INTERNATIONAL STANDARD

ISO/IEC 19566-7

First edition
2022-10

Information technologies — JPEG systems —

Part 7: JPEG linked media format (JLINK)

Technologies de l'Information — Systèmes JPEG —

Partie 7: Format de média de liaison JPEG (JLINK)

Reference number
ISO/IEC 19566-7:2022(E)

ISO copyright office
CP 401 • Ch. de Blandonnet 8
CH-1214 Vernier, Geneva
Phone: +41 22 749 01 11
Email: copyright@iso.org
Website: www.iso.org

Published in Switzerland

Contents

Page

Foreword ..... **iv**
Introduction ..... **v**
1 Scope **1**
2 Normative references **1**
3 Terms and definitions **1**
4 Abbreviations **2**
5 General **2**
5.1 JLINK concept 2
5.2 Description of a user experience 3
6 Components of a scene **5**
6.1 Image 5
6.2 Viewport 5
6.3 Title and note 6
7 Components of a link **6**
7.1 Reference to a destination scene 6
7.2 Linkage region on 2D image of the source scene 6
7.3 Viewport ID on showing 2D image of the destination scene 8
8 Sprites **8**
9 Moving between scenes **8**
9.1 Jump-in effect 8
9.2 Jump-out effect 10
10 Structuring of JLINK metadata **12**
10.1 General 12
10.2 Definition of JUMBF content type for JLINK 12
10.3 Structure by JUMBF boxes for JLINK 12
10.4 File position for JLINK metadata and linked media 13
Annex A (normative) **JUMBF content type for JLINK** **15**
Annex B (normative) **Metadata for JLINK** **16**
Annex C (informative) **Example implementations for scene changes** **26**
Bibliography **30**

Foreword

ISO (the International Organization for Standardization) and IEC (the International Electrotechnical Commission) form the specialized system for worldwide standardization. National bodies that are members of ISO or IEC participate in the development of International Standards through technical committees established by the respective organization to deal with particular fields of technical activity. ISO and IEC technical committees collaborate in fields of mutual interest. Other international organizations, governmental and non-governmental, in liaison with ISO and IEC, also take part in the work.

The procedures used to develop this document and those intended for its further maintenance are described in the ISO/IEC Directives, Part 1. In particular, the different approval criteria needed for the different types of document should be noted. This document was drafted in accordance with the editorial rules of the ISO/IEC Directives, Part 2 (see www.iso.org/directives or www.iec.ch/members_experts/refdocs).

Attention is drawn to the possibility that some of the elements of this document may be the subject of patent rights. ISO and IEC shall not be held responsible for identifying any or all such patent rights. Details of any patent rights identified during the development of the document will be in the Introduction and/or on the ISO list of patent declarations received (see www.iso.org/patents) or the IEC list of patent declarations received (see https://patents.iec.ch).

Any trade name used in this document is information given for the convenience of users and does not constitute an endorsement.

For an explanation of the voluntary nature of standards, the meaning of ISO specific terms and expressions related to conformity assessment, as well as information about ISO's adherence to the World Trade Organization (WTO) principles in the Technical Barriers to Trade (TBT) see www.iso.org/iso/foreword.html. In the IEC, see www.iec.ch/understanding-standards.

This document was prepared by Joint Technical Committee ISO/IEC JTC 1, *Information technology*, Subcommittee SC 29, *Coding of audio, picture, multimedia and hypermedia information*.

A list of all parts in the ISO/IEC 19566 series can be found on the ISO and IEC websites.

Any feedback or questions on this document should be directed to the user's national standards body. A complete listing of these bodies can be found at www.iso.org/members.html and www.iec.ch/national-committees.

Introduction

This document describes the JPEG Linked Media Format (JLINK) international standard, which enables the embodiment of multiple image types and media elements into a single media content. This document elaborates on the inherent properties and functionalities of the file format, such as file structure and navigation.

This document defines the image container for structuring multiple types of media into a single file, including definition of metadata specification for multiple types of media. It supports legacy technology in the domain, such as image coding technology as well as metadata standards that signal access policies and others.

In this document, the following verbal forms are used:

— “shall” indicates a requirement;

— “should” indicates a recommendation;

— “may” indicates a permission;

— “can” indicates a possibility or a capability.

Information marked as “NOTE” is intended to assist the understanding or use of the document. “Notes to entry” used in Clause 3 provide additional information that supplements the terminological data and can contain provisions relating to the use of a term.

Information technologies — JPEG systems —

Part 7:
JPEG linked media format (JLINK)

1 Scope

This document specifies an image file format capable of linking multiple media elements, such as image and text in any box-based JPEG file format.

2 Normative references

The following documents are referred to in the text in such a way that some or all of their content constitutes requirements of this document. For dated references, only the edition cited applies. For undated references, the latest edition of the referenced document (including any amendments) applies.

ISO/IEC 19566-5, *Information technologies — JPEG systems — Part 5: JPEG universal metadata box format (JUMBF)*

3 Terms and definitions

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

ISO and IEC maintain terminology databases for use in standardization at the following addresses:

— ISO Online browsing platform: available at https://www.iso.org/obp

— IEC Electropedia: available at https://www.electropedia.org/

3.1
bitstream
sequence of bits comprising an image file

3.2
codestream
collection of one or more bit streams and the main header, tile-part headers, and the EOC required for their decoding and expansion into image data

Note 1 to entry: This is the image data in a compressed form with all of the signalling needed to decode.

[SOURCE: ISO/IEC 15444-1:2019, 3.19]

3.3
link
relational description of a scene to another scene composed of linkage region, sprite, and visual effect for scene change

3.4
linkage region
specific region in a source scene to which a link is active

3.5
metadata
data about data

[SOURCE: ISO 19115:2003]

3.6
point of origin
reference location (e.g. 0,0) used to describe offset positions

3.7
scene
basic unit of a JLINK composed of a 2D image and text

3.8
sprite
visual element on the user display which indicates user interaction events

4 Abbreviations

FOV	field of view
JLINK	JPEG linked media format
JPEG	joint photographic experts group
JUMBF	JPEG universal metadata box format
URI	uniform resource identifier
XML	extensible markup language
XMP	extensible metadata platform
RDF	resource description framework
UMF	universal metadata framework

