International Telecommunication Union

ITU-T

TELECOMMUNICATION
STANDARDIZATION SECTOR
OF ITU

T.813

(06/2012)

SERIES T: TERMINALS FOR TELEMATIC SERVICES

Still-image compression – JPEG 2000

Information technology – JPEG 2000 image coding system – XML representation and reference

Recommendation ITU-T T.813

ITU-T T-SERIES RECOMMENDATIONS

TERMINALS FOR TELEMATIC SERVICES

Facsimile – Framework	T.0–T.19
Still-image compression – Test charts	T.20–T.29
Facsimile – Group 3 protocols	T.30–T.39
Colour representation	T.40–T.49
Character coding	T.50–T.59
Facsimile – Group 4 protocols	T.60–T.69
Telematic services – Framework	T.70–T.79
Still-image compression – JPEG-1, Bi-level and JBIG	T.80–T.89
Telematic services – ISDN Terminals and protocols	T.90–T.99
Videotext – Framework	T.100–T.109
Data protocols for multimedia conferencing	T.120–T.149
Telewriting	T.150–T.159
Multimedia and hypermedia framework	T.170–T.189
Cooperative document handling	T.190–T.199
Telematic services – Interworking	T.300–T.399
Open document architecture	T.400–T.429
Document transfer and manipulation	T.430–T.449
Document application profile	T.500–T.509
Communication application profile	T.510–T.559
Telematic services – Equipment characteristics	T.560–T.649
Still-image compression – JPEG 2000	**T.800–T.829**
Still-image compression | JPEG XR	T.830–T.849
Still-image compression – JPEG-1 extensions	T.850–T.899

For further details, please refer to the list of ITU-T Recommendations.

INTERNATIONAL STANDARD ISO/IEC 15444-14
RECOMMENDATION ITU-T T.813

Information technology – JPEG 2000 image coding system – XML representation and reference

Summary

Recommendation ITU-T T.813 | ISO/IEC 15444-14 specifies an XML document, referred to as JPXML, which is designed primarily for representing JPEG 2000 file format and codestream marker segments, and a reference method for embedding internal binary data in a JPEG 2000 image.

This Recommendation | International Standard:

- specifies an XML schema for general box file formats;
- specifies an XML schema for JPEG 2000 family file formats and codestream segments;
- specifies a complete referring location path to address to exact box or codestream data in an image;
- provides guidance on processes for converting source image data to an XML structural document;
- provides guidance on how to implement these processes in practice.

History

Edition	Recommendation	Approval	Study Group
1.0	ITU-T T.813	2012-06-29	16

FOREWORD

The International Telecommunication Union (ITU) is the United Nations specialized agency in the field of telecommunications, information and communication technologies (ICTs). The ITU Telecommunication Standardization Sector (ITU-T) is a permanent organ of ITU. ITU-T is responsible for studying technical, operating and tariff questions and issuing Recommendations on them with a view to standardizing telecommunications on a worldwide basis.

The World Telecommunication Standardization Assembly (WTSA), which meets every four years, establishes the topics for study by the ITU-T study groups which, in turn, produce Recommendations on these topics.

The approval of ITU-T Recommendations is covered by the procedure laid down in WTSA Resolution 1.

In some areas of information technology which fall within ITU-T's purview, the necessary standards are prepared on a collaborative basis with ISO and IEC.

NOTE

In this Recommendation, the expression "Administration" is used for conciseness to indicate both a telecommunication administration and a recognized operating agency.

Compliance with this Recommendation is voluntary. However, the Recommendation may contain certain mandatory provisions (to ensure, e.g., interoperability or applicability) and compliance with the Recommendation is achieved when all of these mandatory provisions are met. The words "shall" or some other obligatory language such as "must" and the negative equivalents are used to express requirements. The use of such words does not suggest that compliance with the Recommendation is required of any party.

INTELLECTUAL PROPERTY RIGHTS

ITU draws attention to the possibility that the practice or implementation of this Recommendation may involve the use of a claimed Intellectual Property Right. ITU takes no position concerning the evidence, validity or applicability of claimed Intellectual Property Rights, whether asserted by ITU members or others outside of the Recommendation development process.

As of the date of approval of this Recommendation, ITU had not received notice of intellectual property, protected by patents, which may be required to implement this Recommendation. However, implementers are cautioned that this may not represent the latest information and are therefore strongly urged to consult the TSB patent database at http://www.itu.int/ITU-T/ipr/.

