProxyResolutionType"/>
    <xsd:element name="AspectRatio" type="AspectRatioType" minOccurs="0"/>
    <xsd:element name="OnScreenTimecode" type="OnScreenTimecodeType"/>
  </xsd:sequence>
</xsd:complexType>
```

B.7.2.17 ProxyVideoCodec element

The ProxyVideoCodec element specifies the codec type of this Video element.

- Value

Value	Definition
MPEG-4	MPEG-4.

- Schema

```
<xsd:simpleType name="ProxyVideoCodecType">
  <xsd:restriction base="xsd:NMTOKEN">
    <xsd:enumeration value="MPEG-4"/>
  </xsd:restriction>
</xsd:simpleType>
```

B.7.2.18 ProxyVideoBitRate element

The ProxyVideoBitRate element specifies the bit rate (in bits/sec) of this Video element. This element is specified as xsd:positiveInteger and the range of the value shall be from 192000 to 2000000.

- Schema

```
<xsd:simpleType name="ProxyVideoBitRateType">
  <xsd:restriction base="xsd:positiveInteger">
    <xsd:minInclusive value="192000"/>
    <xsd:maxInclusive value="2000000"/>
  </xsd:restriction>
</xsd:simpleType>
```

B.7.2.19 ProxyFrameRate element

The ProxyFrameRate element specifies the frame rate of this Video element.

- Value

Value	Definition
29.97	29.97 frames/sec.
25	25 frames/sec.
29.97/2	29.97/2 frames/sec.
25/2	25/2 frames/sec.

- Schema

```
<xsd:simpleType name="ProxyFrameRateType">
  <xsd:restriction base="xsd:string">
    <xsd:enumeration value="29.97"/>
    <xsd:enumeration value="25"/>
    <xsd:enumeration value="29.97/2"/>
    <xsd:enumeration value="25/2"/>
  </xsd:restriction>
</xsd:simpleType>
```

B.7.2.20 ProxyResolution element

The ProxyResolution element specifies the resolution of this Video element.

- Value

Value	Definition
SIF_NTSC	352 x 240
SIF_PAL	352 x 288
QVGA	320 x 240

- Schema

```
<xsd:simpleType name="ProxyResolutionType">
  <xsd:restriction base="xsd:NMTOKEN">
    <xsd:enumeration value="SIF_NTSC"/>
    <xsd:enumeration value="SIF_PAL"/>
    <xsd:enumeration value="QVGA"/>
  </xsd:restriction>
</xsd:simpleType>
```

B.7.2.21 AspectRatio element

The AspectRatio element specifies the aspect ratio of the Proxy image that results in no geometric distortion (this may include black bars to give letterbox or pillarbox pictures). This element is optional. See Section 8.2.3 for detail.

B.7.2.22 OnScreenTimecode element

The OnScreenTimecode element specifies whether the timecode is overlayed on the screen.

- Value

Value	Definition
ON	Overlayed.
OFF	Not overlayed.

- Schema

```
<xsd:simpleType name="OnScreenTimecodeType">
  <xsd:restriction base="xsd:NMTOKEN">
    <xsd:enumeration value="ON"/>
    <xsd:enumeration value="OFF"/>
  </xsd:restriction>
</xsd:simpleType>
```

B.7.2.23 ProxyAudio element

The ProxyAudio element describes an Audio element of this Proxy. This element is optional.

- Child Elements

Name	Definition
ProxyAudioCodec	Required. 1 sibling only.
ProxyAudioBitRate	Required. 1 sibling only.
ProxySamplingRate	Required. 1 sibling only.
OriginalChannel	Required. 1 sibling only.

- Schema

```
<xsd:complexType name="ProxyAudioType">
  <xsd:sequence>
    <xsd:element name="ProxyAudioCodec" type="ProxyAudioCodecType"/>
    <xsd:element name="ProxyAudioBitRate" type="ProxyAudioBitRateType"/>
    <xsd:element name="ProxySamplingRate" type="ProxySamplingRateType"/>
    <xsd:element name="OriginalChannel" type="OriginalChannelType"/>
  </xsd:sequence>
</xsd:complexType>
```

B.7.2.24 ProxyAudioCodec element

The ProxyAudioCodec element specifies the codec type of this Audio element.

- Value

Value	Definition
AAC	Advanced Audio Codec

- Schema