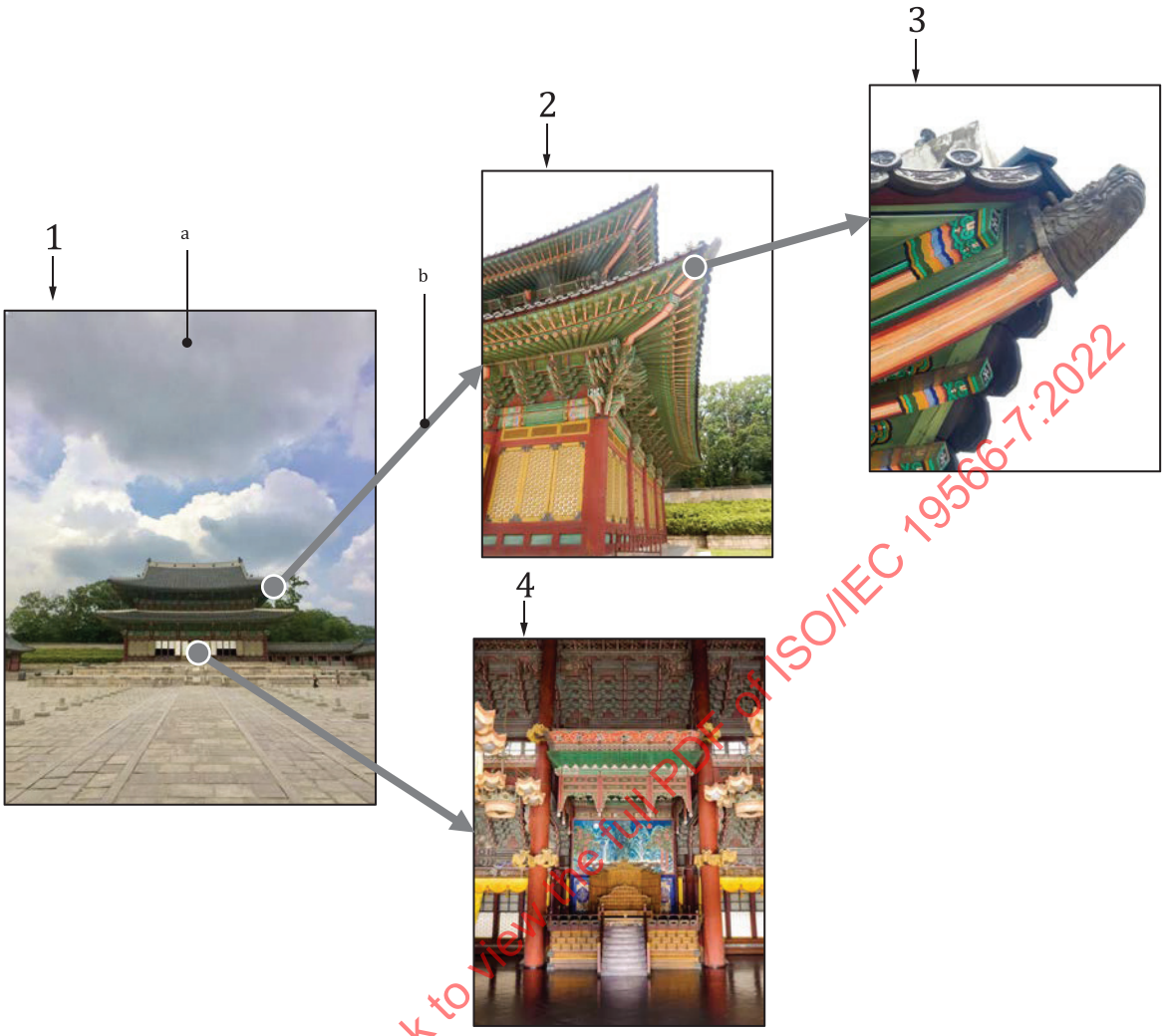
5 General

5.1 JLINK concept

JLINK structures sets of 2D images and text into a theme-related content. The basic elements of JLINK are a 'scene' and a 'link'. A scene contains 2D image (or portions thereof) and other elements such as an image title, an image note, and image viewports.

A link describes the way(s) scenes are associated, and it contains a description of interactive region of interest, overlaid sprites and their locations on the image, and visual effect between scene renderings. Combining scenes and links creates curated multimedia experience of the images.

Figure 1 shows a JLINK where scene0 has links to scene1 and scene3 and scene1 to scene2 (Figure 1). scene0 and scene1 have specific linkage regions to scene1, scene3, and scene2, respectively, with the location that originated from the top left (Figure 4).



Key

1 scene0
2 scene1
3 Scene2
4 scene3
a A scene contains image, title, note, and viewports.
b A link contains linkage region, sprite, and scene change.

Figure 1 — An example of JLINK image with four linked scenes

5.2 Description of a user experience

As a typical usage example for the JLINK shown as Figure 1, a viewer software shall provide the following user experience (Figure 2):

a) Presentation of a scene and sprites

— An image of any scene (e.g., scene0) is presented to a viewport. When the presented scene (scene0) is linked to the other scenes (scene1 and scene2), the sprite corresponding to each link is placed at each linkage region (see 7.2) on the image of the presented scene.

b) Selection of a sprite

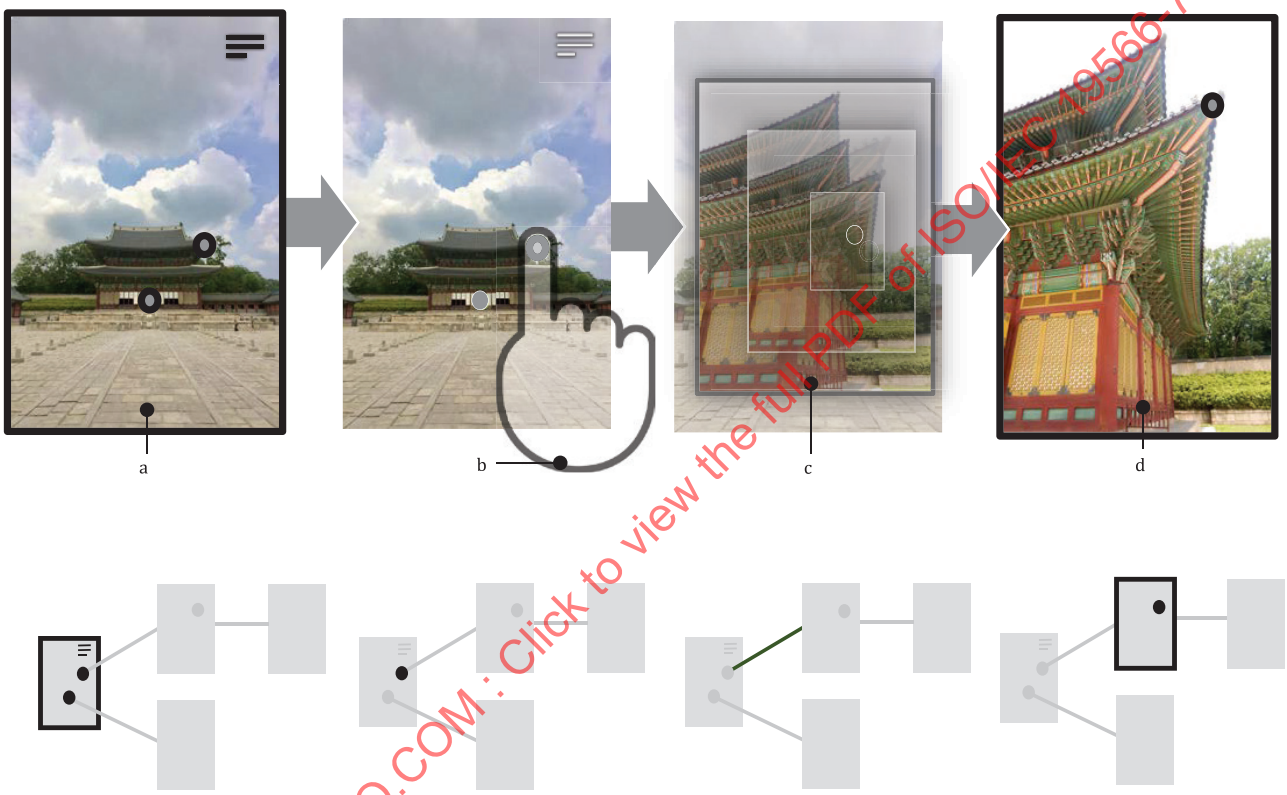
— Sprites placed on the scene receive user interface events. Image 2 in Figure 2 shows an example that one of the sprites receives a user interface event (selection).

c) Visual effect for scene change and preparing subsequent scene

— When a sprite receives a user interface event, the screen transitions to another scene (scene1) that is the destination of the corresponding link.

d) Presentation of the destination scene and sprites

— The image of the destination scene (scene1) is presented in the same way as described in '1. Presentation of a scene and sprites.'



a The viewer software presents a scene with sprite, title and note.

b The user selects a sprite.

c The viewer software presents the visual effect for scene change and preparing the destination scene.

d The viewer software presents the destination scene and sprites.

Figure 2 — An example of the JLINK user experience

As a typical usage example for title and note associated with a scene, a viewer software shall provide the following user experience (Figure 3).

a) Presentation of a scene with a title and note

— An image of any scene (scene0) is presented. When the presented scene has text information, *i.e.*, title or/and note, the viewer software places an icon (text button) on the scene to indicate that text information is available.

b) Selection of the text button

— The text button placed on the scene receives user interface events.

c) Presentation of a title and note

— When the text button receives a user event, the text window appears with the scene associated title and note.

a The viewer software presents a scene with sprite, title and note.

b The user selects the text button.

c The viewer software presents the title and note.

