SMPTE REGISTERED DISCLOSURE DOCUMENT

Material Exchange Format — Carriage of ARRI Camera System Metadata

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

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

Every attempt has been made to ensure that the information contained in this document is accurate. Errors in this document should be reported to the proponent identified below, with a copy to eng@smpte.org.

This document is intended to support the development of applications that read, interpret and write ARRI camera system metadata.

All inquiries in respect of this document, including inquiries as to intellectual property requirements that may be attached to use of the disclosed technology, should be addressed to the proponent identified below.

Proponent contact information:

Joseph Goldstone
Arnold & Richter Cine Technik GmbH & Co. Betriebs KG
Herbert-Bayer-Straße 10
D-80807 München
Germany

Email: jgoldstone@arri.com

Approved May 5, 2022

Table of Contents Page

1 Introduction

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

ARRI camera systems produce metadata to document artistic, technical, and administrative aspects of created image essence, and to guide the later processing of that essence. The camera system can create audio essence in addition to image essence, in which case that audio essence is accompanied by audio-related metadata. The camera system can also integrate various user-provided static or dynamic metadata. Lastly, the metadata created by the camera system can be modified in post-production, e.g. to incorporate user-provided dynamic metadata not available at the time of the original recording, while maintaining the relationships between image essence and metadata or between various metadata that are established by this document.

2 Scope

This RDD specifies the carriage of ARRI camera system metadata by mechanisms of the MXF Constrained Generic Container described in SMPTE ST 379-2. These metadata describe picture essence originally created by ARRI cameras that could be encoded as ARRIRAW, as Apple ProRes or as some other encoding of picture essence, created audio essence (if audio essence is present), and additional metadata supplied by the user of the ARRI camera system during production or an application processing essence or metadata during post-production. The description encompasses the semantics of the metadata, their KLV coding, and their carriage using per-partition and per-frame mechanisms.

3 Conformance Notation

This RDD uses the conformance notation specified in SMPTE ST 377-1.

4 Normative References

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

SMPTE RDD 54:2022, Material Exchange Format — Mapping ARRIRAW Bitstreams into the MXF Generic Container

SMPTE ST 12-3:2016, Time Code for High Frame Rate Signals and Formatting in the Ancillary Data Space

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

SMPTE ST 330:2011, Unique Material Identifier (UMID)

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

SMPTE ST 377-1:2019, Material Exchange Format (MXF) — File Format Specification

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

SMPTE ST 385:2004, Material Exchange Format (MXF) — Mapping SDTI-CP Essence and Metadata into the MXF Generic Container

IEEE 754-2019, IEEE Standard for Floating-Point Arithmetic

IEEE 1588-2019, IEEE Standard for a Precision Clock Synchronization Protocol for Networked Measurement and Control Systems

ISO/TS 19127:2005, Geographic Information — Geodetic codes and parameters

ISO/IEC 21778:2017, Information technology — The JSON data interchange syntax

IETF STD RFC 7143, Internet Small Computer System Interface (iSCSI) Protocol (Consolidated)

IETF STD RFC 3986, Uniform Resource Identifier (URI): Generic Syntax

IETF STD RFC 7303, XML Media Types

5 Terms and Definitions

For the purposes of this document, the following terms and definitions apply.

5.1 ARRI MXF File

MXF file conforming to the provisions of this Registered Disclosure Document, containing ARRI camera system-originated Picture Essence and, optionally, audio and/or data essence, and associated metadata

5.2 ARRIRAW Essence

essence encoded as described in SMPTE RDD 54

5.3 Dynamic Metadata

metadata the values of which are not necessarily constant throughout the duration of the essence container

5.4 SingleFloat

single-precision floating-point number as defined by IEEE 754

Note 1 to entry: All instances of SingleFloat data in the Sets described in this document are coded as big-endian data, as described in SMPTE ST 377-1:2019, § 6.4.2 (“MXF Byte Order”).

5.5 Photosite

individual element of an image sensor’s photomosaic array

5.6 Reference Plane

plane perpendicular to the optical axis corresponding to where a sensor would be positioned for equivalent focus if there were no optical elements between the sensor and the lens

Note 1 to entry: In a typical digital cinema camera, the sensor will be covered by thin optical elements such as an optical low-pass filter (OLPF).

5.7 Sensor Coordinate System

cartesian coordinate system superimposed on the portion of the image sensor that registers incoming light, with coordinate system origin established at the upper-left corner of the uppermost, leftmost photosite, with x and y axes parallel to the sensor photosite grid and with unit distances along the x and y axes corresponding to increments of a single photosite’s width and height, respectively

5.8 Static Metadata

metadata the values of which are unchanging and that are applicable to all frames in the essence container

5.9 Target Color Space

set of RGB primary chromaticities, encoding white point chromaticity, and electro-optical transfer function (EOTF) that are characteristic of a display on which the captured imagery is evaluated

6 Metadata Carriage

6.1 Overview (informative)

ARRIRAW or Apple ProRes essence originally created by ARRI camera systems is stored in the Constrained Generic Container using frame-wrapping.

Static metadata is stored using two mechanisms. Metadata essential to image formation or interpretation are stored in SubDescriptors of the Image Descriptors. Other static metadata are stored in a static Descriptive Metadata Framework capable of carrying multiple instances of a concrete subclass of DescriptiveObject, with each instance containing a text payload. Items of such concrete subclasses indicate the encoding of the payload's text and the IANA-registered media type, along with (optionally) a URI of a schema for the payload. ARRI camera metadata is carried in UTF-16-encoded JSON documents, with schemae the URIs of which are given below.

Dynamic metadata are likewise stored using two mechanisms. The System Metadata Pack of the System Item of the Content Package carries the package's creation timestamp. Other dynamic metadata are carried in a Supplemental Data Essence Container, as instances of subclasses of StreamingData. This allows a separation of dynamic metadata by function and by data source (through weak reference from either a Data StreamSubDescriptor or a DataStreamDeviceSubDescriptor instance to an associated instance of a StreamingData subclass), and also provides for metadata that are created at rates that differ from the camera sensor rate.

A key point is that both static and dynamic metadata are carried independently of the picture essence encoding, and can be applied either to ARRIRAW- or Apple ProRes-encoded pictures, or to other picture essence using other appropriate encodings.

6.2 Static Metadata

Metadata essential to image formation or interpretation that are applicable to all the frames of the MXF file shall be carried using the Generic Picture Essence Descriptor defined in SMPTE ST 377-1, in conjunction with an instance of PictureEssenceProcessingSubDescriptor, an instance of TextureGen1ProcessingSubDescriptor or of TextureGen2ProcessingSubDescriptor, and an instance of ClipSubDescriptor. These SubDescriptors shall be strongly referenced through the SubDescriptors property of the GenericPictureEssenceDescriptor.

6.2.1 Generic Picture Essence Descriptor

The following items of the Generic Picture Essence Descriptor shall be present:

- Frame Layout
- Stored Height
- Stored Width
- Sampled Height
- Sampled Width
- SampledXOffset
- SampledYOffset
- DisplayWidth
- DisplayHeight
- DisplayXOffset
- DisplayYOffset
- Aspect Ratio
- Transfer Characteristic
- Image Start Offset
- Picture Essence Coding
- Color Primaries

When the picture essence described by the Generic Picture Essence Descriptor is ARRIRAW Essence, the Transfer Characteristic and Color Primaries items shall be interpreted as pertaining to the RGB data resulting from the demosaicing operation having been applied to said stored ARRIRAW Essence.

The Image End Offset item in the Generic Picture Essence Descriptor may be present.

The Frame Layout item in the Generic Picture Essence Descriptor shall have the value 0.

The following items of the Generic Picture Essence Descriptor are not used to represent ARRI picture essence and shall not be present in an ARRI MXF File:

- StoredF2Offset
- DisplayF2Offset
- Active Format Descriptor
- Alpha Transparency
- Image Alignment Offset
- FieldDominance
- Coding Equations

NOTE Though for ARRIRAW Essence there is no associated codec *per se* as the content is uncompressed, Sampled Width, Sampled Height, SampledXOffset and SampledYOffset can have values other than their defaults. Such values can be used to allow applications creating files containing ARRIRAW Essence or Apple ProRes Essence flexibility in padding the rectangle of photosite or pixel data on one or more sides.

6.2.2 Picture Essence Processing

Metadata concerning processing of the picture essence in ways not specific to either ARRIRAW Essence or Apple ProRes Essence shall be carried by an instance of Picture Essence Processing SubDescriptor, a subclass of SubDescriptor.

6.2.2.1 Image Processing Model ULs

The Image Processing Model denotes the image processing pipeline used to process ARRI Picture Essence into scene colors and display colors. ULs representing valid values for such processing shall be as specified in Table 1. Definitions of the image processing models indicated by the ULs in Table 1 are outside the scope of this document.

Table 1 — Image Processing Model ULs

Item Name	Symbol	Kind	Item UL
Image Processing Generation 1	`ImageProcessingGen1`	LEAF	`060e2b34.0401010d.0e170102.01010000`
Image Processing Generation 2	`ImageProcessingGen2`	LEAF	`060e2b34.0401010d.0e170102.01020000`
Image Processing Generation 3	`ImageProcessingGen3`	LEAF	`060e2b34.0401010d.0e170102.01030000`

6.2.2.2 ARRI Colorimetric Data Set ULs

The colorimetric data set used in processing the photosite data shall be identified using one of the ULs as specified in Table 2. Definitions of the colorimetric data sets indicated by the ULs in Table 2 are outside the scope of this document.

Table 2 — ARRI Colorimetric Data Set ULs

Item Name	Symbol	Kind	Item UL
Colorimetric Data Set A3RevB	ColorimetricDataSetA3RevB	LEAF	060e2b34.0401010d.0e170102.02010000
Colorimetric Data Set A3RevB-FSND	ColorimetricDataSetA3RevBFSND	LEAF	060e2b34.0401010d.0e170102.02020000
Colorimetric Data Set A3RevC	ColorimetricDataSetA3RevC	LEAF	060e2b34.0401010d.0e170102.02030000
Colorimetric Data Set A3RevC-FSND	ColorimetricDataSetA3RevCFSND	LEAF	060e2b34.0401010d.0e170102.02040000
Colorimetric Data Set A2X	ColorimetricDataSetA2X	LEAF	060e2b34.0401010d.0e170102.02050000
Colorimetric Data Set A2X-FSND	ColorimetricDataSetA2XFSND	LEAF	060e2b34.0401010d.0e170102.02060000
Colorimetric Data Set A3X	ColorimetricDataSetA3X	LEAF	060e2b34.0401010d.0e170102.02070000
Colorimetric Data Set A4	ColorimetricDataSetA4	LEAF	060e2b34.0401010d.0e170102.02080000
Colorimetric Data Set A4-FSND	ColorimetricDataSetA4FSND	LEAF	060e2b34.0401010d.0e170102.02090000

6.2.2.3 Picture Essence Processing SubDescriptor ULs

The ULs of the Picture Essence Processing SubDescriptor Key and of the Picture Essence Processing SubDescriptor items shall be as specified in Table 3.

Table 3 — Picture Essence Processing ULs

Item Name	Symbol	Kind	Item UL
Picture Essence Processing SubDescriptor	PictureEssenceProcessingSubDescriptor	LEAF	060e2b34.027f0101.0e170102.01000000
Image Processing Model	ImageProcessingModel	LEAF	060e2b34.0101010e.0e170102.01010000
Colorimetric Data Set Identifier	ColorimetricDataSetIdentifier	LEAF	060e2b34.0101010e.0e170102.01020000
Flip	Flip	LEAF	060e2b34.0101010e.0e170102.01030000
Flop	Flop	LEAF	060e2b34.0101010e.0e170102.01040000

6.2.2.4 Picture Essence Processing SubDescriptor

The Picture Essence Processing SubDescriptor shall be a subclass of SubDescriptor and shall be as specified in Table 4. All items marked as required shall be present. Any item marked as optional may be present. The Picture Essence Processing SubDescriptor may include additional items, with any such items being outside the scope of this document.

Table 4 — Picture Essence Processing SubDescriptor

Symbol	Type	Len	Req ?	Meaning	Default
PictureEssenceProcessingSubDescriptor	Set Key	16	Req	Defines image processing parameters for all ARRI camera system images.	
Length	BER Length	4	Req	Set Length.	
All items in ST 377-1:2019 B.3 (SubDescriptor) except the Key or Group UL and the Length, if present					
ImageProcessingModel	AUID	16	Req	Image processing model. The value shall be one of the ULs specified in Table 1.	
ColorimetricDataSetIdentifier	AUID	16	Req	Colorimetric Data Set Identifier. The value shall be one of the ULs specified in Table 2.	
Flip	UInt8	1	Opt	Flags for mirroring across horizontal centerline. Bits 7-1: Reserved, shall be 0 Bit 0: Is Mirrored	0x00
Flop	UInt8	1	Opt	Flags for mirroring across vertical centerline. Bits 7-1: Reserved, shall be 0 Bit 0: Is Mirrored	0x00

For the Flip and Flop items, if the item is present, and the Is Mirrored flag is set, and the application wishes to view or save content such that the viewed or saved image orientation represents creative intent, then the image shall be mirrored across the horizontal or vertical centerlines, respectively.

6.2.3 Texture Processing

Texture Processing SubDescriptor is an abstract class with two concrete subclasses, Texture Gen1 Processing SubDescriptor and Texture Gen2 Processing SubDescriptor, an instance one and only one of which shall be present in any single ARRI MXF File. Definitions of the image processing algorithms the parameters of which are carried in Texture Processing SubDescriptor or its subclasses are outside the scope of this document.