CONTENTS

INTERNATIONAL STANDARD

RECOMMENDATION ITU-T

Information technology – JPEG 2000 image coding system – XML representation and reference

1 Scope

This Recommendation | International standard specifies an XML document, referred to as JPXML, which is designed primarily for representing JPEG 2000 file format and marker segments in the codestream, and a referring method for embedding internal data in a JPEG 2000 image.

This Recommendation | International Standard

- specifies JPXML conversion rules for general box file formats;
- specifies JPXML conversion rules for JPEG 2000 family file formats and codestream segments;
- specifies a complete referring location path to address to exact box or codestream data in an image;
- provides guidance on processes for converting source image data to an XML structural document;
- provides guidance on how to implement these processes in practice.

2 Normative references

The following Recommendations and International Standards contain provisions which, through reference in this text, constitute provisions of this Recommendation | International Standard. At the time of publication, the editions indicated were valid. All Recommendations and Standards are subject to revision, and parties to agreements based on this Recommendation | International Standard are encouraged to investigate the possibility of applying the most recent edition of the Recommendations and Standards listed below. Members of IEC and ISO maintain registers of currently valid International Standards. The Telecommunication Standardization Bureau of the ITU maintains a list of currently valid ITU-T Recommendations.

2.1 Identical Recommendation | International Standards

- Recommendation ITU-T T.800 (2002) | ISO/IEC 15444-1:2004, *Information technology – JPEG 2000 image coding system: Core coding system.*
- Recommendation ITU-T T.801 (2002) | ISO/IEC 15444-2:2004, *Information technology – JPEG 2000 image coding system: Extensions.*
- Recommendation ITU-T T.802 (2005) | ISO/IEC 15444-3:2007, *Information technology – JPEG 2000 image coding system: Motion JPEG 2000.*
- Recommendation ITU-T T.805 (2012) | ISO/IEC 15444-6 (2013), *Information technology – JPEG 2000 image coding system: Compound image file format.*
- Recommendation ITU-T T.807 (2006) | ISO/IEC 15444-8:2007, *Information technology – JPEG 2000 image coding system: Secure JPEG 2000.*
- Recommendation ITU-T T.808 (2005) | ISO/IEC 15444-9:2005, *Information technology – JPEG 2000 image coding system: Interactivity tools, APIs and protocols.*
- Recommendation ITU-T T.809 (2011) | ISO/IEC 15444-10:2011, *Information technology – JPEG 2000 image coding system: Extensions for three-dimensional data.*
- Recommendation ITU-T T.810 (2006) | ISO/IEC 15444-11:2007, *Information technology – JPEG 2000 image coding system: Wireless.*
- Recommendation ITU-T T.812 (2007) | ISO/IEC 15444-13:2008, *Information technology – JPEG 2000 image coding system: Wireless.*

2.2 Paired Recommendations | International Standards equivalent in technical content

- Recommendation ITU-T T.832 (2009), *Information technology – JPEG XR image coding system: An entry level JPEG 2000 encoder.*

  ISO/IEC 29199-2:2009, *Information technology – JPEG XR image coding system – Part 2: Image coding specification.*

- Recommendation ITU-T T.833 (2010), *Information technology – JPEG XR image coding system – Motion JPEG XR*

  ISO/IEC 29199-3:2010, *Information technology – JPEG XR image coding system – Part 3: Motion JPEG XR.*
- ISO 12639:1998, Graphic technology – *Prepress digital data exchange – Tag image file format for image technology (TIFF/IT).*

2.3 Additional references

- ISO/IEC 646:1991, *Information technology – ISO 7-bit coded character set for information interchange.*
- ISO/IEC 15444-12:2012, *Information technology – JPEG 2000 image coding system – Part 12: ISO base media file format.*
- IETF RFC 2045 (1996), *Multipurpose Internet Mail Extensions (MIME) Part One.*
- IETF RFC 2279 (1998), *UTF-8, A transformation format of ISO 10646.*
- IETF RFC 4648 (2006), *The Base16, Base32, and Base64 Data Encodings*.
- W3C Recommendation (2009), *Namespaces in XML 1.0 (Third Edition).*
- W3C Recommendation (2008), *Extensible Markup Language (XML), Version 1.0 (Fifth Edition).*
- W3C Recommendation (2004), *XML Schema Part 0: Primer.*
- W3C Recommendation (2004), *XML Schema Part 1: Structures.*
- W3C Recommendation (2004), *XML Schema Part 2: Datatypes.*
- W3C Recommendation (2010), *XML Path Language (XPath) 2.0.*