Figure 3 — An example of the scene associated title and text viewing process

6 Components of a scene

6.1 Image

A scene has an image component to display a 2D still image on the screen. In accordance with ISO/IEC 19566-5, a JUMBF URI reference to the box where the corresponding image codestream is described. Values to the components shall be found in Annex B.

6.2 Viewport

A scene has optional viewports which are used to define specific regions of interest to be displayed. When the media to be presented on the screen is a 2D image, the viewport shall be described by:

— X coordinate of the center of the region as a percent ratio for the image of the scene ,

— Y coordinate of the center of the region as a percent ratio for the image of the scene,

— Field of view span for the X-axis as a percent ratio for the image of the scene, and

— Field of view span for the Y-axis as a percent ratio for the image of the scene.

Multiple viewports are described for each scene, and each viewport is identified by an ID. An ID is an integer starting at 1; ID value of 0 is reserved as described in Annex B, Table B.1. As described in subclause 7.3, it is specified which viewport to use through the viewport ID value in the link to the scene. If viewports are not described, the complete image is used as the viewport with default ID equal to 0. Values for the components shall be found in Annex B.

6.3 Title and note

A scene has text information such as title and note. If the title and note are described, the viewer software shall provide a user interface as shown in subclause 5.2. The properties of the title and note are defined in Annex B.

7 Components of a link

7.1 Reference to a destination scene

In accordance with ISO/IEC 19566-5, a JUMBF URI reference to the box where the destination scene is described. The properties of the reference are defined in Annex B.

7.2 Linkage region on 2D image of the source scene

A link is associated to a linkage region on the scene which expresses the specific part of the source scene to which a link is active. (Figure 4). A linkage region has a shape such as point, rectangle, and ellipse. The properties of the linkage region are defined in Annex B.

A linkage region as a point of a 2D image shall be described by:

— X coordinate of the center of the region as a percent ratio for the image of the scene and

— Y coordinate of the center of the region as a percent ratio for the image of the scene.

A linkage region as rectangle or ellipse on a 2D image shall be described by:

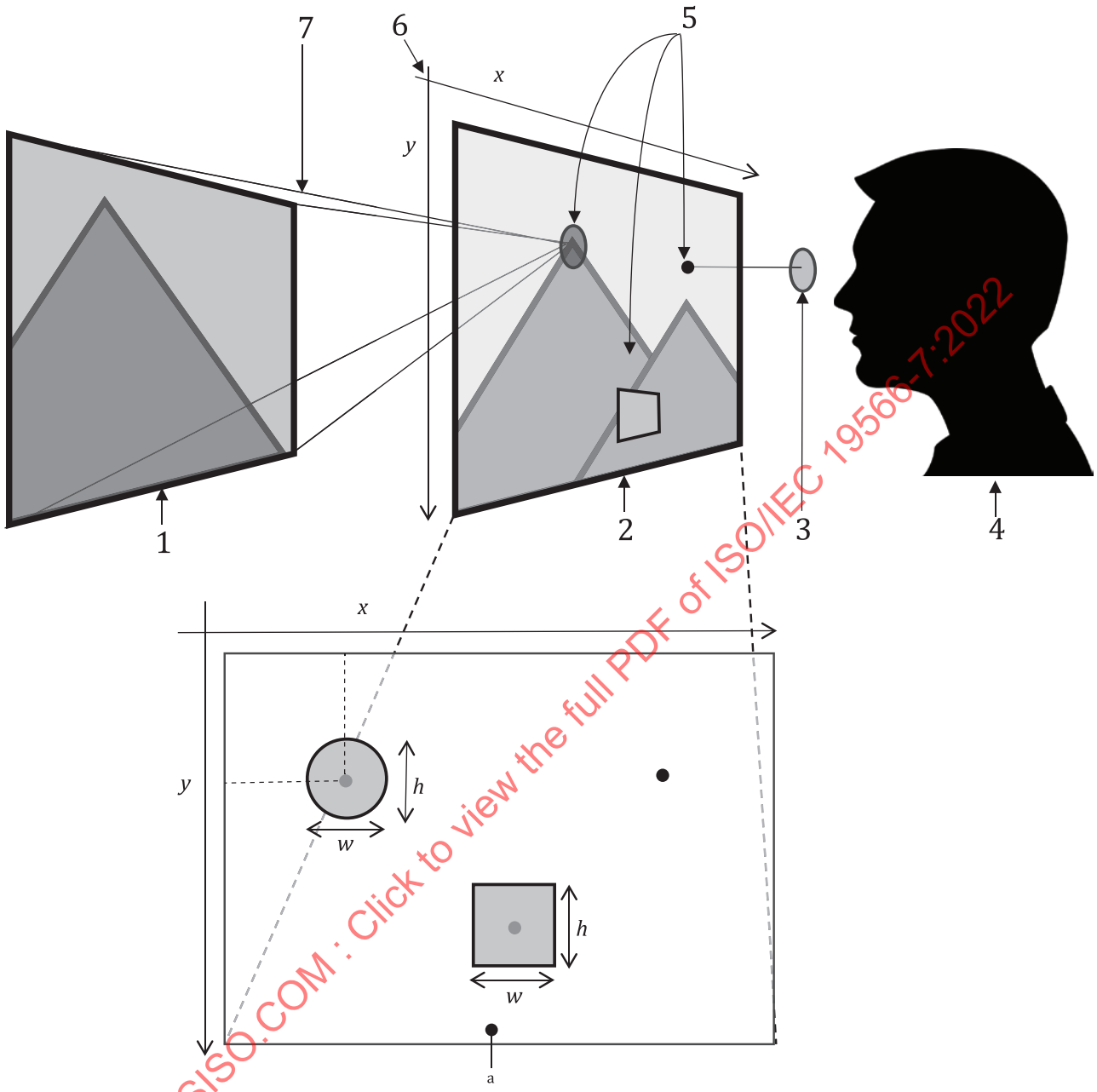
— X coordinate of the center of the region as a percent ratio for the image of the scene,

— Y coordinate of the center of the region as a percent ratio for the image of the scene,

— Width of the region as a percent ratio for the image of the scene,

— Height of the region as a percent ratio for the image of the scene, and

— Rotation angle (°) in the positive direction of the X axis is 0 ° and the counterclockwise direction in the positive direction.



Key

1 subsequent scene
2 scene
3 sprite
4 viewer
5 linkage region
6 point of origin
7 scene change
w width
h height
[a] A linkage region has a shape such as point, rectangle, and ellipse.

Figure 4 — Linked scenes of 2D images through linkage point

7.3 Viewport ID on showing 2D image of the destination scene

When navigating from one scene to another through a link, it selects which viewport of the destination scene to be used.

If the viewport ID is not described or set as 0, the complete image is used as the viewport. The properties of the viewport ID are defined in Annex B.

8 Sprites

When a scene and a scene are connected through a link, sprite information are described in the link in order to present to a user interaction means that move from one location of a scene to another. The viewer software shall locate the center of the sprite on the center point of the linkage region. The properties of the sprite are defined in Annex B.

9 Moving between scenes

When navigating from one scene (source scene) to another (destination scene) through a link, the viewer software shall provide the visual effect during moving from the source scene to the destination scene. This effect is called 'jump-in,' which is defined in subclause 9.1.

The viewer software shall provide backward navigation through a link from the destination scene. For backward navigation, the viewer software shall provide the visual effect called 'jump-out,' which is defined in subclause 9.2.

9.1 Jump-in effect

Jump-in effect is the visual transition for navigation forward through a link from the source scene to the destination scene.

Before the jump-in effect begins, the image of the source scene is displayed and the sprites corresponding to each link are placed on the image.

Jump-in effect proceeds as follows:

1) All the sprites on the source scene immediately become invisible and their actions are disabled.

2) The destination scene of the jump-in link becomes visible on the source scene in the given viewport, at 0 % scale placed at the center point of the corresponding linkage region.