6.2.3.1 Texture Processing SubDescriptor

6.2.3.1.1 Texture Processing SubDescriptor ULs

The ULs of the Texture Processing SubDescriptor Key and of Texture Processing SubDescriptor items shall be as specified in Table 5.

Table 5 — Texture Processing SubDescriptor ULs

Item Name	Symbol	Kind	Item UL
Texture Processing SubDescriptor	`TextureProcessingSubDescriptor`	LEAF	`060e2b34.027f0101.0e170102.02000000`
Texture Applied	`TextureApplied`	LEAF	`060e2b34.0101010e.0e170102.02010000`
Temporal Processing Applied	`TemporalProcessingApplied`	LEAF	`060e2b34.0101010e.0e170102.02020000`
Burst Noise Reduction Applied	`BurstNoiseReductionApplied`	LEAF	`060e2b34.0101010e.0e170102.02030000`

6.2.3.1.2 Texture Processing SubDescriptor

The Texture Processing SubDescriptor shall be as specified in Table 6.

Table 6 — Texture Processing SubDescriptor

Symbol	Type	Len	Req ?	Meaning	Default
`TextureProcessingSubDescriptor`	Set Key	16	Req	Defines texture processing information applicable to all generations of ARRI texture processing.	
`Length`	BER Length	4	Req	Set Length.	
All items in ST 377-1:2019 B.3 (SubDescriptor) except the Key or Group UL and the Length, if present					
`TextureApplied`	Boolean	1	Req	If texture has been applied to produce the output, the Boolean value shall be TRUE, otherwise (texture has not been applied to produce the output) the Boolean value shall be FALSE.	
`TemporalProcessingApplied`	Boolean	1	Req	If temporal processing has been applied to produce the output, the Boolean value shall be TRUE, otherwise (temporal processing has not been applied to produce the output) the Boolean value shall be FALSE.	
`BurstNoiseReductionApplied`	Boolean	1	Req	If Burst Noise Reduction has been applied to produce the output, the Boolean value shall be TRUE, otherwise (Burst Noise Reduction has not been applied to produce the output) the Boolean value shall be FALSE.	

6.2.3.2 Texture Gen1 Processing SubDescriptor

Texture Gen1 Processing SubDescriptor shall be a subclass of Texture Processing SubDescriptor.

6.2.3.2.1 Texture Gen1 Processing SubDescriptor ULs

The ULs of the Texture Gen1 Processing SubDescriptor Key and of Texture Gen1 Processing SubDescriptor items shall be as specified in Table 7.

Table 7 — Texture Gen1 Processing SubDescriptor ULs

Item Name	Symbol	Kind	Item UL
Texture Gen1 Processing SubDescriptor	TextureGen1ProcessingSubDescriptor	LEAF	060e2b34.027f0101.0e170102.03000000
Texture Noise Reduction Mode	TextureNoiseReductionMode	LEAF	060e2b34.0101010e.0e170102.03010000
Texture Image Sharpness	TextureImageSharpness	LEAF	060e2b34.0101010e.0e170102.03020000
Texture Image Detail	TextureImageDetail	LEAF	060e2b34.0101010e.0e170102.03030000
Texture Noise Reduction Strength	TextureNoiseReductionStrength	LEAF	060e2b34.0101010e.0e170102.03040000

6.2.3.2.2 Texture Gen1 Processing SubDescriptor

The Texture Gen1 Processing SubDescriptor shall be as specified in Table 8.

Table 8 — Texture Gen1 Processing SubDescriptor

Symbol	Type	Len	Req ?	Meaning	Default
TextureGen1ProcessingSubDescriptor	Set Key	16	Req	Defines texture processing information applicable to 1st-generation ARRI texture processing.	
Length	BER Length	4	Req	Set Length.	
All items in TextureProcessingSubDescriptor except the Key or Group UL and the Length, if present					
TextureNoiseReductionMode	UInt8	1	Req	Type of the noise reduction. 0x00: None 0x01: ANR-1	
TextureImageSharpness	SingleFloat	4	Req	Amount S of image sharpness, with -5 ≤ S ≤ +35.	
TextureImageDetail	SingleFloat	4	Req	Amount D of image detail, with -5 ≤ D ≤ +5.	
TextureNoiseReductionStrength	SingleFloat	4	Opt	Strength S of the noise reduction, with 1.0 ≤ S ≤ 3.5. This element shall be present if and only if the value of Texture Noise Reduction Mode is ANR-1.	

6.2.3.3 Texture Gen2 Processing SubDescriptor

The Texture Gen2 Processing SubDescriptor shall be a subclass of Texture Processing SubDescriptor.

6.2.3.3.1 Texture Gen2 Processing SubDescriptor ULs

The ULs of the Texture Gen2 Processing SubDescriptor Key and of the Texture Gen2 Processing SubDescriptor items shall be as specified in Table 9.

Table 9 — Texture Gen2 Processing SubDescriptor ULs

Item Name	Symbol	Kind	Item UL
Texture Gen2 Processing SubDescriptor	TextureGen2ProcessingSubDescriptor	LEAF	060e2b34.027f0101.0e170102.04000000
Texture Data	TextureData	LEAF	060e2b34.0101010e.0e170102.04010000
Texture File Name	TextureFileName	LEAF	060e2b34.0101010e.0e170102.04020000

6.2.3.3.2 Texture Gen2 Processing SubDescriptor

The Texture Gen2 Processing SubDescriptor shall be as specified in Table 10.

Table 10 — Texture Gen2 Processing SubDescriptor

Symbol	Type	Len	Req ?	Meaning	Default
TextureGen2ProcessingSubDescriptor	Set Key	16	Req	Defines texture processing information applicable to 2nd-generation ARRI texture processing.	
Length	BER Length	4	Req	Set Length.	
All items in TextureProcessingSubDescriptor except the Key or Group UL and the Length, if present					
TextureData	DataValue	var	Req	Texture Data.	
TextureFileName	UTF16String	var	Opt	Name of the Texture File.	

6.2.4 Clip Information

Static metadata essential to image formation or interpretation are carried by an instance of ClipSubDescriptor, a subclass of SubDescriptor. The ULs for and descriptions of the components of the Clip SubDescriptor and the Sets it strongly references shall be as described below.

6.2.4.1 Sensor Dimensions

6.2.4.1.1 Background (Informative)

The physical size of the light-sensitive portion of the sensor is an essential datum for many algorithms that reconstruct the camera's position and orientation relative to the captured scene.

6.2.4.1.2 Sensor Dimensions ULs

The ULs of the Sensor Dimensions Set Key and of the Sensor Dimensions items shall be as specified in Table 11.

Table 11 — Sensor Dimension ULs

Item Name	Symbol	Kind	Item UL
Sensor Dimensions	SensorDimensions	LEAF	060e2b34.027f0101.0e170102.05000000
Sensor Width	SensorWidth	LEAF	060e2b34.0101010e.0e170102.05010000
Sensor Height	SensorHeight	LEAF	060e2b34.0101010e.0e170102.05020000

6.2.4.1.3 Sensor Dimensions Set

The Sensor Dimensions Set shall be a subclass of Interchange Object, and shall be as specified in Table 12.

Table 12 — Sensor Dimensions Set

Symbol	Type	Len	Req ?	Meaning	Default
SensorDimensions	Set Key	16	Req	Dimensions of light-sensitive area of sensor, independent from the subset of that region from which image data are obtained.	
Length	BER Length	4	Req	Set Length.	
All items in ST 377-1:2019 A.1 (InterchangeObject) except the Key or Group UL and the Length, if present					
SensorWidth	SingleFloat	4	Req	Width of light-sensitive area of sensor, in millimeters.	
SensorHeight	SingleFloat	4	Req	Height of light-sensitive area of sensor, in millimeters.	

6.2.4.2 Rectangles

Axis-aligned rectangular areas on the sensor's photosite grid shall be expressed in the sensor coordinate system using a Rectangle Set.

6.2.4.2.1 Rectangle Set ULs

The ULs of the Rectangle Set Key and of the Rectangle Set items shall be as specified in Table 13.

Table 13 — Rectangle Set ULs

Item Name	Symbol	Kind	Item UL
Rectangle	Rectangle	LEAF	060e2b34.027f0101.0e170102.06000000
Rectangle Top Offset	RectangleTopOffset	LEAF	060e2b34.0101010e.0e170102.06010000
Rectangle Left Offset	RectangleLeftOffset	LEAF	060e2b34.0101010e.0e170102.06020000
Rectangle Width	RectangleWidth	LEAF	060e2b34.0101010e.0e170102.06030000
Rectangle Height	RectangleHeight	LEAF	060e2b34.0101010e.0e170102.06040000

6.2.4.2.2 Rectangle Set

The Rectangle Set shall be a subclass of Interchange Object, and shall be as specified in Table 14.

Table 14 — Rectangle Set

Symbol	Type	Len	Req ?	Meaning	Default
Rectangle	Set Key	16	Req	Defines a Rectangle.	
Length	BER Length	4	Req	Set Length.	
All items in ST 377-1:2019 A.1 (InterchangeObject) except the Key or Group UL and the Length, if present					
RectangleTopOffset	UInt16	2	Req	Vertical offset from coordinate origin.	
RectangleLeftOffset	UInt16	2	Req	Horizontal offset from coordinate origin.	
RectangleWidth	UInt16	2	Req	Rectangle width.	
RectangleHeight	UInt16	2	Req	Rectangle height.	

6.2.4.3 Sensor Characteristics

Metadata containing characteristics of the sensor, the values of which are unchanging throughout the duration of the clip, shall be carried by a Sensor Characteristics Set. Definitions of the internal structure of user pixel mask data or its use in image formation are outside the scope of this document.

6.2.4.3.1 Sensor Characteristics Set ULs

The ULs of the Sensor Characteristics Set shall be as specified in Table 15.

Table 15 — Sensor Characteristics Set ULs

Item Name	Symbol	Kind	Item UL
Sensor Characteristics	SensorCharacteristics	LEAF	060e2b34.027f0101.0e170102.07000000
Sensor Name	SensorName	LEAF	060e2b34.0101010e.0e170102.07010000
Sensor Pixel Pitch	SensorPixelPitch	LEAF	060e2b34.0101010e.0e170102.07020000
Overall Sensor Dimensions	OverallSensorDimensions	LEAF	060e2b34.0101010e.0e170102.07030000
Sensor Acquisition Rectangle	SensorAcquisitionRectangle	LEAF	060e2b34.0101010e.0e170102.07040000
Sensor User Pixel Mask Active	SensorUserPixelMaskActive	LEAF	060e2b34.0101010e.0e170102.07050000
Sensor User Pixel Mask Data	SensorUserPixelMaskData	LEAF	060e2b34.0101010e.0e170102.07060000

6.2.4.3.2 Sensor Characteristics Set

The Sensor Characteristics Set shall be a subclass of Interchange Object and shall be specified as in Table 16.

Table 16 — Sensor Characteristics Set

Symbol	Type	Len	Req ?	Meaning	Default
SensorCharacteristics	Set Key	16	Req	Defines a set of sensor-related metadata.	
Length	BER Length	4	Req	Set Length.	
All items in ST 377-1:2019 A.1 (InterchangeObject) except the Key or Group UL and the Length, if present					
SensorName	UTF16String	var	Req	Name of the imaging sensor.	
SensorPixelPitch	SingleFloat	4	Req	Distance between centers of sensor photosites, in µm.	
OverallSensorDimensions	StrongRef (SensorDimensions)	16	Req	Strong reference to Sensor Dimensions instance describing the light-sensitive area of the sensor, in mm, independent of the subset of that region from which image data are obtained.	
SensorAcquisitionRectangle	StrongRef (Rectangle)	16	Req	Strong reference to Rectangle instance representing the rectangular region of the sensor containing the photosites from which the captured image is read.	

Symbol	Type	Len	Req ?	Meaning	Default
SensorUserPixelMaskActive	Boolean	1	Req	If User Pixel Mask Data are being used to modify the captured photosite data, the Boolean value shall be TRUE, otherwise (User Pixel Mask Data are not being used to modify the captured photosite data), the Boolean value shall be FALSE. If the value of this item is TRUE, then the Sensor User Pixel Mask Data item of this Set shall be present.	
SensorUserPixelMaskData	DataValue	Var	Opt	Opaque array of data containing information on location of user-flagged pixels where one or more photosites contributing to that pixel occasionally or constantly produce incorrect data. This item shall be present if and only if the value of the Sensor User Pixel Masking item is TRUE.	

6.2.4.4 Color Encodings

6.2.4.4.1 Background (Informative)

Image data are represented as RGB triplets in many parts of the color processing pipeline. The Color Encoding Set uses values from two tables of ULs, or from the SMPTE Labels Register defined in SMPTE ST 400, to indicate a particular combination of primaries and adopted white point, and a transfer function.

6.2.4.4.2 Color Encoding RGB Primaries and Adopted White ULs

Color encodings define combinations of RGB primaries and adopted white point. ULs representing valid values for two such combinations shall be as specified in Table 17. Definitions of the chromaticities of the primaries and white points indicated by the ULs in Table 17 are outside the scope of this document.

Table 17 — Color Encoding RGB Primaries and Adopted White ULs

Item Name	Symbol	Kind	Item UL
ARRI Wide Gamut 3 and D65 White	AWG3D65	LEAF	060e2b34.0401010d.0e170102.03010000
ARRI Wide Gamut 4 and D65 White	AWG4D65	LEAF	060e2b34.0401010d.0e170102.03020000

6.2.4.4.3 Color Encoding Transfer Function ULs

ULs representing valid values for two transfer functions converting a linear signal to a near-logarithmic signal shall be as specified in Table 18. Definition of the transfer functions indicated by the ULs in Table 18 are outside the scope of this document.