```
<xsd:simpleType name="ProxyAudioCodecType">
  <xsd:restriction base="xsd:NMTOKEN">
    <xsd:enumeration value="AAC"/>
  </xsd:restriction>
</xsd:simpleType>
```

B.7.2.25 ProxyAudioBitRate element

The ProxyAudioBitrate element specifies the bit rate of this Audio element per channel.

- Value

Value	Definition
32000	32000 bits/sec.
64000	64000 bits/sec.

- Schema

```
<xsd:simpleType name="ProxyAudioBitRateType">
  <xsd:restriction base="xsd:positiveInteger">
    <xsd:enumeration value="32000"/>
    <xsd:enumeration value="64000"/>
  </xsd:restriction>
</xsd:simpleType>
```

B.7.2.26 ProxySamplingRate element

The ProxySamplingRate element specifies the sampling rate of this Audio element.

- Value

Value	Definition
24000	24000 samples/sec.

- Schema

```
<xsd:simpleType name="ProxySamplingRateType">
  <xsd:restriction base="xsd:positiveInteger">
    <xsd:enumeration value="24000"/>
  </xsd:restriction>
</xsd:simpleType>
```

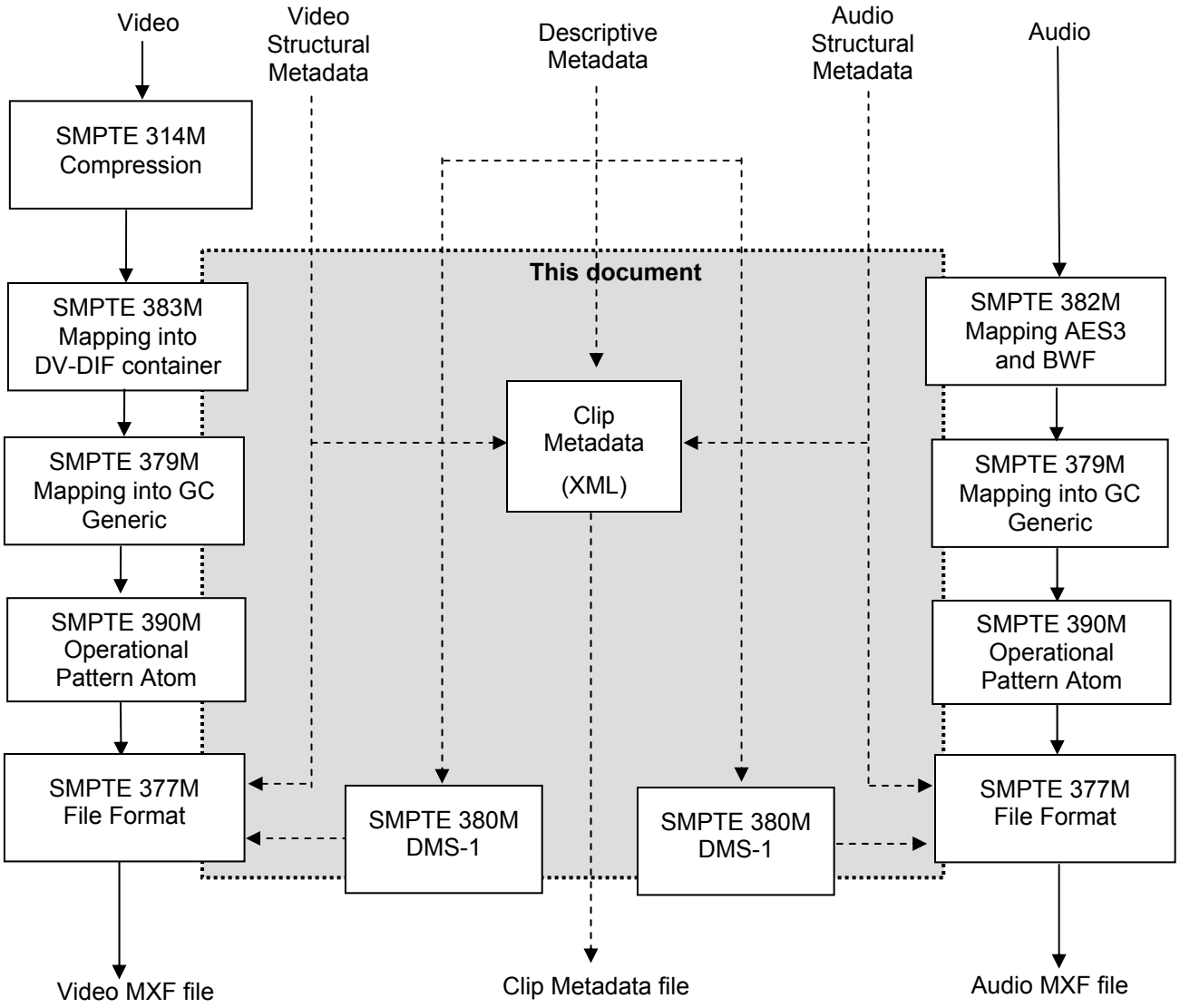
B.7.2.27 OriginalChannel element

The OriginalChannel element specifies the original channel number of this Audio element within the original Clip. This element is specified as xsd:NMTOKENS. When multiple channels within the original Clip are mixed down, all of the channel numbers must be specified in ascending order. For example, when Audio Essences of which the channel numbers are "0", "2" and "3" are mixed down to this Audio element, this OriginalChannel element should be specified as "0 2 3".

- Schema