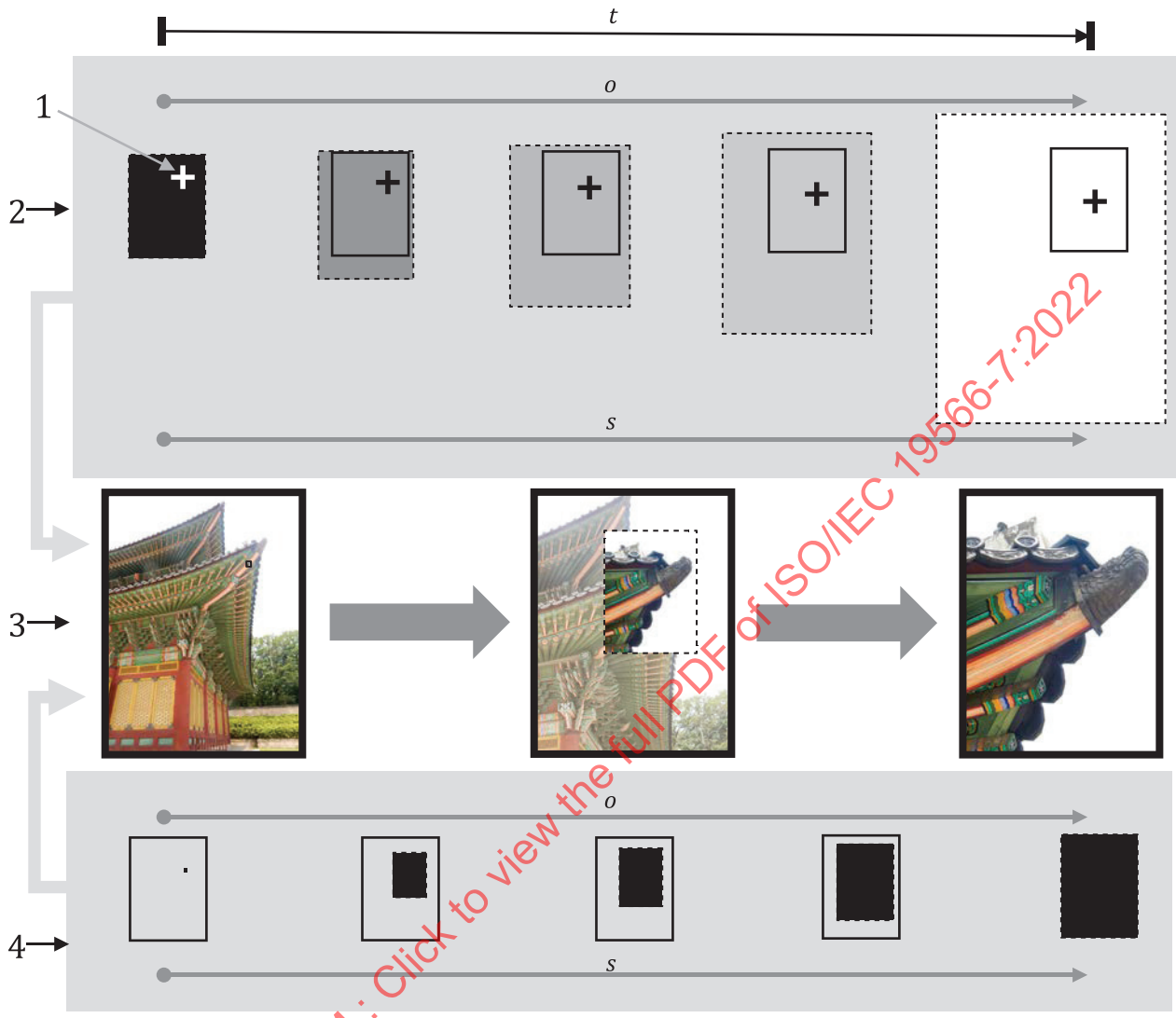
3) The source scene and the destination scene are animated simultaneously.

   (Figure 5):

   i. The source scene is animated to be shown at more than 100 % scale (as defined by the viewer software; a recommended scaling value is 300 %). The source scene fading away to 0 % opacity, and shifting the user's view to put the jump-in linkage point at the center of the viewport.

   ii. The destination scene is animated to the state to be shown after the effect at scale and position (expected to fit to the window).

4) The destination scene sprites become visible and their actions are enabled.



	t	o	s
2	0 ~ 600	100 ~ 0	100 ~ 300
4	0 ~ 600	100 ~ 100	0 ~ 100

Key

1 the linkage point of source scene
2 source scene
3 combination
4 destination scene
t time in milliseconds
o opacity in percent
s scale in percent

Figure 5 — Jump-in visual effects of the source scene and destination scene

The animation of step 3 above is a combination of zooming-in both the source and destination scene and fading out the source scene.

During the animation of step 3, changes in values such as scale, opacity, and position should be smooth and monotonic.

The duration of step 3 default 600ms.

Annex C.2 describes an example of expected implementation.

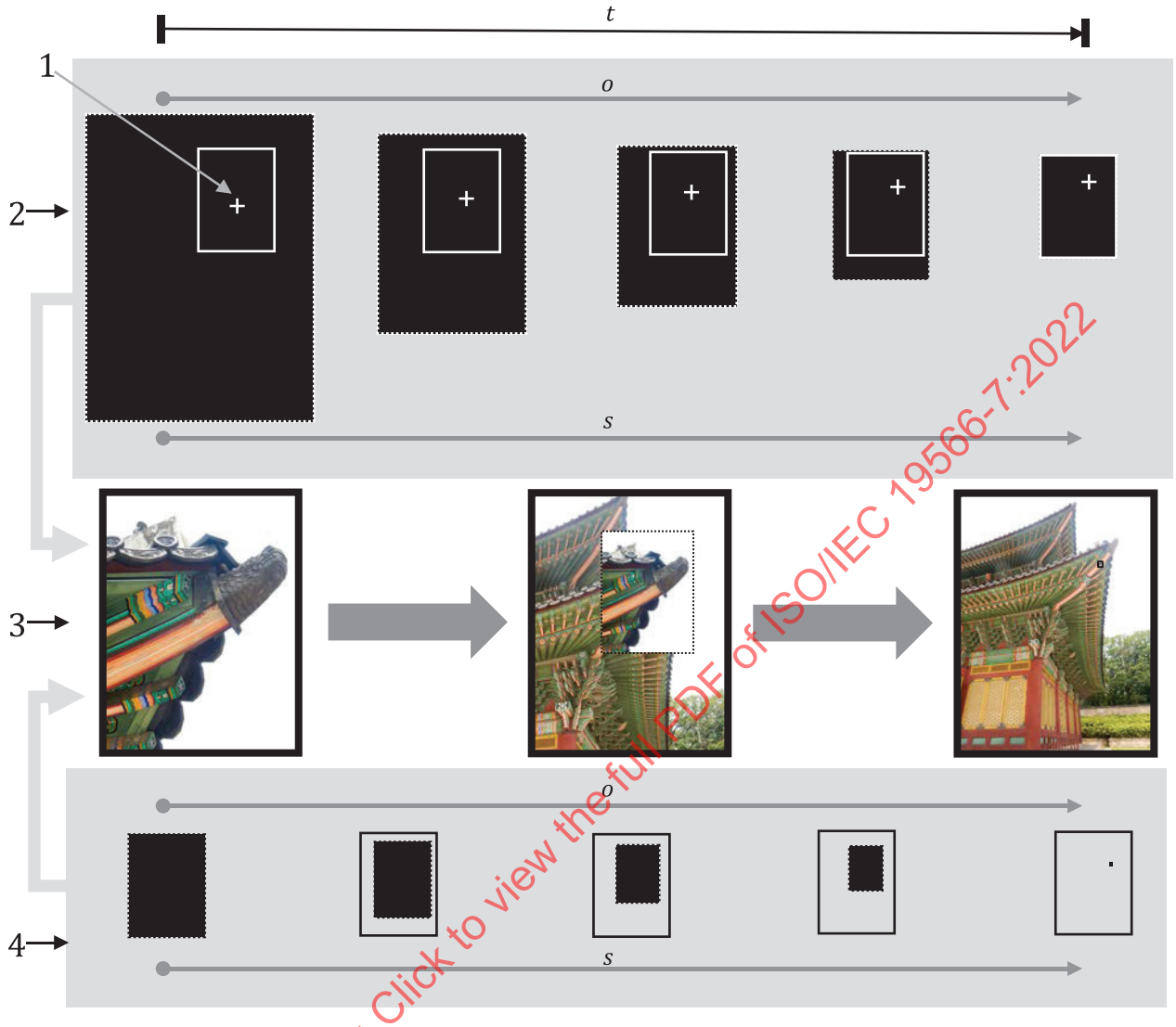
9.2 Jump-out effect

Jump-out effect is the visual transition for navigation backward through a link from the destination scene to the source scene.