Table 18 — Color Encoding Transfer Function ULs

Item Name	Symbol	Kind	Item UL
ARRI LogC3	LOGC3	LEAF	060e2b34.0401010d.0e170102.04010000
ARRI LogC4	LOGC4	LEAF	060e2b34.0401010d.0e170102.04020000

6.2.4.4.4 Color Encoding Set ULs

The ULs of the Color Encoding Set Key and of the items of the Color Encoding Set shall be as specified in Table 19.

Table 19 — Color Encoding Set ULs

Item Name	Symbol	Kind	Item UL
Color Encoding	ColorEncoding	LEAF	060e2b34.027f0101.0e170102.08000000
Color Encoding Primaries And White Point	ColorEncodingPrimariesAndWhitePoint	LEAF	060e2b34.0101010e.0e170102.08010000
Color Encoding Transfer Function	ColorEncodingTransferFunction	LEAF	060e2b34.0101010e.0e170102.08020000

6.2.4.4.5 Color Encoding Set

The Color Encoding Set shall be a subclass of Interchange Object, and shall be as specified in Table 20.

Table 20 — Color Encoding Set

Symbol	Type	Len	Req ?	Meaning	Default
ColorEncoding	Set Key	16	Req	Defines a Color Encoding.	
Length	BER Length	4	Req	Set Length.	
All items in ST 377-1:2019 A.1 (InterchangeObject) except the Key or Group UL and the Length, if present					
ColorEncodingPrimariesAndWhitePoint	AUID	16	Req	Primaries and white point of a color encoding.	
ColorEncodingTransferFunction	AUID	16	Req	Transfer function of a color encoding.	

6.2.4.5 3D LUTs

6.2.4.5.1 Background (Informative)

3D LUTs modify the camera's mapping from logarithmically encoded RGB data to displayable RGB. The mapping from this logarithmically encoded RGB to displayable RGB is termed the Look.

6.2.4.5.2 3D LUT Data Layout ULs

3D LUT Sets shall contain an item indicating the layout of the data represented by the 3D LUT Data item and the value of that indicative item shall be one of the ULs specified in Table 21.

Table 21 — 3D LUT Data Layout ULs

Item Name	Symbol	Kind	Item UL
Interleaved 16-Bit Little-Endian RGB Blue Changes Fastest	Interleaved16BitLittleEndianRGB BlueChangesFastest	LEAF	060e2b34.0401010d.0e170102.05010000

6.2.4.5.3 3D LUT Set ULs

The ULs of the 3D LUT Set Key and of the items of the 3D LUT Set shall be as specified in Table 22.

Table 22 — 3D LUT Set ULs

Item Name	Symbol	Kind	Item UL
3D LUT	LUT3D	LEAF	060e2b34.027f0101.0e170102.09000000
3D LUT Scaling Factor	LUT3DScalingFactor	LEAF	060e2b34.0101010e.0e170102.09010000
3D LUT Normalization Gain	LUT3DNormalizationGain	LEAF	060e2b34.0101010e.0e170102.09020000
3D LUT Normalization Offset	LUT3DNormalizationOffset	LEAF	060e2b34.0101010e.0e170102.09030000
3D LUT Mesh Points Per Channel	LUT3DMeshPointsPerChannel	LEAF	060e2b34.0101010e.0e170102.09040000
3D LUT Data	LUT3DData	LEAF	060e2b34.0101010e.0e170102.09050000
3D LUT Data Layout	LUT3DDataLayout	LEAF	060e2b34.0101010e.0e170102.09060000
3D LUT Source Color Encoding	LUT3DSourceColorEncoding	LEAF	060e2b34.0101010e.0e170102.09070000
3D LUT Target Color Encoding	LUT3DTargetColorEncoding	LEAF	060e2b34.0101010e.0e170102.09080000
3D LUT Identifier	LUT3DIdentifier	LEAF	060e2b34.0101010e.0e170102.09090000

6.2.4.5.4 3D LUT Set

The 3D LUT Set shall be a subclass of Interchange Object, and shall be as specified in Table 23.

Table 23 — 3D LUT Set

Symbol	Type	Len	Req ?	Meaning	Default
LUT3D	Set Key	16	Req	Defines an ARRI 3D LUT.	
Length	BER Length	4	Req	Set Length.	
All items in ST 377-1:2019 A.1 (InterchangeObject) except the Key or Group UL and the Length, if present					
LUT3DScalingFactor	UInt32	4	Req	Factor that was applied to produce values stored as the 3D LUT data.	
LUT3DNormalizationGain	SingleFloat	4	Req	Gain added to output of 3D LUT lookup.	
LUT3DNormalizationOffset	SingleFloat	4	Req	Offset added to output of 3D LUT lookup.	
LUT3DMeshPointsPerChannel	UInt16	2	Req	Number of mesh points per channel.	
LUT3DData	DataValue	$2*n^3$	Req	Contents of the 3D LUT, with data layout given by the 3D LUT Data Layout Element. The variable n in the size calculation is the value of the 3D LUT Mesh Points Per Channel Item.	
LUT3DDataLayout	AUID	16	Req	Layout of the 3D LUT. The value shall be one of the ULs specified in Table 21.	
LUT3DSourceColorEncoding	StrongRef (ColorEncoding)	16	Req	Color encoding of data that are input to 3D LUT.	
LUT3DTargetColorEncoding	StrongRef (ColorEncoding)	16	Req	Color encoding of data produced by 3D LUT.	
LUT3DIdentifier	UTF16String	var	Opt	Identifier of this 3D LUT.	

6.2.4.6 Color Processing

Metadata concerning the scene color encoding of the clip, the application of a white balance algorithm, the embedded 3D LUT(s) and their application shall be carried by a Color Processing Set. Definition of the color processing the version of which is carried in the Video Rendering Version of the Color Processing Set is outside the scope of this document.

6.2.4.6.1 Color Processing Set ULs

The ULs of the Color Processing Set shall be as specified in Table 24.

Table 24 — Color Processing Set ULs

Item Name	Symbol	Kind	Item UL
Color Processing	ColorProcessing	LEAF	060e2b34.027f0101.0e170102.0a000000
White Balance Applied	WhiteBalanceApplied	LEAF	060e2b34.0101010e.0e170102.0a010000
Scene Color Encoding	SceneColorEncoding	LEAF	060e2b34.0101010e.0e170102.0a020000
Color Look Applied	ColorLookApplied	LEAF	060e2b34.0101010e.0e170102.0a030000
Look Modified	LookModified	LEAF	060e2b34.0101010e.0e170102.0a050000
CDL Slope	CDLSlope	LEAF	060e2b34.0101010e.0e170102.0a060000
CDL Offset	CDLOffset	LEAF	060e2b34.0101010e.0e170102.0a070000
CDL Power	CDLPower	LEAF	060e2b34.0101010e.0e170102.0a080000
CDL Saturation	CDLSaturation	LEAF	060e2b34.0101010e.0e170102.0a090000
3D LUTs	LUT3Ds	LEAF	060e2b34.0101010e.0e170102.0a040000
Look Intensity	LookIntensity	LEAF	060e2b34.0101010e.0e170102.0a0a0000
Look File Name	LookFileName	LEAF	060e2b34.0101010e.0e170102.0a0b0000
Look Note	LookNote	LEAF	060e2b34.0101010e.0e170102.0a0c0000
Video Rendering Version	VideoRenderingVersion	LEAF	060e2b34.0101010e.0e170102.0a0d0000

6.2.4.6.2 Color Processing Set

The Color Processing Set shall be a subclass of Interchange Object and shall be specified as in Table 25.

Table 25 — Color Processing Set

Symbol	Type	Len	Req ?	Meaning	Default
ColorProcessing	Set Key	16	Req	Defines a set of metadata used in color processing.	
Length	BER Length	4	Req	Set Length.	
All items in ST 377-1:2019 A.1 (InterchangeObject) except the Key or Group UL and the Length, if present					
WhiteBalanceApplied	Boolean	1	Req	If White Balancing has been applied to produce the output, the Boolean value shall be TRUE, otherwise (White Balancing has not been applied) the Boolean value shall be FALSE.	
SceneColorEncoding	StrongRef (ColorEncoding)	16	Req	Encoding of the input to the color processing model, for ASC CDL and 3D LUT application.	

Symbol	Type	Len	Req ?	Meaning	Default
ColorLookApplied	Boolean	1	Req	If the original image has been modified by the application of ASC CDL and/or a 3D LUT, the Boolean value shall be TRUE, otherwise (no ASC CDL has been applied, nor any 3D LUT) the Boolean value shall be FALSE.	
LookModified	Boolean	1	Req	If the Look has been modified since being loaded from a Look File, the Boolean value shall be TRUE, otherwise (the Look has not been modified since being loaded from a Look File) the Boolean value shall be FALSE.	
CDLSlope	Array of SingleFloat	8+4 *3	Req	Triad of ASC CDL slope values, for red, green and blue channels, respectively.	
CDLOffset	Array of SingleFloat	8+4 *3	Req	Triad of ASC CDL offset values, for red, green and blue channels, respectively.	
CDLPower	Array of SingleFloat	8+4 *3	Req	Triad of ASC CDL power values, for red, green and blue channels, respectively.	
CDLSaturation	SingleFloat	4	Req	ASC CDL saturation value.	
LUT3Ds	Batch of StrongRef (LUT3D)	8+1 6*n	Opt	Batch of Strong References to LUT3D objects.	
LookIntensity	UInt32	4	Opt	Look intensity in percent as adjusted in-camera, in the closed interval [0, 100]. When the Look intensity is 0, the Look is not applied; when the Look intensity is 100, the Look is fully applied.	100
LookFileName	UTF16String	var	Opt	Name of Look File from which was loaded the Look active when clip was captured.	
LookNote	UTF16String	var	Opt	User-supplied note as stored in Look File.	
VideoRenderingVersion	UTF16String	var	Opt	Identifier specifying the version of the video rendering transform used by the camera.	

6.2.4.7 Clip SubDescriptor

Metadata essential to image formation or interpretation shall be carried by a Clip SubDescriptor. This SubDescriptor shall be strongly referenced through the SubDescriptors property of the Generic Picture Essence Descriptor.

6.2.4.7.1 Clip SubDescriptor ULs

The ULs of the Clip SubDescriptor Set Key and of the items of the Clip SubDescriptor shall be as specified in Table 26.

Table 26 — Clip SubDescriptor ULs

Item Name	Symbol	Kind	Item UL
Clip SubDescriptor	`ClipSubDescriptor`	LEAF	`060e2b34.027f0101.0e170102.0b000000`
Image Sensor Characteristics	`ImageSensorCharacteristics`	LEAF	`060e2b34.0101010e.0e170102.0b010000`
Clip Color Processing	`ClipColorProcessing`	LEAF	`060e2b34.0101010e.0e170102.0b020000`
Image Checksum Type	`ImageChecksumType`	LEAF	`060e2b34.0101010e.0e170102.0b030000`

6.2.4.7.2 Clip SubDescriptor

The ARRI Clip Information SubDescriptor shall be as specified in Table 27. All items marked as required shall be present. Any item marked as optional may be present.

Table 27 — Clip SubDescriptor

Symbol	Type	Len	Req ?	Meaning	Default
`ClipSubDescriptor`	Set Key	16	Req	Defines a Clip SubDescriptor.	
`Length`	BER Length	4	Req	Set Length.	
All items in ST 377-1:2019 B.3 (SubDescriptor) except the Key or Group UL and the Length, if present					
`ImageSensorCharacteristics`	StrongRef (SensorCharacteristics)	16	Req	Strong Reference to Sensor Characteristics Object describing static characteristics of the sensor.	
`ClipColorProcessing`	StrongRef (ColorProcessing)	16	Req	Strong Reference to Color Processing Object describing processing operations applied to image data in that are unvarying throughout the duration of the clip.	
`ImageChecksumType`	UInt8	1	Opt	Type of Image Checksum. 0x00: None 0x01: The CRC32C algorithm of IETF RFC 7143 § 13.1. The image checksum value element of the ARRI Camera System Frame-Based Metadata shall be present if and only if the image checksum type element is present.	

The 3DLUTs property of the Clip SubDescriptor shall contain at least one strong reference to a 3DLUT Set.

6.3 Dynamic Metadata

6.3.1 System Item

6.3.1.1 System Metadata Pack

The System Metadata Pack shall be compliant with SMPTE ST 326 and shall be mapped into the Constrained Generic Container as specified in SMPTE ST 385.

6.3.1.2 Package Metadata Set

The Package Metadata Set shall be compliant with SMPTE ST 326 and shall be mapped into the Constrained Generic Container as specified in SMPTE ST 385.

The Package Metadata Set shall contain a SMPTE UMID Metadata item as specified in SMPTE ST 331.

The value of this item should be a basic UMID. This basic UMID shall be equal to the Package UID of the Top-level File Package.

If the value of this item is an extended UMID, its basic UMID part shall correspond to the Package UID of the Top-Level File Package.

NOTE According to SMPTE ST 330, in an extended UMID the length byte within the basic UMID part assumes the value of 0x33.

The Package Metadata Set may include other metadata blocks. The specification of any such metadata blocks is outside the scope of this document.

6.3.2 Camera System State

Camera system state shall be carried in a Supplemental Data element essence container as specified in Annex A.