3 Definitions

For the purposes of this Recommendation | International Standard, the definitions given in Rec. ITU-T T.800 | ISO/IEC 15444-1 and Rec. ITU-T T.801 | ISO/IEC 15444-2 and those listed below apply. Should there be any difference between the definition given in this clause and the one given in one of the other Recommendation | International Standard cited above, the one given in this clause prevails.

3.1 **…**: elision mark. This mark denotes that some words or characters are erased or abbreviated.

3.2 **4CC**: Four-character codes of the box type generally referred to by an ISO 646 character string translation of the integer value. This value is used for a box type that specifies its contents.

3.3 **absolute offset**: Offset to internal image data from the start of an image file. By the JPXML converter, the offset will be made with "length" attributes from the top to the target elements.

3.4 **box**: A sequence of byte blocks that contains its length, 4CC data type, and contents. Some boxes, such as the "jp2c" box, contain an image codestream; other boxes contain image properties such as image width and height. This data block is the atom of the JPEG 2000 and MPEG 4 image file format.

3.5 **box-based format**: A sequence of boxes that contains several image properties and expresses an image file format. This image format starts with a signature box and contains at least one codestream.

3.6 **box element**: A JPXML element for a box, and this element name is translated from the 4CC of the box type by using conversion rules described in clause 7.

3.7 **codestream**: A sequence of bits contained in a sequence of *bytes*, created by an image coder. This data sequence contains marker segments for holding image coding properties that are parsed by a decoder or translator. This may be arranged so that the most significant bit of the first *byte* is the first bit of the *codestream*, the next most significant bit of the first *byte* is the second bit of the *codestream*, and so on, to the least significant *bytes*.

3.8 **fat representation**: A JPXML document that contains whole image data on text nodes. This representation can be translated to a genuine image without any additional image information because it contains whole image information. However, because of translating whole chunk data into the XML format, this representation needs more data space than was required for the original image. For more details, see 6.1.

3.9 **fat-skeleton representation**: A JPXML document that contains image properties excluding codestream chunk data. This representation may have a location path to chunk image data by using the JPXML format. This represents a box file format image structure. For more details, see 6.1.

3.10 JPXML converter: A converter that translates data between an image and a JPXML document. The JPXML "forward" converter translates an image to a JPXML document, and the JPXML "inverse" converter translates the edited document and codestream data to an image. These converters use rules of element name creation, defined container and undefined chunk container conversions.

3.11 JPXML document: An XML document that corresponds to the box file format or codestream, categorized according to included contents; skeleton, fat-skeleton and fat representations. For more details, see 6.1.

3.12 JPXML document structure: The structure of a JPXML element in a JPXML document, which expresses an image data structure. This structure is used for a location path using XPath expression.

3.13 JPXML element: An XML element represents a box file format or codestream structure, and is translated from a box, marker, or its content. The 4CC box type, marker type, and the "content" are used as this element name. For more details, see 6.2.

3.14 location path: The location of an internal image data using XPath expression with a JPXML document. This expression represents the absolute offset value and the JPXML document structure.

3.15 marker element: A JPXML element for the marker segment. This element name is translated from the marker type using the conversion rule described in 7.1 and 7.3.

3.16 marker segment: A binary data block in a codestream which contains the marker type and may contain marker properties for coding information.

3.17 skeleton representation: A JPXML document which does not contains text nodes. This only represents the structure of a box file format image. For more details, see 6.1.

4 Abbreviations and symbols

The abbreviations and symbols defined in Rec. ITU-T T.800 | ISO/IEC 15444-1, Rec. ITU-T T.801 | ISO/IEC 15444-2, Rec. ITU-T T.805 | ISO/IEC 15444-6, Rec. ITU-T T.808 | ISO/IEC 15444-9, Rec. ITU-T T.810 | ISO/IEC 15444-11, and Rec. ITU-T T.812 | ISO/IEC 15444-13 also apply to this Recommendation | International Standard.