Before the effect begins, the image of the destination scene is displayed and the sprites corresponding to each link are placed on the image.

Jump-out effect proceeds as follows:

1) All the sprites on the destination scene immediately become invisible and their actions are disabled.
2) The source scene of the jump-out link becomes visible as the background of the destination scene in the given viewport, at more than 100 % scale (as defined by the viewer software, a recommended scaling value is 300 %) and putting the center of the linkage region at the center of the window.
3) The destination scene and the source scene are animated simultaneously (Figure 6):
    i. The destination scene is animated to be shown at 0 % scale, moving to the center point of the corresponding linkage region.
    ii. The source scene is animated to the state to be shown after the effect at scale and position (expected to fit to the window).
4) The destination scene becomes invisible.
5) The source scene sprites become visible, and their actions are enabled.



	t	*o*	*s*
2	0 ~ 600	100 ~ 100	300 ~ 100
4	0 ~ 600	100 ~ 100	100 ~ 0

Key

1 the linkage point of source scene
2 source scene
3 combination
4 destination scene
t time in milliseconds
o opacity in percent
s scale in percent

Figure 6 — Jump-out visual effects of the source scene and destination scene

The animation step 3 above is a combination of zooming-out both the destination scene and the source scene.

During the animation of step 3, changes in values such as scale and position should be smooth and monotonic.

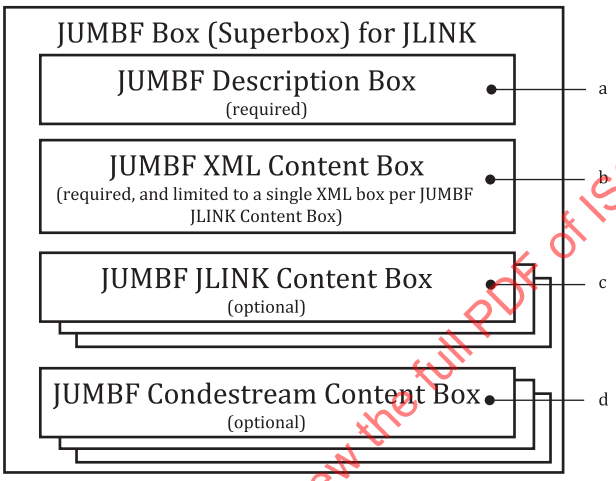
The duration of step 3 default 600ms.

Annex C.3 describes an example of expected implementation.

10 Structuring of JLINK metadata

10.1 General

This clause defines the JLINK Content Type which is based on the JUMBF superbox defined by ISO/IEC 19566-5. The sub-box components are defined, which include the definition of an XML box, the use of nested JLINK Content Type boxes, and the use of Codestream Content Type boxes for images which are associated to scenes or sprites. An overview of the JLINK Content Type box is shown in Figure 7.



a JUMBF Description box contains type of metadata or application.

b JUMBF XML box contains XML metadata.

c JUMBF boxes for JLINK contain nested JUMBF boxes for JLINK.

d JUMBF Codestream boxes contain image codestreams for scenes and sprites.

Figure 7 — Example box diagram of JLINK Content Type

10.2 Definition of JUMBF content type for JLINK

In accordance with ISO/IEC 19566-5, the JLINK Content Type shall be as defined in:

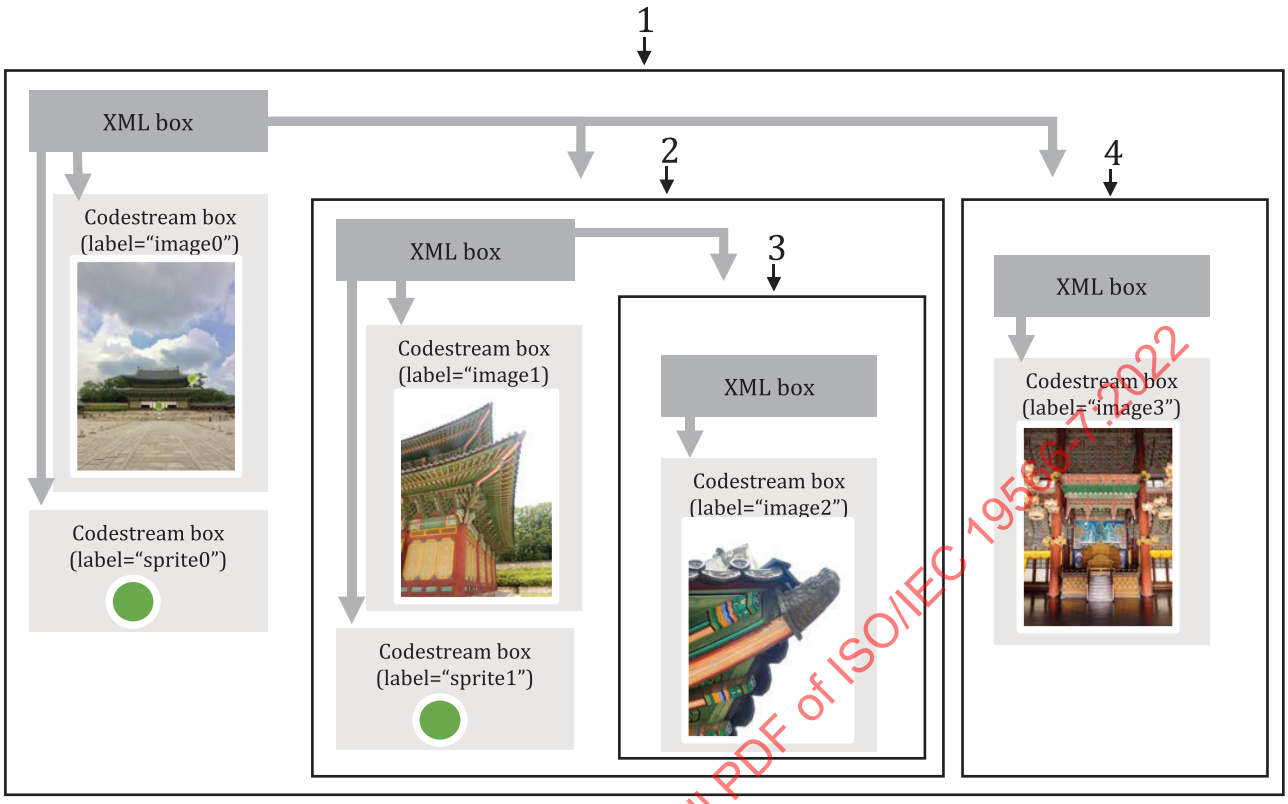
— Annex A of this document, which contains the definition of the JUMBF box for JLINK (e.g, the structure and definitions of required content of this box).

— Annex B of this document, which contains the definition of the JLINK metadata schema.

10.3 Structure by JUMBF boxes for JLINK

JLINK box stores the metadata and codestream that define the scene. The metadata of a scene possesses link information that connects to another scene, which is described in a way that points to the JLINK address to which the destination scene to be accessed belongs.