Each Content Package in the frame-wrapped camera system state shall contain at least one instance of the Camera Position And Orientation Set, one instance of the Lens Configuration Set and one instance of the Frame Capture Set.

Other Sets may be present in the Supplemental Data element essence, but the description of any such Sets is outside the scope of this document, save to specify that any such Set shall be a concrete subclass of StreamingData.

The Camera Position And Orientation Set, the Lens Configuration Set or the Frame Capture Set may each appear one or more times within any one Content Package in the MXF file. Other subclasses of StreamingData may appear zero or more times within any one Content Package in the MXF file.

If values of items in the Camera Position And Orientation Set, the Lens Configuration Set or the Frame Capture Set originate in the camera system itself, then the data stream containing the sets carrying those values shall be described by a DataStreamSubDescriptor. If the values of items in those sets originate outside the camera system itself, then the data stream containing the sets carrying those values shall be described by a DataStreamDeviceSubDescriptor.

NOTE As an illustrative example, the camera system might originate roll and tilt information using internal sensors, but have no way of sensing the pan angle of the camera. If the camera were mounted on a swiveling base that could digitally encode the pan angle, then all of the roll, tilt and pan information could be conveyed, with roll and tilt carried in a data stream described by a DataStreamSubDescriptor, and pan carried in a data stream described by a DataStreamDeviceSubDescriptor, the device-specific fields of the latter being filled in with the relevant information about the swiveling base and its pan-angle encoder.

6.3.2.1 Camera Position And Orientation

Metadata concerning the camera position and orientation shall be carried by a Camera Position And Orientation Set.

6.3.2.1.1 Camera Position And Orientation Set ULs

The ULs of the Camera Position And Orientation Set shall be as specified in Table 28.

Table 28 — Camera Position And Orientation ULs

Item Name	Symbol	Kind	Item UL
Camera Position And Orientation	CameraPositionAndOrientation	LEAF	060e2b34.027f0101.0e170102.0c000000
X Coordinate	XCoordinate	LEAF	060e2b34.0101010e.0e170102.0c010000
Y Coordinate	YCoordinate	LEAF	060e2b34.0101010e.0e170102.0c020000
Tilt	Tilt	LEAF	060e2b34.0101010e.0e170102.0c030000
Roll	Roll	LEAF	060e2b34.0101010e.0e170102.0c040000
Z Coordinate	ZCoordinate	LEAF	060e2b34.0101010e.0e170102.0c050000
Coordinate Reference System	CoordinateReferenceSystem	LEAF	060e2b34.0101010e.0e170102.0c060000
Pan	Pan	LEAF	060e2b34.0101010e.0e170102.0c070000

6.3.2.1.2 Camera Position And Orientation Set

The Camera Position And Orientation Set shall be a subclass of StreamingData and shall be as specified in Table 29.

Table 29 — Camera Position And Orientation Set

Symbol	Type	Len	Req ?	Meaning	Default
CameraPositionAndOrientation	Set Key	16	Req	Defines Camera Position And Orientation Set.	
Length	BER Length	4	Req	Set Length.	
All items in StreamingData except the Key or Group UL and the Length, if present					
XCoordinate	SingleFloat	4	Opt	Second horizontal coordinate.	
YCoordinate	SingleFloat	4	Opt	First horizontal coordinate.	
Tilt	SingleFloat	4	Opt	Camera tilt angle in degrees, with increasing rotation angle corresponding to clockwise movement of the camera body as seen from the left of the camera body.	
Roll	SingleFloat	4	Opt	Camera roll angle in degrees, with increasing rotation angle corresponding to clockwise movement of the camera body as seen from the rear of the camera body.	
ZCoordinate	SingleFloat	4	Opt	Vertical coordinate.	
CoordinateReferenceSystem	UTF16String	var	Opt	Reference to a definition in a register of geodetic codes and parameters conforming to the requirements of ISO/TS 19127.	

Symbol	Type	Len	Req ?	Meaning	Default
Pan	SingleFloat	4	Opt	Camera pan angle in degrees, with increasing rotation angle corresponding to clockwise movement of the camera body as seen from above, in the half-closed interval [0, 360).	

6.3.2.1.3 Lens Configuration

Metadata concerning the configuration of the lens whose values can vary throughout the duration of the clip shall be carried by a Lens Configuration Set. Definition of the raw lens or motor encoding data carried by items of the Lens Configuration Set is outside the scope of this document.

6.3.2.1.3.1 Lens Configuration Set ULs

The ULs of the Lens Configuration Set shall be as specified in Table 30.

Table 30 — Lens Configuration Set ULs

Item Name	Symbol	Kind	Item UL
Lens Configuration	LensConfiguration	LEAF	060e2b34.027f0101.0e170102.0d000000
Focus Distance Metric	FocusDistanceMetric	LEAF	060e2b34.0101010e.0e170102.0d010000
Focus Distance Imperial	FocusDistanceImperial	LEAF	060e2b34.0101010e.0e170102.0d020000
Linear Iris	LinearIris	LEAF	060e2b34.0101010e.0e170102.0d030000
Entrance Pupil Offset	EntrancePupilOffset	LEAF	060e2b34.0101010e.0e170102.0d040000
Axial Nominal Focal Length	AxialNominalFocalLength	LEAF	060e2b34.0101010e.0e170102.0d050000
Axial Effective Focal Length	AxialEffectiveFocalLength	LEAF	060e2b34.0101010e.0e170102.0d060000
Lens Raw Encoder Value Focus	LensRawEncoderValueFocus	LEAF	060e2b34.0101010e.0e170102.0d070000
Motor Raw Encoder Value Focus	MotorRawEncoderValueFocus	LEAF	060e2b34.0101010e.0e170102.0d080000
Lens Raw Encoder Value Iris	LensRawEncoderValueIris	LEAF	060e2b34.0101010e.0e170102.0d090000
Motor Raw Encoder Value Iris	MotorRawEncoderValueIris	LEAF	060e2b34.0101010e.0e170102.0d0a0000
Lens Raw Encoder Value Focal Length	LensRawEncoderValueFocalLength	LEAF	060e2b34.0101010e.0e170102.0d0b0000
Motor Raw Encoder Value Focal Length	MotorRawEncoderValueFocalLength	LEAF	060e2b34.0101010e.0e170102.0d0c0000
Optical Image Stabilization	OpticalImageStabilization	LEAF	060e2b34.0101010e.0e170102.0d0d0000

6.3.2.1.3.2 Lens Configuration Set

The Lens Configuration Set shall be a subclass of StreamingData and shall be as specified in Table 31.

Table 31 — Lens Configuration Set

Symbol	Type	Len	Req ?	Meaning	Default
LensConfiguration	Set Key	16	Req	Defines Lens Configuration Set.	
Length	BER Length	4	Req	Set Length.	
All items in StreamingData except the Key or Group UL and the Length, if present					
FocusDistanceMetric	Int32	4	Opt	Lens focus distance, in millimeters, measured from the point where the lens optical axis intersects the reference plane, with a value of -1 indicating a value of infinity.	
FocusDistanceImperial	Int32	4	Opt	Lens focus distance, in thousandths of an inch, measured from the point where the lens optical axis intersects the reference plane, with a value of -1 indicating a value of infinity.	
LinearIris	Int32	4	Opt	T-stop as linear iris value, in thousandths of a stop, with T1.0 being 1000; if -2, the iris is fully closed; if -3, the iris is nearly closed in that the lens iris ring is past its last marked position, but not to where the iris is fully closed. NOTE 1 The T-stop corresponds to the f-number of an ideal lens of 100% transmittance that would produce the same image illuminance on axis as the actual lens to which the Linear Iris item pertains, at the aperture at which the actual lens was configured.	
EntrancePupilOffset	Int32	4	Opt	Distance, in µm, along the lens optical axis between the reference plane and the entrance pupil, with a value of -1 indicating a value of infinity.	
AxialNominalFocalLength	Int32	4	Opt	Indicated focal length of the lens, in µm, when focused at infinity. NOTE 2 The nominal focal length of the lens is typically printed on the lens barrel for a 'prime' (that is, fixed focal length) lens; for a 'zoom' lens the approximate nominal focal length can typically be read off of a lens control.	
AxialEffectiveFocalLength	Int32	4	Opt	Distance, in µm, between the back nodal point and the back focus point. NOTE 3 The represented effective focal length can be thought of as the focal length of a hypothetical pinhole camera that has the same field of view as the lens at the time of image creation.	

Symbol	Type	Len	Req ?	Meaning	Default
LensRawEncoderValueFocus	UInt32	4	Opt	Undecoded value from focus lens encoder.	
MotorRawEncoderValueFocus	UInt32	4	Opt	Undecoded value from focus motor encoder.	
LensRawEncoderValueIris	UInt32	4	Opt	Undecoded value from linear iris lens encoder.	
MotorRawEncoderValueIris	UInt32	4	Opt	Undecoded value from linear iris motor encoder.	
LensRawEncoderValueFocalLength	UInt32	4	Opt	Undecoded value from focal length lens encoder.	
MotorRawEncoderValueFocalLength	UInt32	4	Opt	Undecoded value from focal length motor encoder.	
OpticalImageStabilization	Boolean	1	Opt	If optical image stabilization has been applied to produce the output, the Boolean value shall be TRUE, otherwise (optical image stabilization has not been applied) the Boolean value shall be FALSE.	

6.3.2.1.4 Frame Capture

Metadata concerning the hardware and software camera settings whose values can vary throughout the duration of the clip shall be carried by a Frame Capture Set.

6.3.2.1.4.1 Frame Capture Set ULs

The ULs of Frame Capture Set items shall be as specified in Table 32.

Table 32 — Frame Capture Set ULs

Item Name	Symbol	Kind	Item UL
Frame Capture	FrameCapture	LEAF	060e2b34.027f0101.0e170102.0e000000
Exposure Time	ExposureTime	LEAF	060e2b34.0101010e.0e170102.0e010000
Averaged Frames	AveragedFrames	LEAF	060e2b34.0101010e.0e170102.0e020000
Exposure Index	ExposureIndex	LEAF	060e2b34.0101010e.0e170102.0e030000
ND Filter Index	NDFilterIndex	LEAF	060e2b34.0101010e.0e170102.0e040000
Color Temperature	ColorTemperature	LEAF	060e2b34.0101010e.0e170102.0e050000
Tint	Tint	LEAF	060e2b34.0101010e.0e170102.0e060000
Sensor Sample Rate	SensorSampleRate	LEAF	060e2b34.0101010e.0e170102.0e070000
ND Filter Density	NDFilterDensity	LEAF	060e2b34.0101010e.0e170102.0e080000
Image Checksum	ImageChecksum	LEAF	060e2b34.0101010e.0e170102.0e090000

6.3.2.1.4.2 Frame Capture Set

The Frame Capture Set shall be a subclass of StreamingData and shall be as specified in Table 33. All items marked as required shall be present. Any item marked as optional may be present. The Frame Capture Set may include additional items, with any such items being outside the scope of this document.

Table 33 — Frame Capture Set

Symbol	Type	Len	Req ?	Meaning	Default
FrameCapture	Set Key	16	Req	Defines metadata concerning individual frames of a clip, the values of which might vary throughout the duration of the clip.	
Length	BER Length	4	Req	Set Length.	
All items in StreamingData except the Key or Group UL and the Length, if present					
ExposureTime	Rational	8	Req	Exposure time of the imaging sensor in seconds.	
AveragedFrames	UInt32	4	Req	Number of frames sampled from the image sensor that have been averaged to produce this frame.	
ExposureIndex	UInt32	4	Req	Numerical value inversely proportional to the exposure provided to the camera sensor to obtain the captured image. NOTE The exposure index is often colloquially termed the 'ISO setting'.	
NDFilterIndex	UInt16	2	Req	Index into internal camera model-specific table used in color processing.	
ColorTemperature	UInt16	2	Req	Correlated Color Temperature T in Kelvin, where 2000 ≤ T ≤ 11000.	
Tint	SingleFloat	4	Req	Green/Magenta (GM) correction applied to color temperature, with -16.0 ≤ GM ≤ +16.0, with a unit increment representing the color shift of a Kodak CC035 green or magenta filter.	
SensorSampleRate	Rational	8	B. Effort	Sampling rate of the image sensor in frames per second, calculated as the reciprocal of the difference between the time of the center of this frame's exposure interval and the time of the center of the prior frame's exposure interval. If there is no prior frame, the rate may be calculated as if such a frame existed at a desired time. A value of 0/0 shall be used to indicate the frame's sample rate is undefined.	
NDFilterDensity	SingleFloat	4	Opt	Optical density of the active ND filter.	
ImageChecksum	UInt32	4	Opt	Image checksum value. The data supplied to the image checksum algorithm denoted by the Image Checksum Type of the Clip SubDescriptor are those of the Stored Data exclusive of the Start Fill and End Fill as specified in SMPTE ST 377-1:2019 §G.1.1. The image checksum value shall be present if and only if the image checksum type is present in the Clip SubDescriptor.	

NOTE The rate at which the images are originated is the quotient of the Sensor Sample Rate item and the Averaged Frames item. For example, if Sensor Sample Rate were 48000/1001 and Averaged Frames were 2, then the rate at which the images comprising the clip would be acquired would be 24000/1001.

6.4 Descriptive Metadata

Static metadata not essential to image formation or image interpretation shall be carried using a Descriptive Metadata Scheme. This document describes such a Descriptive Metadata Scheme, specified herein as a Clip Descriptive Metadata Scheme, as well as a Clip Descriptive Metadata Framework Set and a Clip Descriptive Metadata Set. A unique instance of the Clip Descriptive Metadata Framework Set shall strongly reference one or more instances of the Clip Descriptive Metadata Set. Each instance of the Clip Descriptive Metadata Set shall encode metadata as appropriate for the IANA media type identified by the DescriptiveMetadataMediaType item.