4.1 Abbreviations

For the purposes of this Recommendation | International Standard, the following abbreviations apply:

JPXML	Refers to this Recommendation | International Standard
MIME	Multipurpose Internet Mail Extension
TIFF	Tag Image File Format
XML	eXtended Metadata Language
XSLT	XML Stylesheet Language Transformation

5 Conventions

This Recommendation | International Standard consists of normative and informative text.

Normative text is that text which expresses mandatory requirements. The word "shall" is used to express mandatory requirements strictly to be followed in order to conform to this Specification and from which no deviation is permitted. A conforming implementation is one that fulfils all mandatory requirements.

Informative text is text that is potentially helpful to the user, but not indispensable and can be removed, changed or added editorially without affecting interoperability. All text in this Recommendation | International Standard is normative, with the following exceptions: the Introduction, any parts of the text that are explicitly labelled as "informative", and statements appearing with the preamble "NOTE" and behaviour described using the word "should". The word "should" is used to describe behaviour that is encouraged but is not required for conformance to this Specification.

The keywords "may" and "need not" indicate a course of action that is permissible in a conforming implementation.

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

6 General description

The structural representation for the box-based format and the JPEG 2000 family codestreams is defined in this Recommendation | International Standard, which does not intend to make a new image file format. This image description is described with XML, and this temporary XML document is created as an intermediate image description for accessing internal data robustly and converting an image type. The following subclauses describe more details of this document.

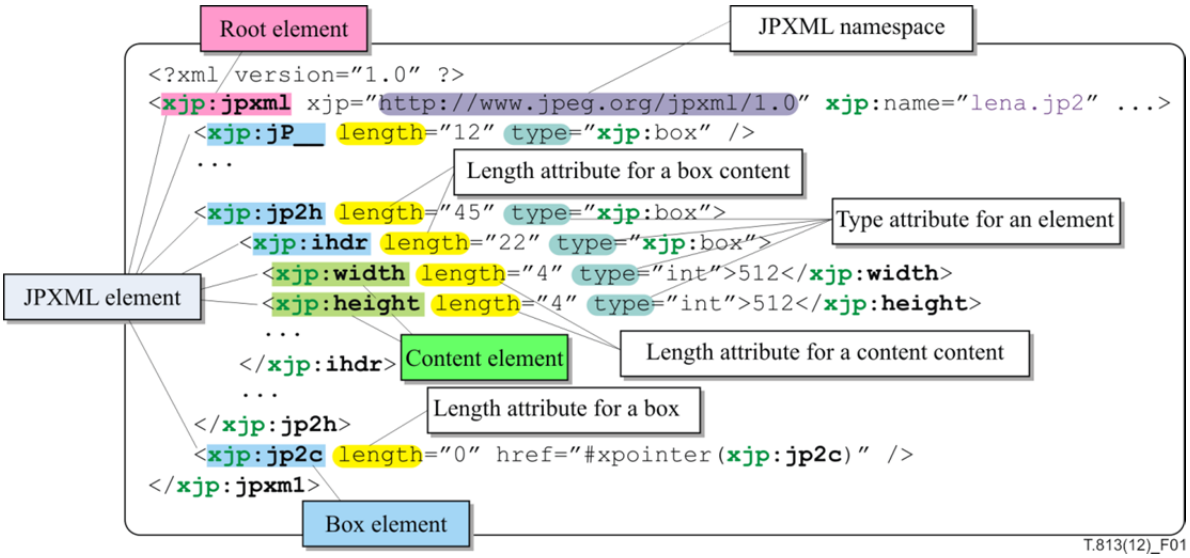


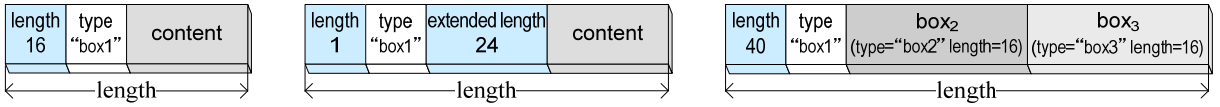
Figure 1 – Example of a JPXML document for a JPEG 2000 image

6.1 Structure of the JPXML document

The JPXML document is described with three elements; a JPXML element, its attribute, and its content value. The JPXML element structure represents an image structure; box, marker segment, and content structure. This document namespace shall be "http://www.iso.org/jpeg/jpxml/1.0", and this document's root element name shall be 'jpxml'. The JPXML element has two types; the first element is a container element which expresses a box or a marker segment itself, and the second one is a content element which expresses a container's property or a box content. Some containers, such as a superbox, contain other containers, and so a JPXML document will have a tree structure. Each JPXML element shall have 'length' and 'type' attributes, and these attributes denote the byte length and data type of each data chunk, respectively. The content value may be described with text, and its data type denoted with 'type' attribute. Figure 1 is an example of a JPXML document for a JPEG 2000 file format.