It is recommended that the JUMBF box for the destination scene is embedded within the JUMBF box for the source scene. If necessary, the JUMBF box for the destination scene is located out of the JUMBF box for the source scene, even in the other file. Figure 8 shows an example that JUMBF boxes for JLINK are achieved layered structure.



Key

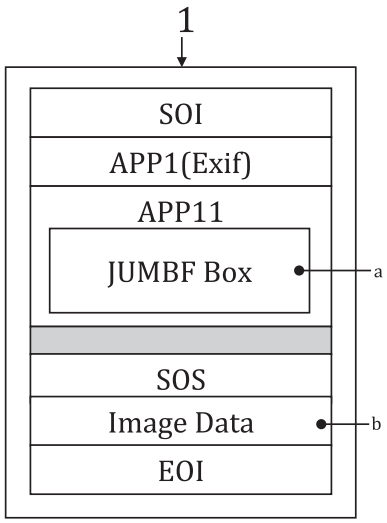
1 JUMBF box for JLINK whose label is "scene0"
2 JUMBF box for JLINK whose label is "scene1"
3 JUMBF box for JLINK whose label is "scene2"
4 JUMBF box for JLINK whose label is "scene3"

Figure 8 — Layered structure by JUMBF box for JLINK

10.4 File position for JLINK metadata and linked media

Embedding a JLINK Content Type box shall be done in accordance with ISO/IEC 19566-5:2019, Annex D.

NOTE ISO/IEC 19566-5 defines specific locations in the file structure for an image codestream. With the new usages supported by JLINK, images are included in a broader definition of metadata. It is possible to signal that the encoded image is located within JLINK Content Type box or is located in the file position of earlier standard. A simplified diagram of the file structure is shown in Figure 9. This "legacy" file position is desirable so that an image is decoded when the file is opened by conventional JPEG viewing applications. For additional information, refer to ISO/IEC 19566-5.



Key

1 JPEG-1 bitstream

a JLINK metadata and codestreams are located within JUMBF box.

b A single codestream which is compatible with conventional viewers is located after the SOS marker.

Figure 9 — Example of file position for JLINK metadata and codestreams within JUMBF box

Annex A
(normative)

JUMBF content type for JLINK

A.1 General

This annex defines the use of JUMBF for JLINK.

A.2 JUMBF description box: type for JLINK

Table A.1 — JLINK metadata elements

Parameter	Value
TYPE	0x4c494e4b -0011-0010-8000-00AA00389B71

Annex B
(normative)

Metadata for JLINK

B.1 General

This annex defines the components and usage of JLINK metadata.

B.2 Definition of JLINK metadata

The schema elements of the JLINK Metadata contain basic properties, described in Table B.1.

Table B.1 — JLINK metadata elements

JLINK element names				Data type	Default value if not this present	Definition
Scene				string	Not applicable	Schema name for scene
	Version			string	"1.0.0"	Schema version number specified by Semantic Versioning 2.0.0[1]
	Title			string	""	Title of the scene
	Note			string	""	Description of the scene
	Viewport			string	Not applicable	Name of subschema for viewport of the scene
		X		real	50	Horizontal center location of this viewport in percent
		Y		real	50	Vertical center location of this viewport in percent
		XFOV		real	100	Horizontal field-of-view of this viewport in percent
		YFOV		real	100	Vertical field-of-view of this viewport in percent
		ID		integer	Not applicable	Viewport ID for this viewport, ID start with one
	Image			string	Not applicable	Name of subschema for image component of the scene
		Format		string	Not applicable	Media Type of this image
		Href		string	Not applicable	JUMBF URI reference to the codestream of this image
	Link			string	Not applicable	Name of subschema for link
		Region		string	rectangle	Name of subschema for the linkage region on the on the source scene
			Shape	string	rectangle	Shape type of this linkage region
			X	real	Not applicable	x position of center of this linkage region in percent
			Y	real	Not applicable	y position of center of this linkage region in percent
			W	real	Not applicable	Width of this linkage region in percent

Table B.1 *(continued)*

JLINK element names				Data type	Default value if not this present	Definition
			H	real	Not applicable	Height of this linkage region in percent
			Rotation	real	Not applicable	Rotation angle of this linkage region in degree
		Duration		integer	600	Duration of the animation for scene change in milliseconds
		VPID		integer	0	Viewport ID on showing 2D image of the destination scene, vpid=0 for a whole region
		Sprite		string	Not applicable	JUMBF URI reference to the image codestream of the sprite
		To		string	Not applicable	JUMBF URI reference to the destination scene

B.3 Representation of metadata

This Annex defines JLINK schema based on XMP specification which is also covered as part of ISO 16684-1. The data source layer implements serialization, deserialization and embedding of metadata with using XMP with the addition of new tags to express a schema descriptor, and to associate metadata elements with that schema descriptor.

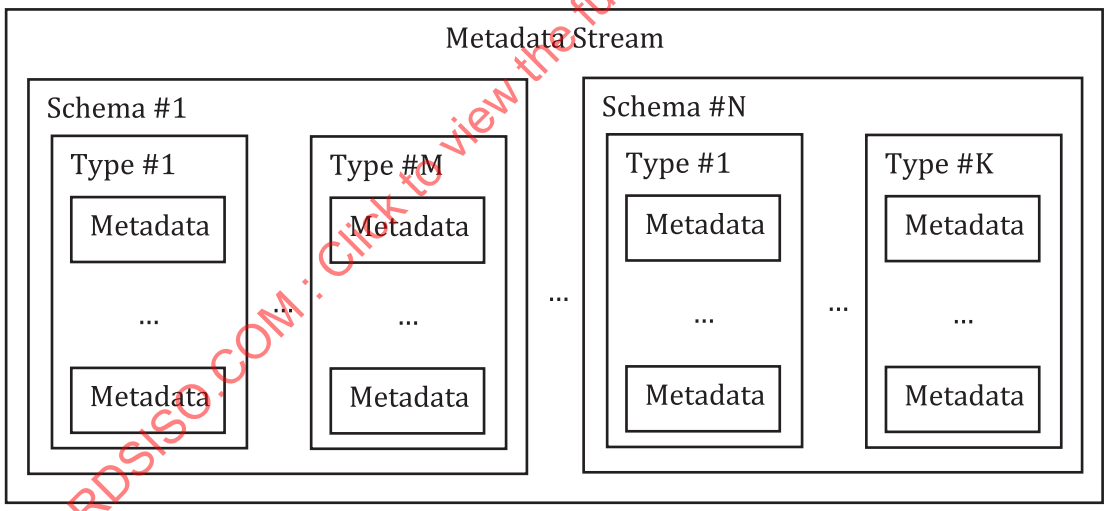


Figure B.1 — Logical view of the metadata

The high-level structuring of the JLINK XML box is shown in Table B.2.

Table B.2 — The high-level structuring of the JLINK XML box

XMP framing	Purpose	Top level structure and tags	Second level structure and tags
	Internal counter	umf:next-id	
	Schemas description storage	XMP arrays of schemas umf:schemas	XMP array of schema names umf: descriptors
			XMP array of fields/types umf:fields

Table B.2 *(continued)*

XMP framing	Purpose	Top level structure and tags	Second level structure and tags
	Metadata storage	XMP array of metadata umf:metadata	XMP array of metadata items umf:set
			XMP array of metadata field/values, references umf:fields, umf:refs

It is to be understood that the top-level structure ordering is not critical as the structure shown in Table B.3 and equivalent to the structure in Table B.2.

Table B.3 — The structuring equivalent to Table B.2

XMP framing	Purpose	Top level structure and tags	Second level structure and tags
	Internal counter	umf:next-id	
	Metadata storage	XMP array of metadata umf:metadata	XMP array of metadata items umf:set
			XMP array of metadata field/values, references umf:fields, umf:refs
	Schemas description storage	XMP arrays of schemas umf:schemas	XMP array of schema names umf:descriptors
			XMP array of fields/types umf:fields

B.4 JLINK metadata syntax and schema descriptors for XMP

The JLINK schema is serialized and stored using a subset of the W3C Resource Description Framework (RDF), expressed in XML.