Every file described by this Registered Disclosure Document shall have a Static Track (DM), referenced by the top-level file package. The Sequence item of that Static Track (DM) shall reference exactly one Sequence (DM), and the Structural Components item of that Sequence (DM) shall reference exactly one DM Segment.

The DM Framework item of this DM Segment, in turn, shall reference exactly one Clip Descriptive Metadata Framework Set. The Descriptive Metadata Scheme item of the DM Segment shall be present and its value shall be that of the Clip Descriptive Metadata Scheme UL specified in Table 34.

6.4.1 Clip Descriptive Metadata Scheme

The DM Schemes item of the Preface Set shall include exactly one instance of the UL specified in Table 34.

Table 34 — Clip Descriptive Metadata Scheme UL

Item Name	Symbol	Kind	Item UL
Clip Descriptive Metadata Scheme	`ClipDescriptiveMetadataScheme`	LEAF	`060e2b34.0401010d.0e170102.07010000`

6.4.2 Clip Descriptive Metadata Framework

There shall be at least one instance of the Clip Descriptive Metadata Framework Set.

6.4.2.1 Clip Descriptive Metadata Framework Set ULs

The ULs of the Clip Descriptive Metadata Framework Set shall be as specified in Table 35.

Table 35 — Clip Descriptive Metadata Framework Set ULs

Item Name	Symbol	Kind	Item UL
Clip Descriptive Metadata Framework	`ClipDescriptiveMetadataFramework`	LEAF	`060e2b34.027f0101.0e170102.0f000000`
Descriptive Metadata	`DescriptiveMetadata`	LEAF	`060e2b34.0101010e.0e170102.0f010000`

6.4.2.2 Clip Descriptive Metadata Framework Set

The Clip Descriptive Metadata Framework Set, a concrete subclass of Descriptive Framework, shall be as specified in Table 36. It shall contain an inherited InstanceUID from Descriptive Framework, but shall not contain any optional item from Descriptive Framework.

Table 36 — Clip Descriptive Metadata Framework Set

Symbol	Type	Len	Req ?	Meaning	Default
ClipDescriptiveMetadataFramework	Set Key	16	Req	Identifies a Clip Descriptive Metadata Framework.	
Length	BER Length	4	Req	Set Length.	
All items in ST 377-1:2019 D.1 (Descriptive Framework) except the Key or Group UL and the Length, if present					
DescriptiveMetadata	Batch of StrongRef (ClipDescriptiveMetadata)	8+16*n	Req	Batch of Strong References to Clip Descriptive Metadata.	

6.4.3 Clip Descriptive Metadata

6.4.4 Clip Descriptive Metadata Set ULs

The ULs of the Clip Descriptive Metadata shall be as specified in Table 37.

Table 37 — Clip Descriptive Metadata Set ULs

Item Name	Symbol	Kind	Item UL
Clip Descriptive Metadata	ClipDescriptiveMetadata	LEAF	060e2b34.027f0101.0e170102.10000000
Descriptive Metadata Name	DescriptiveMetadataName	LEAF	060e2b34.0101010e.0e170102.10010000
Descriptive Metadata Media Type	DescriptiveMetadataMediaType	LEAF	060e2b34.0101010e.0e170102.10020000
Descriptive Metadata Payload	DescriptiveMetadataPayload	LEAF	060e2b34.0101010e.0e170102.10030000
Descriptive Metadata Schema URI	DescriptiveMetadataSchemaURI	LEAF	060e2b34.0101010e.0e170102.10040000

6.4.5 Clip Descriptive Metadata Set

The Clip Descriptive Metadata Set, a concrete subclass of Descriptive Object, shall be as specified in Table 38. It shall contain an inherited InstanceUID from Descriptive Object, but shall not contain any optional item from Descriptive Object.

Table 38 — Clip Descriptive Metadata Set

Symbol	Type	Len	Req ?	Meaning	Default
`ClipDescriptiveMetadata`	Set Key	16	Req	Defines Clip Descriptive Metadata.	
`Length`	BER Length	4	Req	Set Length.	
All items in ST 377-1:2019 D.2 (Descriptive Object) except the Key or Group UL and the Length, if present					
`DescriptiveMetadataName`	UTF16String	var	Req	Name of Descriptive Metadata.	
`DescriptiveMetadataMediaType`	UTF16String	var	Req	Media type of Descriptive Metadata. The value of DescriptiveMediaType shall be either the string 'application/json' to indicate the JSON data interchange syntax specified in ISO/IEC 21778:2017, 'application/xml' or media types with a '+xml' suffix to identify XML document entities as specified by RFC 7303, or 'plain/text' to indicate text without any normatively specified structure.	
`DescriptiveMetadataPayload`	UTF8String	var	Req	Encoded payload of Descriptive Metadata.	
`DescriptiveMetadataSchemaURI`	UTF16String	var	Opt	URI of schema of payload of Descriptive Metadata.	

6.4.6 Clip Descriptive Metadata Constraints

There shall be at least one instance of Clip Descriptive Metadata in an MXF file containing essence originally captured by an ARRI camera system. Each instance of Clip Descriptive Metadata contains a UTF8String carrying encoded metadata, the structure and semantics of which are outside the scope of this document, as well as a name, a registered encoding, and a Uniform Resource Identifier (URI).

Annex C specifies the names, cardinalities, media types and schema URIs of Clip Descriptive Metadata generated by an ARRI camera system. An informative section of that Annex shows the contents of the schema denoted by those URIs at the time of this document's writing.

NOTE The URI item could usefully be employed to guide users or applications of the file to a schema for the encoded data.

Clip Descriptive Metadata Frameworks may contain additional Clip Descriptive Metadata Sets not specified in Annex C. These may be defined either by ARRI or by other parties. The names, cardinalities, media types and schema URIs of such Sets are outside the scope of this document, save that the definition of any such Set shall include its name, cardinality, media type and schema URI.

7 Application Considerations

7.1 Partition Structure

The value of the HeaderByteCount item in the Header Partition Pack shall be at least 16,777,216 (2^{24}) bytes.

7.2 System Item

The System Item shall be as described in 6.3.1.

Annex A Supplemental Data Element (normative)

This annex defines an essence container that specifies the mapping of supplemental data into a data element of the MXF Generic Container. In the context of SMPTE RDD 55, this container is used to carry streaming camera state metadata.

NOTE: SMPTE ST 429-14 specifies a similar mapping of unspecified data into the data element of the MXF Generic Container. The use of SMPTE ST 429-14 is, however, constrained to Digital Cinema Packages and MXF OP-Atom applications.

A.1 Supplemental Data Element Wrapping

The Supplemental Data element shall use frame wrapping.

A.2 Supplemental Data Element KLV Coding

A.2.1 Element Key

Supplemental Data essence shall be identified with the Supplemental Data element key specified in Table A.1. Bytes 1-13 of the label are defined in SMPTE ST 379-2.

Table A.1 — Supplemental Data Element Key

Item Name	Symbol	Kind	Item UL
Supplemental Data Element	SupplementalDataElement	LEAF	060e2b34.01020101.0d010301.17kk0fnn

The symbol ‘kk’ is the count of Data Elements in the Data Item, and the symbol ‘nn’ is the number (used as an Index) of this Data Element in the Data Item.Element Length

The BER length field of the KLV triple for a Supplemental Data element should be 4 bytes BER long-form encoded.

If the byte count of the KLV value cannot be expressed using 4-byte BER encoding, the BER length of the KLV triple for a Supplemental Data element shall be 8 byte BER long-form encoded.

The size of the length shall be constant for all edit units of the essence container and shall be either 4 bytes or 8 bytes as appropriate.

A.2.2 Element Value

The value of a Supplemental Data element shall be a sequence of KLV encoded data.

A.3 Supplemental Data Element Essence Container Label

The Supplemental Data Element Essence Container Label shall be as specified in Table A.2.

Table A.2 — Supplemental Data Element Essence Container Label

Item Name	Symbol	Kind	Item UL
Supplemental Data Essence Container	MXFGCSupplementalDataFrameWrapped	LEAF	060e2b34.0401010d.0d010301.02260100

A.4 Supplemental Data Element Essence Descriptor

Supplemental Data Element Essence Descriptor shall be a subclass of Generic Data Essence Descriptor.

Applications of the Supplemental Data Element Essence Container shall use SubDescriptors to identify the payload in the essence container.

A.4.1 Supplemental Data Element Essence Descriptor ULs

The ULs of the Supplemental Data Element Essence Descriptor shall be as specified in Table A.3.

Table A.3 — Supplemental Data Essence Descriptor ULs

Item Name	Symbol	Kind	Item UL
Supplemental Data Essence Descriptor	`SupplementalDataEssenceDescriptor`	LEAF	`060e2b34.027f0101.0d010101.0101810a`

A.4.2 Supplemental Data Element Essence Descriptor

The Supplemental Data Element Essence Descriptor shall be as specified in Table A.4.

Table A.4 — Supplemental Data Element Essence Descriptor

Symbol	Type	Len	Req ?	Meaning	Default
`SupplementalDataEssenceDescriptor`	Set Key	16	Req	Defines a Supplemental Data Descriptor.	
`Length`	BER Length	4	Req	Set Length.	
All items in ST 377-1:2019 F.6 (Data Essence Descriptor) except the Key or Group UL and the Length, if present					

The Data Essence Coding item of the descriptor shall be present.

A.5 Supplemental Data Element Applications

Applications of the Supplemental Data element essence container shall specify

- the payload to be carried inside the value of Supplemental Data elements, and the KLV encoding of that payload,
- the essence encoding label that shall be used to identify the payload, and
- the sub-descriptor or sub-descriptors that shall be used to describe the payload.

Annex B Generic Sets for the Encoding of Data Streams in Supplemental Data Essence Elements (normative)

This annex specifies a template that allows carriage of streams of data inside Supplemental Data elements.

The data are encoded as KLV sets, subclasses of a Streaming Data set defined in this Annex. The user of the data stream(s) shall encode concrete subclasses of StreamingData. The DataStreamSchemeURI item in DataStreamSubDescriptor shall be used to indicate the semantics of the data in the data stream.

The template allows for the transport of one or more streams of data inside one Supplemental Data element. The data streams may originate from one or more distinct sources.

The rate of items in any of the data streams may be the same as or different from the edit rate of the Generic Container of which the Supplemental Data element is a part.

B.1 Data Essence

The Supplemental Data element shall carry a sequence of data sets that are concrete subclasses of Streaming Data. These data sets shall be encoded as SMPTE ST 336 Universal Sets.

B.1.1 Streaming Data

B.1.1.1 Streaming Data Set ULs

The ULs of the Streaming Data Set key and of the Streaming Data Set items shall be as specified in Table B.1.

Table B.1 — Streaming Data Set

Item Name	Symbol	Kind	Item UL
Streaming Data	StreamingData	LEAF	060e2b34.027f0101.0c050101.00000000
Data Stream ID	DataStreamID	LEAF	060e2b34.0101010e.0103040b.00000000
Creation Timestamp	CreationTimestamp	LEAF	060e2b34.0101010e.07020110.01050000

B.1.1.2 Streaming Data Set

The Streaming Data Set shall be as specified in Table B.2.

Table B.2 — Streaming Data Set

Symbol	Type	Len	Req ?	Meaning	Default
StreamingData	Set Key	16	Req	Defines a Data Stream Set.	
Length	BER Length	4	Req	Set Length.	
DataStreamID	UUID	16	Req	Unique identifier for this Data Stream.	
CreationTimestamp	PTPTimestamp	10	Opt	IEEE 1588-2019 PTP Timestamp documenting the time at which the data in the data set were created.	

Streaming Data shall be an abstract class.

The value of the DataStreamID item identifies the supplemental data stream. All Streaming Data sets that share the same value of DataStreamID are part of the same data stream.

NOTE A data stream may be composed of different concrete subclasses of StreamingData.

B.2 Data Stream Essence Coding

When the Supplemental Data element essence is used to carry Data Stream content, the Data Essence Coding item of the Supplemental Data element essence descriptor shall be as shown in Table B.3.

Table B.3 — Data Stream Data Essence Coding Label

Item Name	Symbol	Kind	Item UL
Data Stream Data Essence Coding	DataStreamDataEssenceCoding	LEAF	060e2b34.0401010d.04030301.00000000

B.3 Mapping of Streaming Data into Supplemental Data Elements

Streaming data sets may have a rate that is different from the edit rate of the essence container.

The KLV value of a Supplemental Data element may contain zero, one or multiple streaming data sets of the same type for the same SubDescriptor.

When a streaming data set of the same type for the same SubDescriptor is produced at a rate that is lower than the edit rate of the essence container, the data set shall be repeated in subsequent edit units as long as the information it encodes applies to the edit units of the Generic Container. If the data set's contents are no longer valid for an edit unit of the essence container then the data set shall not be encoded in the value of the Supplemental Data Element.

When a streaming data set of the same type for the same SubDescriptor is produced at a rate that is higher than the edit rate of the essence container, all data sets that are valid within the time interval represented by the edit unit of the Generic Container shall be encoded into the value of the Supplemental Data Element such that they are ordered according to their sample time.

A Supplemental Data element may contain streaming data sets from multiple data streams.

If a StreamingData subclass for a data stream encodes the Creation Timestamp property, that property shall be encoded in all instances of that subclass in the data stream.