```
<xsd:simpleType name="OriginalChannelType">
  <xsd:restriction base="xsd:NMTOKENS"/>
</xsd:simpleType>
```

Annex C (Informative)
Document flow



Annex D (Normative)
Specification of Bitmap file

D.1 Structure of a Bitmap file

The bitmap file consists of the BITMAPFILEHEADER structure, the BITMAPINFOHEADER and the bitmap image data as shown in table D.1.

Table D.1 – Structure of Bitmap file

BITMAPFILEHEADER structure
BITMAPINFOHEADER structure
Bitmap data

NOTE – The byte order of all data values shall be little endian (least significant byte first).

D.1.1 BITMAPFILEHEADER

The BITMAPFILEHEADER structure contains information about the type, size, and layout of a file.

```
typedef struct tag BITMAPFILEHEADER {
    WORD        bfType;
    DWORD       bfSize;
    WORD        bfReserved1;
    WORD        bfReserved2;
    DWORD       bfOffBits;
} BITMAPFILEHEADER, *PBITMAPFILEHEADER;
```

bfType

Specifies the file type. The file type shall be set to 424Dh (“BM”) in compliance with ISO/IEC 646 to indicate a bitmap file.

bfSize

Specifies the size, in bytes, of the bitmap file.

bfReserved1

Reserved; set to zero.

bfReserved2

Reserved; set to zero.

bfOffBits

Specifies the offset, in bytes, from the beginning of the BITMAPFILEHEADER structure to the bitmap bits.

D.1.2 BITMAPINFOHEADER

The BITMAPINFOHEADER structure contains information about the dimensions and color format.

```
typedef struct tagBITMAPINFOHEADER{

    DWORD       biSize;
    LONG        biWidth;
    LONG        biHeight;
    WORD        biPlanes;
    WORD        biBitCount;
    DWORD       biCompression;
    DWORD       biSizeImage;
    LONG        biXPelsPerMeter;
    LONG        biYPelsPerMeter;
    DWORD       biClrUsed;
    DWORD       biClrImportant;

} BITMAPINFOHEADER, *PBITMAPINFOHEADER;
```

biSize

Specifies the number of bytes required by the structure.

biWidth

Specifies the width of the bitmap, in pixels.

biHeight

Specifies the height of the bitmap, in pixels. If biHeight is positive, the bitmap is a bottom-up and its origin is the lower-left corner. If biHeight is negative, the bitmap is a top-down and its origin is the upper-left corner.

biPlanes

Specifies the number of planes for the target device. This value shall be set to 1.

biBitCount

Specifies the number of bits-per-pixel. The biBitCount member of the BITMAPINFOHEADER structure determines the number of bits that define each pixel and the maximum number of colors in the bitmap. This member is set to the following value.

Value	Meaning
24	The bitmap has a maximum of 2^24 colors, Each 3-byte triplet in the bitmap array represent the relative intensities of blue, green, and red, respectively, for a pixel.

biCompression

Specifies the type of compression for a compressed bottom-up bitmap. This member is set to the following value.

Value	Meaning
0 (BI_RGB)	An uncompressed format.

biSizeImage

Specifies the size, in bytes, of the image. This may be set to zero for BI_RGB bitmaps.

biXPelsPerMeter

Specifies the horizontal resolution, in pixels-per-meter, of the target device for the bitmap. An application uses this value to select a bitmap from a resource group that best matches the characteristics of the current device. This may be set to zero.

biYPelsPerMeter

Specifies the vertical resolution, in pixels-per-meter, of the target device for the bitmap. This may be set to zero.

biClrUsed

Specifies the number of color indexes in the color table that are actually used by the bitmap. If this value is zero, the bitmap uses the maximum number of colors corresponding to the value of the biBitCount member for the compression mode specified by biCompression.

biClrImportant

Specifies the number of color indexes that are required for displaying the bitmap. If this value is zero, all colors are required.

D.1.3 Bitmap image data

The bitmap image data is stored from the bottom line and is stored from the left within a line.

Annex E (Normative)
Specification of WAVE file

E.1 Structure of WAVE file

The WAVE file shall be compliant with the RIFF (Resource Interchange File format) structure as shown in Table E.1. The WAVE file shall start with the RIFF chunk ID followed by file size and data bytes. The first 4 data bytes in the RIFF chunk shall indicate the type of chunk and is set to “WAVE”. The WAVE file chunk shall contain the fmt chunk and the data chunk, each of which shall contain a chunk ID, data size and data bytes.

Table E.1 – Structure of WAVE file

Definition			Number of byte	Value
RIFF chunk	RIFF chunk ID		4	“RIFF” in ISO/IEC 646
	Total file size		4	File size - 8
	WAVE form type		4	“WAVE” in ISO/IEC 646
	fmt chunk	fmt chunk ID	4	“fmt ” in ISO/IEC 646
		fmt chunk size	4	16
		WAVEFORMATEX structure	16	
	data chunk	data chunk ID	4	“data” in ISO/IEC 646
		WAVE data size	4	
		WAVE data	var	

NOTE – Byte order of all data values shall be little endian (least significant byte first).

E.2 WAVE file chunk

E.2.1 WAVE Format Chunk

The WAVE Format chunk (fmt chunk) contains the information about the format of a WAVE file. The structure of the WAVE format extension (WAVEFORMATEX) in the fmt chunk shall be defined as follows:

```
typedef     struct waveformat_extended_tag{
    WORD        wFormatTag;         // format type
    WORD        nChannels;          // number of channels
    DWORD       nSamplesPerSec;     // sampling rate
    DWORD       nAvgBytesPerSec;    // for buffer estimation
    WORD        nBlockAlign;        // data block size
    DWORD       wBitsPerSample;     // number of bits per sample of mono data
    WORD        cbSize;             // the count in bytes of the extra size
} WAVEFORMATEX
```

wFormatTag

The number indicating the WAVE format. The value is set to 0001h to indicate PCM (uncompressed) data.

nChannels

The number of channels represented in the waveform data, such as 1 for mono or 2 for stereo.

nSamplesPerSec

The sampling rate (in samples per second) at which each channel should be played.

nAvgBytesPerSec

The average number of bytes per second at which the waveform data should be transferred. Playback software can estimate the buffer size using this value.

nBlockAlign

The block alignment (in bytes) of the waveform data. Playback software needs to process a multiple of <wBlockAlign> bytes of data at a time, so the value of <wBlockAlign> can be used for buffer alignment.

wBitsPerSample

This is the number of bits per sample per channel data. Each channel is assumed to have the same sample resolution. If this field is not needed, then the value should be set to zero.

cbSize

The size in bytes of the extra information in the WAVE format header not including the size of the WAVEFORMATEX structure.

E.2.2 Data chunk

The WAVE data is stored in the data chunk after the WAVE data size. In a single-channel WAVE file, samples shall be stored consecutively as shown below.

Sample-1		**Sample-2**	
Channel 0 low-order byte	Channel 0 high-order byte	Channel 0 low-order byte	Channel 0 high-order byte

Each sample shall be contained in an integer *i*. The size of *i* shall be the smallest number of bytes required to contain the specified sample size. The bits that represent the sample amplitude shall be stored in the most significant bits of *i*, and the remaining bits shall be set to zero.

For example, if the sample size (recorded in <nBitsPerSample>) is 12 bits, then each sample is stored in a two-byte integer. The least significant four bits of the first (least significant) byte is set to zero. The data format and maximum and minimum values for PCM waveform samples of various sizes are as follows:

Sample Size	Data Format	Maximum Value	Minimum Value
One to eight bits	Unsigned integer	255 (FFh)	0
Nine or more bits	Signed integer *i*	Largest positive value of *i*	Most negative value of *i*

Annex F (Informative)
Bibliography

SMPTE 380M-2004, Television — Material Exchange Format (MXF) — Descriptive Metadata Scheme-1

SMPTE RP 210, Metadata Dictionary Registry of Metadata Element Descriptions

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

SMPTE EG 42-2004, Material Exchange Format (MXF) — MXF Descriptive Metadata

“The File Formats Handbook", Gunter Born, Thomson, 1995

http://web.uccs.edu/wbahn/ECE1021/STATIC/REFERENCES/bmpfileformat.htm

http://www.microsoft.com/whdc/device/audio/multichaud.mspx