The XMP statement to frame a Metadata Stream is:

```
<x:xmpmeta xmlns:x="adobe:ns:meta/" x:xmptk="XMP Core 5.4.0">
   <rdf:RDF xmlns:rdf="http://www.w3.org/1999/02/22-rdf-syntax-ns#">
      <rdf:Description rdf:about="" xmlns:umf="http://ns.intel.com/umf/2.0">
      <!- internal structure -->
      </rdf:Description>
   </rdf:RDF>
</x:xmpmeta>
```

The schema descriptor elements of JLINK metadata are shown in Table B.2.

Table B.4 — Schema descriptor of JLINK metadata elements

JLINK schema descriptor				Data type	Schema descriptor
Scene				string	<umf:schema>Scene</umf:schema>
	Version			integer	<umf:name>Version</umf:name> <umf:type>integer</umf:type>
	Title			string	<umf:name>Title</umf:name> <umf:type>string</umf:type>
	Note			string	<umf:name>Note</umf:name> <umf:type>string</umf:type>
	Viewport			string	<umf:schema>Viewport</umf:schema>
		X		real	<umf:name>X</umf:name> <umf:type>real</umf:type>
		Y		real	<umf:name>Y</umf:name> <umf:type>real</umf:type>
		XFOV		real	<umf:name>XFOV</umf:name> <umf:type>real</umf:type>

Table B.4 *(continued)*

JLINK schema descriptor				Data type	Schema descriptor
		YFOV		real	<umf:name>YFOV</umf:name> <umf:type>real</umf:type>
		ID		integer	<umf:name>ID</umf:name> <umf:type>integer</umf:type>
	Image			string	<umf:schema>Image</umf:schema>
		Format		string	<umf:name>Format</umf:name> <umf:type>string</umf:type>
		Href		string	<umf:name>Href</umf:name> <umf:type>string</umf:type>
	Link			string	<umf:schema>Link</umf:schema>
		Region		string	<umf:schema>Region</umf:schema>
			Shape	string	<umf:name>Shape</umf:name> <umf:type>string</umf:type>
			X	real	<umf:name>X</umf:name> <umf:type>real</umf:type>
			Y	real	<umf:name>Y</umf:name> <umf:type>real</umf:type>
			W	real	<umf:name>W</umf:name> <umf:type>real</umf:type>
			H	real	<umf:name>H</umf:name> <umf:type>real</umf:type>
			Rotation	real	<umf:name>Rotation</umf:name> <umf:type>real</umf:type>
		Duration		integer	<umf:name>Duration</umf:name> <umf:type>integer</umf:type>
		VPID		integer	<umf:name>VPID</umf:name> <umf:type>integer</umf:type>
		Sprite		string	<umf:name>Sprite</umf:name> <umf:type>string</umf:type>
		To		string	<umf:name>To</umf:name> <umf:type>string</umf:type>

B.5 Reserved tags

Table B.5 defines tags reserved in the JLINK XML Box.

The keyword "reserved" indicates a provision that is not specified at this time, shall not be used, and may be specified in the future.

Table B.5 — JLINK reserved XML tags

Reserved tag	Description
umf:schemas	Tag to define start of schema descriptor. Multiple schema descriptors are defined, each with multiple sets of metadata elements.
umf:schema	Tag to define a unique name for a schema.
umf:name	Tag to define name of schema descriptor elements, or to label sets of metadata elements.
umf:type	Tag to associate a data type to a name in the schema descriptor.
umf:id	Unique id of a metadata set.
umf:next-id	Used by implementation(s) to assign a unique id to metadata items.

Table B.5 *(continued)*

Reserved tag	Description
umf:index	Reserved tag. Index into an image sequence that contains the first image associated with this metadata item (-1 in case of global metadata).
umf:nframes	Reserved tag. Number of sequential frames the metadata item is associated with (0 in case of global metadata).
umf:fields	Tag for description of schema fields. Fields of metadata items provide a list of value/name pairs (in case it is a structure) or an array of values (in case it is an array) or just a single value.
umf:refs	Array of references to other metadata items. Each reference is stored as metadata id.
umf:set	Tag to associate fields of metadata elements.

B.6 XMP expression of minimum self-describing schema (without metadata elements)

The JLINK XML Box contains data which is structured using XMP; the XMP expression of the schema defines its properties.

The minimal XMP expression consists of the JLINK schema descriptor and an unpopulated JLINK metadata storage as shown in the XMP expression of minimum self-describing schema presented in this section. When this minimal expression is provided, a number of default values for the JLINK description are assigned; these default values are as specified in Table B.1.

```
<x:xmpmeta xmlns:x="adobe:ns:meta/" x:xmptk="XMP Core 5.5.0">
  <rdf:RDF xmlns:rdf="http://www.w3.org/1999/02/22-rdf-syntax-ns#">
  <rdf:Description rdf:about=""
  xmlns:xmp="http://ns.adobe.com/xap/1.0/"
  xmlns:umf="http://ns.intel.com/umf/2.0">
  <umf:next-id>0</umf:next-id>
  <umf:schemas>
  <rdf:Bag>
  <rdf:li rdf:parseType="Resource">
  <!-- JLINK Metadata -->
  <umf:schema>Scene</umf:schema>
  <umf:descriptors>
  <rdf:Bag>
  <rdf:li rdf:parseType="Resource">
  <umf:name>Version</umf:name>
  <umf:type>integer</umf:type>
  </rdf:li>
  <rdf:li rdf:parseType="Resource">
  <umf:name>Title</umf:name>
  <umf:type>string</umf:type>
  </rdf:li>
  <rdf:li rdf:parseType="Resource">
  <umf:name>Note</umf:name>
  <umf:type>string</umf:type>
  </rdf:li>
  <rdf:li rdf:parseType="Resource">
  <umf:schema>Viewport</umf:schema>
  <umf:descriptors>
  <rdf:Bag>
  <rdf:li rdf:parseType="Resource">
  <umf:name>X</umf:name>
  <umf:type>real</umf:type>
  </rdf:li>
  <rdf:li rdf:parseType="Resource">
  <umf:name>Y</umf:name>
  <umf:type>real</umf:type>
  </rdf:li>
  <rdf:li rdf:parseType="Resource">
  <umf:name>XFOV</umf:name>
  <umf:type>real</umf:type>
  </rdf:li>
  <rdf:li rdf:parseType="Resource">
```

```
<umf:name>YFOV</umf:name>
<umf:type>real</umf:type>
</rdf:li>
<rdf:li rdf:parseType="Resource">
<umf:name>ID</umf:name>
<umf:type>integer</umf:type>
</rdf:li>
</rdf:Bag>
</umf:descriptors>
</rdf:li>
<rdf:li rdf:parseType="Resource">
<umf:schema>Image</umf:schema>
<umf:descriptors>
<rdf:Bag>
<rdf:li rdf:parseType="Resource">
<umf:name>Format</umf:name>
<umf:type>string</umf:type>
</rdf:li>
<rdf:li rdf:parseType="Resource">
<umf:name>Href</umf:name>
<umf:type>string</umf:type>
</rdf:li>
</rdf:Bag>
</umf:descriptors>
</rdf:li>
</rdf:Bag>
</umf:descriptors>
</rdf:li>
<rdf:li rdf:parseType="Resource">
<umf:schema>Link</umf:schema>
<umf:descriptors>
<rdf:Bag>
<rdf:li rdf:parseType="Resource">
<umf:schema>Region</umf:schema>
<umf:descriptors>
<rdf:Bag>
<rdf:li rdf:parseType="Resource">
<umf:name>Shape</umf:name>
<umf:type>string</umf:type>
</rdf:li>
<rdf:li rdf:parseType="Resource">
<umf:name>X</umf:name>
<umf:type>real</umf:type>
</rdf:li>
<rdf:li rdf:parseType="Resource">
<umf:name>Y</umf:name>
<umf:type>real</umf:type>
</rdf:li>
<rdf:li rdf:parseType="Resource">
<umf:name>W</umf:name>
<umf:type>real</umf:type>
</rdf:li>
<rdf:li rdf:parseType="Resource">
<umf:name>H</umf:name>
<umf:type>real</umf:type>
</rdf:li>
<rdf:li rdf:parseType="Resource">
<umf:name>Rotation</umf:name>
<umf:type>real</umf:type>
</rdf:li>
</rdf:Bag>
</umf:descriptors>
</rdf:li>
<rdf:li rdf:parseType="Resource">
<umf:name>Duration</umf:name>
<umf:type>integer</umf:type>
</rdf:li>
<rdf:li rdf:parseType="Resource">
<umf:name>VPID</umf:name>
<umf:type>integer</umf:type>
</rdf:li>
```

```
  <rdf:li rdf:parseType="Resource">
  <umf:name>Sprite</umf:name>
  <umf:type>string</umf:type>
  </rdf:li>
  <rdf:li rdf:parseType="Resource">
  <umf:name>To</umf:name>
  <umf:type>string</umf:type>
  </rdf:li>
  </rdf:Bag>
  </umf:descriptors>
  </rdf:li>
  </rdf:Bag>
  </umf:descriptors>
  </rdf:li>
  </rdf:Bag>
  </umf:schemas>
  </rdf:Description>
  </rdf:RDF>
  </x:xmpmeta>
<?xpacket end="w"?>
```