NOTE 1 For streaming data sets that encode the CreationTimestamp value this means that the successively-generated sets of the same type are ordered by increasing byte offset for the same descriptor.

NOTE 2 There is no relative ordering requirement for the storage of streaming data sets within the value of a Supplemental Data Element among data sets that are referenced by different SubDescriptors or that are of different type.

B.4 Data Stream SubDescriptors

The Supplemental Data Essence Descriptor shall encode one of two SubDescriptors defined in this section for each of the data streams inside the Supplemental Data element essence container.

The Data Stream SubDescriptor is used to identify the data stream and — optionally — the real-time sample rate of items inside the data stream.

The Data Stream Device SubDescriptor adds information that allows identification of the specific device that produces the data stream.

B.4.1 Data Stream SubDescriptor

A data stream may be composed of data sets of one or more types. All data sets that constitute part of the same data stream shall encode the same value in their DataStreamID property. The LinkedDataStreamID item of the Data Stream SubDescriptor shall weakly reference this DataStreamID.

NOTE It is possible that the LinkedDataStreamID may contain a value that does not correspond to any DataStreamID value in the essence container, especially in cases where the file in which the essence container is present is a partial copy of a larger file.

B.4.1.1 Data Stream SubDescriptor ULs

The ULs of the Data Stream SubDescriptor Key and of the DataStreamSubDescriptor items shall be as specified in Table B.4.

Table B.4 — Data Stream SubDescriptor ULs

Item Name	Symbol	Kind	Item UL
Data Stream SubDescriptor	DataStreamSubDescriptor	LEAF	060e2b34.027f0101.0d010101.0101810b
Data Stream Scheme URI	DataStreamSchemeURI	LEAF	060e2b34.0101010e.04061003.00000000
Linked Data Stream ID	LinkedDataStreamID	LEAF	060e2b34.0101010e.06010102.07000000
Data Stream Sample Rate	DataStreamSampleRate	LEAF	060e2b34.0101010e.04060103.00000000

B.4.1.2 Data Stream SubDescriptor

The Data Stream SubDescriptor shall be a subclass of SubDescriptor and shall be as specified in Table B.5.

Table B.5 — Data Stream SubDescriptor

Symbol	Type	Len	Req ?	Meaning	Default
DataStreamSubDescriptor	Set Key	16	Req	Defines a Data Stream SubDescriptor.	
Length	BER Length	4	Req	Set Length.	
All items in ST 377-1:2019 B.3 (SubDescriptor) except the Key or Group UL and the Length, if present					
DataStreamSchemeURI	UTF16String	var	Req	URI identifying application of Data Stream. The syntax of the URI shall be as specified in IETF RFC 3986.	
LinkedDataStreamID	UUID	16	Req	Unique identifier of Data Stream described by this SubDescriptor.	
DataStreamSampleRate	Rational	8	Opt	Rate at which creator samples data conveyed on the Data Stream, per second.	

B.4.2 Data Stream Device SubDescriptor

The Data Stream Device SubDescriptor links a data stream to a particular originating device. A single device may have multiple associated data streams. When a device creates more than one distinct data stream, each shall have a distinct DataStreamID value and a distinct SubDescriptor.

B.4.2.1 Data Stream Device SubDescriptor ULs

The ULs of the Data Stream Device SubDescriptor Key and of the Data Stream Device SubDescriptor items shall be as specified in Table B.6.

Table B.6 — Data Stream Device SubDescriptor ULs

Item Name	Symbol	Kind	Item UL
Data Stream Device SubDescriptor	DataStreamDeviceSubDescriptor	LEAF	060e2b34.027f0101.0d010101.0101810c
Device Creator Name	DeviceCreatorName	LEAF	060e2b34.0101010e.05010206.01000000
Device Product Name	DeviceProductName	LEAF	060e2b34.0101010e.05010206.02000000
Device Version	DeviceVersion	LEAF	060e2b34.0101010e.05010206.03000000
Device Instance Identifier	DeviceInstanceIdentifier	LEAF	060e2b34.0101010e.05010206.04000000

B.4.2.2 Data Stream Device SubDescriptor

The Data Stream Device SubDescriptor shall be a subclass of Data Stream SubDescriptor and shall be as specified in Table B.7.

Table B.7 — Data Stream Device SubDescriptor

Symbol	Type	Len	Req ?	Meaning	Default
DataStreamDeviceSubDescriptor	Set Key	16	Req	Defines a Data Stream Device SubDescriptor.	
Length	BER Length	4	Req	Set Length.	
All items in DataStreamSubDescriptor except the Key or Group UL and the Length, if present					
DeviceCreatorName	UTF16String	var	Req	Name of creator of device creating a Data Stream.	
DeviceProductName	UTF16String	var	Req	Name identifying product creating a Data Stream.	
DeviceVersion	UTF16String	var	Req	Version identifier for device creating a Data Stream.	
DeviceInstanceIdentifier	UTF16String	var	Req	Unambiguous identifier for device creating a Data Stream, in the context of the application. NOTE This might be the device's serial number or another appropriate identifier that allows unique identification within the context of the user's application.	

B.5 Data Stream SubDescriptor Reference to Streaming Data Example (informative)

A simple example of the use of the Data Stream SubDescriptor's LinkedDataStreamID is shown in Figure B.1.

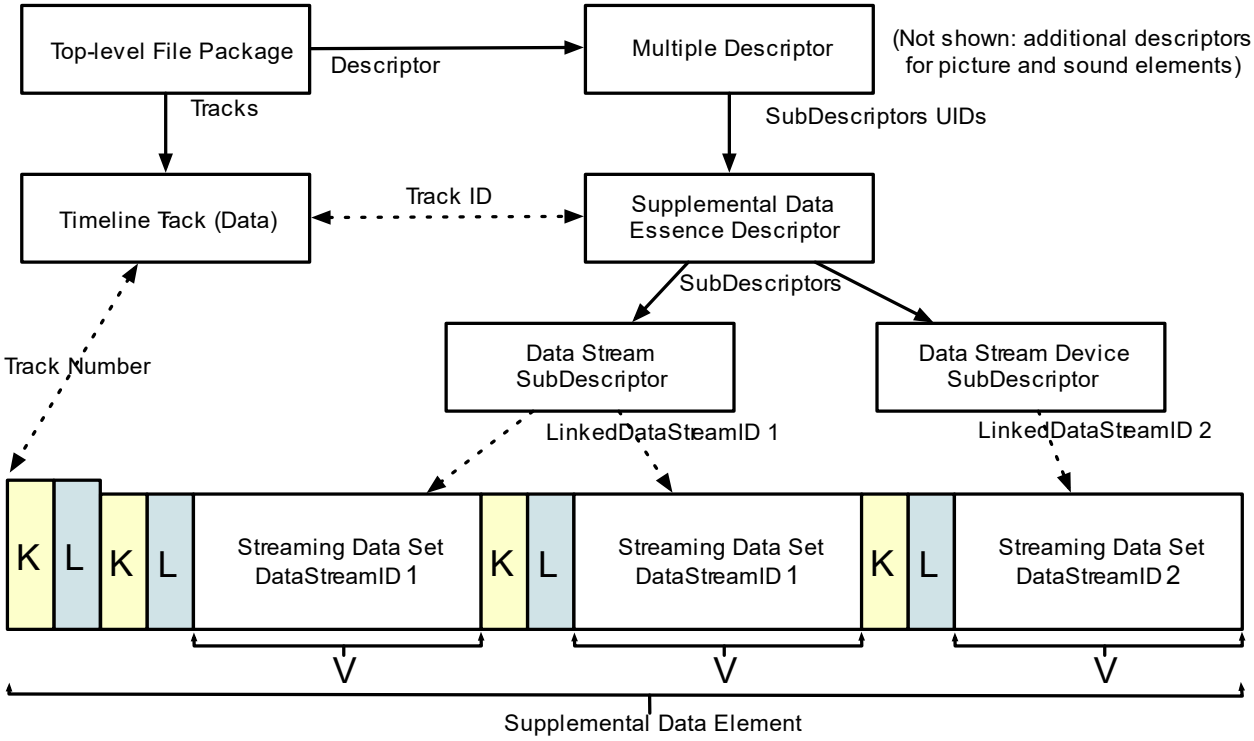

Figure B.1 — Data Stream SubDescriptor references to Streaming Data

Annex C Clip Descriptive Metadata Sets created by ARRI camera systems (normative)

Clip Descriptive Metadata Set instances shall be referenced by the Clip Descriptive Metadata Framework Set specified in 6.4.2. This annex specifies eight Clip Descriptive Metadata Sets created by ARRI camera systems. For each of these Clip Descriptive Metadata Sets, the Set's name, cardinality, media type and a URI by which the Set's most recent clip descriptive metadata schema, expressed in JSON, can be obtained, shall be as specified below.

An informative section follows each specification, in which the contents of the referenced schema (at the time this document was written) are shown. Users of this specification should check https://www.arri.com/schema/json/camera/ for potential updates to these schema, and for definitions of Clip Descriptive Metadata Sets created after this document was written.

C.1 Camera Device

Name: Camera Device
Cardinality: 1
Media Type: 'application/json'
Schema URI: https://www.arri.com/schema/json/camera/camera_device/v1-0-0

C.1.1 Camera Device Schema Contents (informative)

```
{
	"$schema": "http://json-schema.org/draft-07/schema#",
	"$id": "https://www.arri.com/schema/json/camera/camera_device/v1-0-0",
	"title": "Camera Device",
	"description": "Information about the camera device",
	"type": "object",
	"properties": {
		"cameraModel": {
			"title": "Model",
			"description": "Camera model name",
			"type": "string"
		},
		"cameraSerialNumber": {
			"title": "Serial Number",
			"description": "Serial number of the camera unit",
			"type": "string"
		},
		"cameraSoftwarePackageName": {
			"title": "Software Package Name",
			"description": "Name of the software package installed on the camera unit",
			"type": "string"
		},
		"cameraSoftwarePackageVersion": {
			"title": "Software Package Version",
			"description": "Version of the software package installed on the camera unit",
			"type": "string",
			"pattern": "^\\d+\\.\\d+\\.\\d+(\\-[A-Za-z][A-Za-z0-9\\.\\-]*)?(\\+\\d+)?$"
		},
		"cameraVariant": {
			"title": "Variant",
			"description": "Name of the camera model variant",
			"type": "string"
		}
	},
	"required": [
		"cameraModel",
		"cameraSerialNumber"
	]
}
```

C.2 Lens Device

Name: Lens Device
Cardinality: 1, with all top-level properties being optional
Media Type: ‘application/json’
Schema URI: https://www.arri.com/schema/json/camera/lens_device/v1-0-0

C.2.1 Lens Device Schema Contents (informative)


```
{
	"$schema": "http://json-schema.org/draft-07/schema#",
	"$id": "https://www.arri.com/schema/json/camera/lens_device/v1-0-0",
	"title": "Lens Device",
	"description": "Description of the static parameters of the lens",
	"type": "object",
	"properties": {
		"lensLimitsFocusImperial": {
			"title": "Lens Focus Distance Limits Imperial",
			"description": "Focus distance limits of the lens in imperial form",
			"type": "object",
			"properties": {
				"max": {
					"title": "Maximum",
					"description": "Maximum focus distance",
					"type": "integer",
					"oneOf": [
						{
							"description": "value in 1/1000\"",
							"type": "integer",
							"exclusiveMinimum": 0
						},
						{
							"description": "infinity",
							"type": "integer",
							"const": -1
						},
						{
							"description": "not available",
							"type": "integer",
							"const": 0
						}
					]
				},
				"min": {
					"title": "Minimum",
					"description": "Minimum focus distance",
					"type": "integer",
					"oneOf": [
						{
							"description": "value in 1/1000\"",
							"type": "integer",
							"exclusiveMinimum": 0
						},
						{
							"description": "infinity",
							"type": "integer",
							"const": -1
						},
						{
							"description": "not available",
							"type": "integer",
							"const": 0
						}
					]
				}
			},
			"required": [
				"min",
				"max"
			]
```

```
},
"lensLimitsFocusMetric": {
        "title": "Lens Focus Distance Limits Metric",
        "description": "Focus distance limits of the lens in metric form",
        "type": "object",
        "properties": {
                "max": {
                        "title": "Maximum",
                        "description": "Maximum focus distance",
                        "type": "integer",
                        "oneOf": [
                                {
                                        "description": "value in mm",
                                        "type": "integer",
                                        "exclusiveMinimum": 0
                                },
                                {
                                        "description": "infinity",
                                        "type": "integer",
                                        "const": -1
                                },
                                {
                                        "description": "not available",
                                        "type": "integer",
                                        "const": 0
                                }
                        ]
                },
                "min": {
                        "title": "Minimum",
                        "description": "Minimum focus distance",
                        "type": "integer",
                        "oneOf": [
                                {
                                        "description": "value in mm",
                                        "type": "integer",
                                        "exclusiveMinimum": 0
                                },
                                {
                                        "description": "infinity",
                                        "type": "integer",
                                        "const": -1
                                },
                                {
                                        "description": "not available",
                                        "type": "integer",
                                        "const": 0
                                }
                        ]
                }
        },
        "required": [
                "min",
                "max"
        ]
},
"lensLimitsIris": {
        "title": "Lens Linear Iris Limits",
        "description": "Linear iris limits of the lens",
        "type": "object",
        "properties": {
                "max": {
                        "title": "Maximum",
                        "description": "Lens Maximum iris",
                        "type": "integer",
                        "oneOf": [
                                {
                                        "description": "1/1000 of a stop with T1 ≙ 1000",
                                        "type": "integer",
                                        "minimum": 0
```

```
                                        },
                                        {
                                                "description": "not available",
                                                "type": "integer",
                                                "const": -1
                                        },
                                        {
                                                "description": "closed",
                                                "type": "integer",
                                                "const": -2
                                        },
                                        {
                                                "description": "near close",
                                                "type": "integer",
                                                "const": -3
                                        }
                                ]
                        },
                        "min": {
                                "title": "Minimum",
                                "description": "Lens Minimum iris",
                                "type": "integer",
                                "oneOf": [
                                        {
                                                "description": "1/1000 of a stop with T1 ≙ 1000",
                                                "type": "integer",
                                                "minimum": 0
                                        },
                                        {
                                                "description": "not available",
                                                "type": "integer",
                                                "const": -1
                                        },
                                        {
                                                "description": "closed",
                                                "type": "integer",
                                                "const": -2
                                        },
                                        {
                                                "description": "near close",
                                                "type": "integer",
                                                "const": -3
                                        }
                                ]
                        }
                },
                "required": [
                        "min",
                        "max"
                ]
        },
        "lensLimitsZoom": {
                "title": "Lens Focal Length Limits",
                "description": "Focal length limits of the lens",
                "type": "object",
                "properties": {
                        "max": {
                                "title": "Maximum",
                                "description": "Maximum focal length",
                                "type": "integer",
                                "oneOf": [
                                        {
                                                "description": "value in µm",
                                                "type": "integer",
                                                "exclusiveMinimum": 0
                                        },
                                        {
                                                "description": "not available",
                                                "type": "integer",
                                                "const": 0
```

```
                                        }
                                ]
                        },
                        "min": {
                                "title": "Minimum",
                                "description": "Minimum focal length",
                                "type": "integer",
                                "oneOf": [
                                        {
                                                "description": "value in µm",
                                                "type": "integer",
                                                "exclusiveMinimum": 0
                                        },
                                        {
                                                "description": "not available",
                                                "type": "integer",
                                                "const": 0
                                        }
                                ]
                        }
                },
                "required": [
                        "min",
                        "max"
                ]
        },
        "lensModel": {
                "title": "Lens Model",
                "description": "Name of the lens model",
                "type": "string"
        },
        "lensMotorRawEncoderLimitsFL": {
                "title": "Lens Motor Raw Encoder Focal Length Limits",
                "description": "Limits of the raw encoder values for the motor controlling the
focal length axis of the lens",
                "type": "object",
                "properties": {
                        "max": {
                                "title": "Maximum",
                                "description": "Maximum raw encoder value for the focal length
axis",
                                "type": "integer",
                                "oneOf": [
                                        {
                                                "description": "raw value",
                                                "type": "integer",
                                                "minimum": 0
                                        },
                                        {
                                                "description": "not available",
                                                "type": "integer",
                                                "const": -1
                                        }
                                ]
                        },
                        "min": {
                                "title": "Minimum",
                                "description": "Minimum raw encoder value for the focal length
axis",
                                "type": "integer",
                                "oneOf": [
                                        {
                                                "description": "raw value",
                                                "type": "integer",
                                                "minimum": 0
                                        },
                                        {
                                                "description": "not available",
                                                "type": "integer",
                                                "const": -1
```

```
                                        }
                                ]
                        }
                },
                "required": [
                        "min",
                        "max"
                ]
        },
        "lensMotorRawEncoderLimitsFocus": {
                "title": "Lens Motor Raw Encoder Focus Limits",
                "description": "Limits of the raw encoder values for the motor controlling the
focus axis of the lens",
                "type": "object",
                "properties": {
                        "max": {
                                "title": "Maximum",
                                "description": "Maximum raw encoder value for the focus axis",
                                "type": "integer",
                                "oneOf": [
                                        {
                                                "description": "raw value",
                                                "type": "integer",
                                                "minimum": 0
                                        },
                                        {
                                                "description": "not available",
                                                "type": "integer",
                                                "const": -1
                                        }
                                ]
                        },
                        "min": {
                                "title": "Minimum",
                                "description": "Minimum raw encoder value for the focus axis",
                                "type": "integer",
                                "oneOf": [
                                        {
                                                "description": "raw value",
                                                "type": "integer",
                                                "minimum": 0
                                        },
                                        {
                                                "description": "not available",
                                                "type": "integer",
                                                "const": -1
                                        }
                                ]
                        }
                },
                "required": [
                        "min",
                        "max"
                ]
        },
        "lensMotorRawEncoderLimitsIris": {
                "title": "Lens Motor Raw Encoder Iris Limits",
                "description": "Limits of the raw encoder values for the motor controlling the
iris axis of the lens",
                "type": "object",
                "properties": {
                        "max": {
                                "title": "Maximum",
                                "description": "Maximum raw encoder value for the iris axis",
                                "type": "integer",
                                "oneOf": [
                                        {
                                                "description": "raw value",
                                                "type": "integer",
                                                "minimum": 0
```

```
                                        },
                                        {
                                                "description": "not available",
                                                "type": "integer",
                                                "const": -1
                                        }
                                ]
                        },
                        "min": {
                                "title": "Minimum",
                                "description": "Minimum raw encoder value for the iris axis",
                                "type": "integer",
                                "oneOf": [
                                        {
                                                "description": "raw value",
                                                "type": "integer",
                                                "minimum": 0
                                        },
                                        {
                                                "description": "not available",
                                                "type": "integer",
                                                "const": -1
                                        }
                                ]
                        }
                },
                "required": [
                        "min",
                        "max"
                ]
        },
        "lensScaleRawEncoderLimitsFL": {
                "title": "Lens Scale Raw Encoder Focal Length Limits",
                "description": "Limits of the raw encoder values for the focal length axis of the
lens",
                "type": "object",
                "properties": {
                        "max": {
                                "title": "Maximum",
                                "description": "Maximum raw encoder value for the focal length
axis",
                                "type": "integer",
                                "oneOf": [
                                        {
                                                "description": "raw value",
                                                "type": "integer",
                                                "minimum": 0
                                        },
                                        {
                                                "description": "not available",
                                                "type": "integer",
                                                "const": -1
                                        }
                                ]
                        },
                        "min": {
                                "title": "Minimum",
                                "description": "Minimum raw encoder value for the focal length
axis",
                                "type": "integer",
                                "oneOf": [
                                        {
                                                "description": "raw value",
                                                "type": "integer",
                                                "minimum": 0
                                        },
                                        {
                                                "description": "not available",
                                                "type": "integer",
                                                "const": -1
```

```
                                        }
                                ]
                        }
                },
                "required": [
                        "min",
                        "max"
                ]
        },
        "lensScaleRawEncoderLimitsFocus": {
                "title": "Lens Scale Raw Encoder Focus Limits",
                "description": "Limits of the raw encoder values for the focus axis of the lens",
                "type": "object",
                "properties": {
                        "max": {
                                "title": "Maximum",
                                "description": "Maximum raw encoder value for the focus axis",
                                "type": "integer",
                                "oneOf": [
                                        {
                                                "description": "raw value",
                                                "type": "integer",
                                                "minimum": 0
                                        },
                                        {
                                                "description": "not available",
                                                "type": "integer",
                                                "const": -1
                                        }
                                ]
                        },
                        "min": {
                                "title": "Minimum",
                                "description": "Minimum raw encoder value for the focus axis",
                                "type": "integer",
                                "oneOf": [
                                        {
                                                "description": "raw value",
                                                "type": "integer",
                                                "minimum": 0
                                        },
                                        {
                                                "description": "not available",
                                                "type": "integer",
                                                "const": -1
                                        }
                                ]
                        }
                },
                "required": [
                        "min",
                        "max"
                ]
        },
        "lensScaleRawEncoderLimitsIris": {
                "title": "Lens Scale Raw Encoder Iris Limits",
                "description": "Limits of the raw encoder values for the iris axis of the lens",
                "type": "object",
                "properties": {
                        "max": {
                                "title": "Maximum",
                                "description": "Maximum raw encoder value for the iris axis",
                                "type": "integer",
                                "oneOf": [
                                        {
                                                "description": "raw value",
                                                "type": "integer",
                                                "minimum": 0
                                        },
                                        {
```

```
                                                "description": "not available",
                                                "type": "integer",
                                                "const": -1
                                        }
                                ]
                        },
                        "min": {
                                "title": "Minimum",
                                "description": "Minimum raw encoder value for the iris axis",
                                "type": "integer",
                                "oneOf": [
                                        {
                                                "description": "raw value",
                                                "type": "integer",
                                                "minimum": 0
                                        },
                                        {
                                                "description": "not available",
                                                "type": "integer",
                                                "const": -1
                                        }
                                ]
                        }
                },
                "required": [
                        "min",
                        "max"
                ]
        },
        "lensScaleRawEncoderSource": {
                "title": "Lens Scale Raw Encoder Source",
                "description": "Source of the lens raw encoder values for focus, iris, focal
length",
                "type": "string",
                "oneOf": [
                        {
                                "type": "string",
                                "const": "LDS"
                        },
                        {
                                "type": "string",
                                "const": "ENG"
                        }
                ]
        },
        "lensSerialNumber": {
                "title": "Lens Serial Number",
                "description": "Serial number of the lens",
                "type": "string"
        },
        "lensSqueezeFactor": {
                "title": "Lens Squeeze Factor",
                "description": "Anamorphic squeeze factor of the lens",
                "type": "string",
                "pattern": "^[+\\-]?\\d+/\\d+$"
        }
    }
}}
```

C.3 Lens Converter

Name: Lens Converter
Cardinality: 0 or 1
Media Type: ‘application/json’
Schema URI: https://www.arri.com/schema/json/camera/lens_converter/v1-0-0

NOTE The Lens Converter Clip Descriptive Metadata will be encoded if a lens converter/extender is in use during media creation.

C.3.1 Lens Converter Schema Contents (informative)

```
{
	"$schema": "http://json-schema.org/draft-07/schema#",
	"$id": "https://www.arri.com/schema/json/camera/lens_converter/v1-0-0",
	"title": "Lens Converter",
	"description": "Description of the static properties of the attached lens converter, if any",
	"type": "object",
	"properties": {
		"converterFocalLenMultiplier": {
			"title": "Converter Focal Length Multiplier",
			"description": "Focal length multiplier of the converter",
			"type": "number"
		},
		"converterLightLossFactor": {
			"title": "Converter Light Loss Factor",
			"description": "Light loss factor of the converter",
			"type": "number"
		},
		"converterModel": {
			"title": "Converter Model",
			"description": "Name of the converter model",
			"type": "string"
		},
		"converterPhysicalLength": {
			"title": "Converter Physical Length",
			"description": "Physical length of the converter in µm",
			"type": "integer",
			"exclusiveMinimum": 0
		},
		"converterSerialNumber": {
			"title": "Converter Serial Number",
			"description": "Serial number of the converter",
			"type": "string"
		}
	},
	"required": [
		"converterModel",
		"converterSerialNumber",
		"converterPhysicalLength",
		"converterLightLossFactor",
		"converterFocalLenMultiplier"
	]
}
```

C.4 Frame Line

Name: Frame Line
Cardinality: 0, 1 or 2
Media Type: ‘application/json’
Schema URI: https://www.arri.com/schema/json/camera/monitoring/frameline/v1-0-0

If the cardinality of this Set is 2, the framelineFilename property shall differ between the two instances of the Set.

C.4.1 Frame Line Schema Contents (informative)

```
{
	"$schema": "http://json-schema.org/draft-07/schema#",
	"$id": "https://www.arri.com/schema/json/camera/monitoring/frameline/v1-0-0",
	"title": "Frame Line",
	"description": "Rectangle(s) identifying the image region(s) of interest",
	"type": "object",
	"properties": {
		"framelineFilename": {
			"title": "Frame Line File Name",
			"description": "Name of the frame line file",
			"type": "string"
		},
		"framelineRect": {
			"type": "array",
			"items": {
				"title": "Frame Line Rectangle",
				"description": "Rectangle of a frame line",
				"type": "object",
				"properties": {
					"framelineRectName": {
						"title": "Frame Line Rectangle Name",
						"description": "Optional name of the rectangle",
						"type": "string"
					},
					"framelineRectUsage": {
						"title": "Frame Line Rectangle Usage Type",
						"description": "Usage type of the frame line rectangle",
						"type": "string",
						"oneOf": [
							{
								"description": "rectangle is invisible",
								"type": "string",
								"const": "inactive"
							},
							{
								"description": "master rectangle",
								"type": "string",
								"const": "master"
							},
							{
								"description": "auxiliary rectangle",
								"type": "string",
								"const": "aux"
							}
						]
					},
					"height": {
						"title": "Height",
						"description": "Height of the rectangle in pixels",
						"type": "integer",
						"exclusiveMinimum": 0
					},
					"left": {
						"title": "Left Offset",
						"description": "Left coordinate of the top-left corner",
						"type": "integer",
						"minimum": 0
```

```
                                },
                                "top": {
                                        "title": "Top Offset",
                                        "description": "Top coordinate of the top-left corner",
                                        "type": "integer",
                                        "minimum": 0
                                },
                                "width": {
                                        "title": "Width",
                                        "description": "Width of the rectangle in pixels",
                                        "type": "integer",
                                        "exclusiveMinimum": 0
                                }
                        },
                        "required": [
                                "framelineRectUsage",
                                "left",
                                "top",
                                "width",
                                "height"
                        ]
                },
                "minLength": 1,
                "maxLength": 3
        }
    },
    "required": [
        "framelineFilename",
        "framelineRect"
      ]
}
```

C.5 ARRI Video Parameters Gen2

Name: ARRI2 Video Parameters Gen2
Cardinality, 0 or 1
Media Type: ‘application/json’
Schema URI: https://www.arri.com/schema/json/camera/processing/arri2_video_params/v1-0-0

NOTE ARRI Video Parameters Gen2 Clip Descriptive Metadata will be encoded if a 3D LUT Set is present in header metadata and the contents of that 3D LUT Set were created by adjusting ALF-2 video look parameters.