B.7 Example XMP expression with populated metadata elements

```
<x:xmpmeta xmlns:x="adobe:ns:meta/" x:xmptk="XMP Core 5.5.0">
<rdf:RDF xmlns:rdf="http://www.w3.org/1999/02/22-rdf-syntax-ns#">
<rdf:Description rdf:about="" xmlns:xmp="http://ns.adobe.com/xap/1.0/"
xmlns:umf="http://ns.intel.com/umf/2.0">
<umf:next-id>0</umf:next-id>
<umf:schemas>
<rdf:Bag>
<rdf:li rdf:parseType="Resource">
<!-- JLINK Metadata -->
<umf:schema>Scene</umf:schema>
<umf:descriptors>
<rdf:Bag>
<rdf:li rdf:parseType="Resource">
<umf:name>Version</umf:name>
<umf:type>integer</umf:type>
</rdf:li>
<rdf:li rdf:parseType="Resource">
<umf:name>Title</umf:name>
<umf:type>string</umf:type>
</rdf:li>
<rdf:li rdf:parseType="Resource">
<umf:name>Note</umf:name>
<umf:type>string</umf:type>
</rdf:li>
<rdf:li rdf:parseType="Resource">
<umf:schema>Viewport</umf:schema>
<umf:descriptors>
<rdf:Bag>
<rdf:li rdf:parseType="Resource">
<umf:name>X</umf:name>
<umf:type>real</umf:type>
</rdf:li>
<rdf:li rdf:parseType="Resource">
<umf:name>Y</umf:name>
<umf:type>real</umf:type>
</rdf:li>
<rdf:li rdf:parseType="Resource">
<umf:name>XFOV</umf:name>
<umf:type>real</umf:type>
</rdf:li>
<rdf:li rdf:parseType="Resource">
<umf:name>YFOV</umf:name>
<umf:type>real</umf:type>
</rdf:li>
<rdf:li rdf:parseType="Resource">
<umf:name>ID</umf:name>
<umf:type>integer</umf:type>
</rdf:li>
```

```
</rdf:Bag>
</umf:descriptors>
</rdf:li>
<rdf:li rdf:parseType="Resource">
<umf:schema>Image</umf:schema>
<umf:descriptors>
<rdf:Bag>
<rdf:li rdf:parseType="Resource">
<umf:name>Format</umf:name>
<umf:type>string</umf:type>
</rdf:li>
<rdf:li rdf:parseType="Resource">
<umf:name>href</umf:name>
<umf:type>string</umf:type>
</rdf:li>
</rdf:Bag>
</umf:descriptors>
</rdf:li>
</rdf:Bag>
</umf:descriptors>
</rdf:li>
<rdf:li rdf:parseType="Resource">
<umf:schema>Link</umf:schema>
<umf:descriptors>
<rdf:Bag>
<rdf:li rdf:parseType="Resource">
<umf:schema>Region</umf:schema>
<umf:descriptors>
<rdf:Bag>
<rdf:li rdf:parseType="Resource">
<umf:name>Shape</umf:name>
<umf:type>string</umf:type>
</rdf:li>
<rdf:li rdf:parseType="Resource">
<umf:name>X</umf:name>
<umf:type>real</umf:type>
</rdf:li>
<rdf:li rdf:parseType="Resource">
<umf:name>Y</umf:name>
<umf:type>real</umf:type>
</rdf:li>
<rdf:li rdf:parseType="Resource">
<umf:name>W</umf:name>
<umf:type>real</umf:type>
</rdf:li>
<rdf:li rdf:parseType="Resource">
<umf:name>H</umf:name>
<umf:type>real</umf:type>
</rdf:li>
<rdf:li rdf:parseType="Resource">
<umf:name>Rotation</umf:name>
<umf:type>real</umf:type>
</rdf:li>
</rdf:Bag>
</umf:descriptors>
</rdf:li>
<rdf:li rdf:parseType="Resource">
<umf:name>Duration</umf:name>
<umf:type>integer</umf:type>
</rdf:li>
<rdf:li rdf:parseType="Resource">
<umf:name>VPID</umf:name>
<umf:type>integer</umf:type>
</rdf:li>
<rdf:li rdf:parseType="Resource">
<umf:name>Sprite</umf:name>
<umf:type>string</umf:type>
</rdf:li>
<rdf:li rdf:parseType="Resource">
<umf:name>To</umf:name>
<umf:type>string</umf:type>
```

```
</rdf:li>
</rdf:Bag>
</umf:descriptors>
</rdf:li>
</rdf:Bag>
</umf:descriptors>
</rdf:li>
</rdf:Bag>
</umf:schemas>

<umf:metadata>
<rdf:Bag>
<rdf:li rdf:parseType="Resource">
<umf:schema>scene</umf:schema>
<umf:set>
<rdf:Bag>
<rdf:li rdf:parseType="Resource">
<rdf:value>1</rdf:value>
<umf:name>Version</umf:name>
</rdf:li>
<rdf:li rdf:parseType="Resource">
<rdf:value>""</rdf:value>
<umf:name>Title</umf:name>
</rdf:li>
<rdf:li rdf:parseType="Resource">
<rdf:value>""</rdf:value>
<umf:name>Note</umf:name>
</rdf:li>
<rdf:li rdf:parseType="Resource">
<umf:schema>Viewport</umf:schema>
<umf:set>
<rdf:Bag>
<rdf:li rdf:parseType="Resource">
<rdf:value>50</rdf:value>
<umf:name>X</umf:name>
</rdf:li>
<rdf:li rdf:parseType="Resource">
<rdf:value>50</rdf:value>
<umf:name>Y</umf:name>
</rdf:li>
<rdf:li rdf:parseType="Resource">
<rdf:value>100</rdf:value>
<umf:name>XFOV</umf:name>
</rdf:li>
<rdf:li rdf:parseType="Resource">
<rdf:value>100</rdf:value>
<umf:name>YFOV</umf:name>
</rdf:li>
<rdf:li rdf:parseType="Resource">
<rdf:value>1</rdf:value>
<umf:name>ID</umf:name>
</rdf:li>
</rdf:Bag>
</umf:set>
</rdf:li>
<rdf:li rdf:parseType="Resource">
<umf:schema>Image</umf:schema>
<umf:set>
<rdf:Bag>
<rdf:li rdf:parseType="Resource">
<rdf:value>"image/jpeg"</rdf:value>
<umf:name>Format</umf:name>
</rdf:li>
<rdf:li rdf:parseType="Resource">
<rdf:value>"self#jumbf=image1"</rdf:value>
<umf:name>Href</umf:name>
</rdf:li>
</rdf:Bag>
</umf:set>
</rdf:li>
</rdf:Bag>
```