C.5.1 ARRI2 Video Parameters Gen2 Schema Contents (informative)

```
{
    "$schema": "http://json-schema.org/draft-07/schema#",
    "$id": "https://www.arri.com/schema/json/camera/processing/arri2_video_params/v1-0-0",
    "title": "ARRI Video Parameters Gen2",
    "description": "LUT design for ALF-2 parametric video looks",
    "type": "object",
    "properties": {
        "blackGamma": {
            "title": "Black Gamma",
            "type": "number",
            "minimum": 0,
            "maximum": 1
        },
        "gamma": {
            "title": "Gamma",
            "type": "number",
            "minimum": 0.5,
            "maximum": 2
        },
        "knee": {
            "title": "Knee",
            "type": "number",
            "minimum": 0,
            "maximum": 1
```

```
		},
		"lut3DID": {
			"title": "3D LUT Identifier",
			"description": "Identifier of the 3D LUT, referenced by its design data and associated
CDL",
			"type": "string",
			"pattern": "^\\i\\c*$"
		},
		"saturation": {
			"title": "Saturation",
			"type": "number",
			"minimum": 0,
			"maximum": 2
		},
		"saturationByHue": {
			"title": "Saturation By Hue",
			"type": "object",
			"properties": {
				"blue": {
					"title": "Blue Saturation",
					"type": "number",
					"minimum": 0,
					"maximum": 2
				},
				"cyan": {
					"title": "Cyan Saturation",
					"type": "number",
					"minimum": 0,
					"maximum": 2
				},
				"green": {
					"title": "Green Saturation",
					"type": "number",
					"minimum": 0,
					"maximum": 2
				},
				"magenta": {
					"title": "Magenta Saturation",
					"type": "number",
					"minimum": 0,
					"maximum": 2
				},
				"red": {
					"title": "Red Saturation",
					"type": "number",
					"minimum": 0,
					"maximum": 2
				},
				"yellow": {
					"title": "Yellow Saturation",
					"type": "number",
					"minimum": 0,
					"maximum": 2
				}
			},
			"required": [
				"red",
				"yellow",
				"green",
				"cyan",
				"blue",
				"magenta"
			]
		},
		"videoGamma": {
			"title": "Video Gamma",
			"type": "object",
			"properties": {
				"b": {
					"title": "Blue Video Gamma",
```

```
                        "description": "Video gamma for the blue channel",
                        "type": "number",
                        "minimum": 0.5,
                        "maximum": 2
                  },
                  "g": {
                        "title": "Green Video Gamma",
                        "description": "Video gamma for the green channel",
                        "type": "number",
                        "minimum": 0.5,
                        "maximum": 2
                  },
                  "r": {
                        "title": "Red Video Gamma",
                        "description": "Video gamma for the red channel",
                        "type": "number",
                        "minimum": 0.5,
                        "maximum": 2
                  }
            },
            "required": [
                  "r",
                  "g",
                  "b"
            ]
      },
      "videoPedestal": {
            "title": "Video Pedestal",
            "type": "object",
            "properties": {
                  "b": {
                        "title": "Blue Pedestal",
                        "description": "Video pedestal for the blue channel",
                        "type": "number",
                        "minimum": -0.3,
                        "maximum": 0.3
                  },
                  "g": {
                        "title": "Green Pedestal",
                        "description": "Video pedestal for the green channel",
                        "type": "number",
                        "minimum": -0.3,
                        "maximum": 0.3
                  },
                  "r": {
                        "title": "Red Pedestal",
                        "description": "Video pedestal for the red channel",
                        "type": "number",
                        "minimum": -0.3,
                        "maximum": 0.3
                  }
            },
            "required": [
                  "r",
                  "g",
                  "b"
            ]
      },
      "videoSlope": {
            "title": "Video Slope",
            "type": "object",
            "properties": {
                  "b": {
                        "title": "Blue Video Slope",
                        "description": "Video slope for the blue channel",
                        "type": "number",
                        "minimum": 0.5,
                        "maximum": 2
                  },
                  "g": {
```

```
                        "title": "Green Video Slope",
                        "description": "Video slope for the green channel",
                        "type": "number",
                        "minimum": 0.5,
                        "maximum": 2
                    },
                    "r": {
                        "title": "Red Video Slope",
                        "description": "Video slope for the red channel",
                        "type": "number",
                        "minimum": 0.5,
                        "maximum": 2
                    }
                },
                "required": [
                    "r",
                    "g",
                    "b"
                ]
            }
        },
        "required": [
            "knee",
            "blackGamma",
            "gamma",
            "saturation",
            "saturationByHue",
            "videoSlope",
            "videoGamma",
            "videoPedestal"
        ]
}
```

C.6 Custom LUT3D Design

Name: Custom LUT3D Design
Cardinality: 0 or 1
Media Type: 'application/json'
Schema URI: https://www.arri.com/schema/json/camera/processing/custom_lut3d_design/v1-0-0

NOTE 1 Custom LUT3D Design Clip Descriptive Metadata can be encoded if a 3D LUT Set is present in header metadata and the contents of that 3D LUT Set were created by tools which opaquely encoded their parameters and control settings.

NOTE 2 If Custom LUT3D Design Clip Descriptive Metadata is present, the value of the lut3DID property will be the 3D LUT Identifier of a 3D LUT Set stored in header metadata.

C.6.1 Custom LUT3D Design Schema Contents (informative)

```
{
    "$schema": "http://json-schema.org/draft-07/schema#",
    "$id": "https://www.arri.com/schema/json/camera/processing/custom_lut3d_design/v1-0-0",
    "title": "Custom 3D LUT Design Data",
    "description": "Description of custom design data for a 3D LUT",
    "type": "object",
    "properties": {
        "lut3DDesignKey": {
            "title": "3D LUT Design Data Key",
            "description": "Key of the custom 3D LUT design data, in reverse domain name notation",
            "type": "string"
        },
        "lut3DDesignValue": {
            "title": "3D LUT Design Data Value",
            "description": "Opaque value of the custom LUT design data",
            "type": "string",
            "maxLength": 65536,
            "contentEncoding": "base64"
        },
```

```
            "lut3DFilename": {
                  "title": "3D LUT File Name",
                  "description": "Name of the original file that contains the 3D LUT data",
                  "type": "string"
            },
            "lut3DID": {
                  "title": "3D LUT Identifier",
                  "description": "Identifier of the 3D LUT, referenced by its design data and associated
CDL",
                  "type": "string",
                  "pattern": "^\\i\\c*$"
            }
      },
      "required": [
            "lut3DID",
            "lut3DDesignKey",
            "lut3DDesignValue"
      ]
}
```

C.7 Slate Info

Name: Slate Info
Cardinality: 1
Media Type: ‘application/json’
Schema URI: https://www.arri.com/schema/json/camera/slate_info/v1-1-0

C.7.1 Slate Info Schema Contents (informative)

```
{
        "$schema": "http://json-schema.org/draft-07/schema#",
        "$id": "https://www.arri.com/schema/json/camera/slate_info/v1-1-0",
        "title": "Slate Info",
        "description": "Descriptive metadata of the clip",
        "type": "object",
        "properties": {
                "cameraIndex": {
                        "title": "Camera Index",
                        "description": "User-defined two-letter camera index",
                        "type": "string",
                        "pattern": "^[A-Z][A-Z]$"
                },
                "cinematographer": {
                        "title": "Cinematographer",
                        "description": "Name of the director of photography / cinematographer",
                        "type": "string"
                },
                "circleTake": {
                        "title": "Circle Take",
                        "description": "Flag for marking good takes",
                        "type": "boolean"
                },
                "clipName": {
                        "title": "Clip Name",
                        "description": "Name of the clip",
                        "type": "string",
                        "pattern": "^([A-Z][0-9]{3}C[0-9]{3}_[0-9]{6}_[A-Z_][0-9A-Z]{3})|([A-Z][A-Z_][0-
9]{4}C[0-9]{3}_[0-9]{6}_[0-9]{6}_[ap][0-9A-Z]{4})$"
                },
                "clipNumber": {
                        "title": "Clip Number",
                        "description": "Clip number/counter within the reel",
                        "type": "integer",
                        "exclusiveMinimum": 0,
                        "maximum": 999
                },
                "director": {
                        "title": "Director",
                        "description": "Name of the director",
                        "type": "string"
```

```
		},
		"episode": {
			"title": "Episode",
			"description": "Name of the episode",
			"type": "string"
		},
		"location": {
			"title": "Location",
			"description": "Location of the shooting",
			"type": "string"
		},
		"production": {
			"title": "Production",
			"description": "Name of the production",
			"type": "string"
		},
		"productionCompany": {
			"title": "Production Company",
			"description": "Name of the production company",
			"type": "string"
		},
		"reelName": {
			"title": "Reel Name",
			"description": "Reel name",
			"type": "string",
			"pattern":	"^([A-Z][0-9]{3}[A-Z_][0-9A-Z]{3})|([A-Z][A-Z_][0-9]{4}_[0-9A-
Z]{4})$"
		},
		"reelNumber": {
			"title": "Reel Number",
			"description": "Reel number",
			"type": "integer",
			"exclusiveMinimum": 0
		},
		"scene": {
			"title": "Scene",
			"description": "Scene name",
			"type": "string"
		},
		"season": {
			"title": "Season",
			"description": "Name of the season / series",
			"type": "string"
		},
		"shootingDay": {
			"title": "Shooting Day",
			"description": "Day of shooting (counting)",
			"type": "integer",
			"exclusiveMinimum": 0
		},
		"take": {
			"title": "Take",
			"description": "Take number or name",
			"type": "string"
		},
		"userInfo": {
			"type": "array",
			"items": {
				"title": "User Info",
				"description": "Additional user-supplied information",
				"type": "object",
				"properties": {
					"key": {
						"title": "Key",
						"description": "Key of the custom user info, in reverse
domain name notation",
						"type": "string"
					},
					"value": {
						"title": "Value",
```

```
                                                "description": "Value of the custom user info",
                                                "type": "string"
                                        }
                                },
                                "required": [
                                        "key",
                                        "value"
                                ]
                        },
                        "minLength": 0
                }
        },
        "required": [
              "clipNumber",
              "clipName",
              "cameraIndex"
        ]

}
```

C.8 Recording Medium

Name: Recording Medium
Cardinality: 1
Media Type: ‘application/json’
Schema URI: https://www.arri.com/schema/json/camera/recording_medium/v1-0-0

C.8.1 Recording Medium Schema Contents (informative)

```
{
        "$schema": "http://json-schema.org/draft-07/schema#",
        "$id": "https://www.arri.com/schema/json/camera/recording_medium/v1-0-0",
        "title": "Recording Medium",
        "description": "Information about the recording medium",
        "type": "object",
        "properties": {
                "mediumCapacity": {
                        "title": "Medium Capacity",
                        "description": "Nominal capacity of the recording medium in MB",
                        "type": "integer",
                        "exclusiveMinimum": 0
                },
                "mediumFirmwareVersion": {
                        "title": "Medium Firmware Version",
                        "description": "Firmware version number of the recording medium",
                        "type": "string",
                        "pattern": "^\\d+\\.\\d+\\.\\d+(\\-[A-Za-z][A-Za-z0-9\\.\\-]*)?(\\+\\d+)?$"
                },
                "mediumManufacturer": {
                        "title": "Medium Manufacturer",
                        "description": "Manufacturer of the recording medium",
                        "type": "string"
                },
                "mediumModelName": {
                        "title": "Medium Model Name",
                        "description": "Model of the recording medium",
                        "type": "string"
                },
                "mediumSerialNumber": {
                        "title": "Medium Serial Number",
                        "description": "Serial number of the recording medium",
                        "type": "string"
                },
                "mediumType": {
                        "title": "Medium Type",
                        "description": "Type of the recording medium",
                        "type": "string",
                        "oneOf": [
                                {
                                        "type": "string",
```

```
                                        "const": "CFast 2.0 Card"
                                },
                                {
                                        "type": "string",
                                        "const": "Codex Capture Drive"
                                },
                                {
                                        "type": "string",
                                        "const": "Codex Compact Drive"
                                }
                        ]
                }
        },
        "required": [
                "mediumType"
        ]
}
```

Bibliography (informative)

SMPTE RDD 31:2014, Deferred Demosaicing of an ARRIRAW Image File to a Wide-Gamut Logarithmic Encoding

SMPTE ST 400:2012, SMPTE Labels Structure

SMPTE ST 429-14:2014, D-Cinema Packaging — Aux Data Track File

D. Reisner, J. Pines, “The ASC Color Decision List” in American Cinematographer Manual, 11th Ed. (D. Mullen and R. Hummel, Eds.), Hollywood, Calif: ASC Press, 2021, pp. 210-224.

JSON Schema: A Media Type for Describing JSON documents, (A. Wright, M. Andrews, B. Hutton, Eds.), 28 January 2020, at https://datatracker.ietf.org/doc/html/draft-bhutton-json-schema-00

JSON Schema Validation: A Vocabulary for Structural Validation of JSON, (A. Wright, M. Andrews, B. Hutton, Eds.), 8 December 2020, at https://datatracker.ietf.org/doc/html/draft-bhutton-json-schema-validation-00