The container element name, or the box or marker element name, shall be created with the 4CC box type or the segment marker name, respectively, and the converting rules are described in clause 7. The container element may contain some content elements, which are optional in the JPXML document. There may be only one content element even if a container has several data containers, and this content element type attribute should be a "hexbyte" or "unknown" type. This element name shall be "content" or a name predefined in this Recommendation | International Standard. The attributes in the JPXML elements are used for creating an absolute offset from a location path for indicating chunk data in the image. The detail of this process is described in 7.5.

The box described in Rec. ITU-T T.800 | ISO/IEC 15444-1 and the JPXML document for box format are illustrated in Figure 2. Figures 2 (a), (b) and (c) are illustrations of box format structures; normal box, box with extended length and superbox, and documents (d), (e) and (f) represent the (a), (b) and (c) box structure, respectively. The superbox element shall have whole children box and superbox elements. All box elements with an extended length have a length element for storing the actual box length.



(a) box (b) box with extended length (c) superbox

```
<box1 length="16" type="box">
  <content length="8" />
</box1>
```

```
<box1 length="1" type="box">
  <length length="8" type="int">
    24</length>
  <content length="8" />
</box1>
```

```
<box1 length="40" type="box">
  <box2 length="16" type="box" />
  <box3 length="16" type="box" />
</box1>
```

(d) JPXML document for (a) (e) JPXML document for (b) (f) JPXML document for (c)

Figure 2 – Examples of box format and JPXML documents

The JPXML document is generated from an image file format and/or codestreams, and its kind varies from none property to including codestream data representations. When kinds of image property representation are included, the JPXML document is categorized with three levels of representation: "skeleton", "fat-skeleton", and "fat" representations.

The first-level representation, the skeleton representation, shall express only the structure of the image itself, and may contain an attribute for the absolute offset or the location path to the element block. The skeleton shall have no text node in the JPXML elements. This representation is used for a location path that is comparatively robust for changing the box structure of the image and/or marker segment structure of the codestream.

The second-level representation, the fat-skeleton representation, expresses the image structure and some variables of box and/or marker contents. The fat skeleton is an intermediate representation between skeleton and fat representations. Consequently, it also has the skeleton's attribute and the same text node value of JPXML elements, but no binary data (such as a coded codestream). This representation is used for a location path and also some image transformation with XSLT.

The third-and final level representation, the fat representation, expresses the image structure and whole image property values. This whole property may represent a binarized format for use of some applications, such as secure purpose. The binarized contents are translated with MIME's base64 encoding. As this representation requires more data space than the original image data, it is unsuited for use in a storage file format for image data.

6.2 Creation of a JPXML document

A JPXML document is created from a box-based format and/or a JPEG 2000 codestream. This generation process may consist of several steps. Figure 3 is an illustration of a block diagram of an image conversion system using the JPXML, and includes forward and inverse JPXML generators. This example consists of a forward JPXML generator, an image edit tool, and an inverse JPXML generator. The processes of forward and inverse JPXML generators may consist of two modules: a JPXML document generator and a location path generator. The JPXML document generator converts between a binary image and an XML document, and the location path generator converts between an absolute offset number to target data and an XPath location path for a target element.

For creating a JPXML document, the forward JPXML document generator uses several rules: the common conversion rule and the three element name rules. The common conversion rule is defined in 7.1. The document element name rules are for a box format, a marker segment, and a tagged image format. The inverse JPXML document generator uses the inverse rules of the forward converter's rules.

The element name rule for the box-based format creates an element name related to the four-character code (4CC) identifying the box container type. Not all 4CC values are allowable for an XML element name, such as an 'xml' and space character, and these 4CC values are modified for an XML element by using the conversion rules defined in 7.2. The details of the box-based format and four-character code are defined in Rec. ITU-T T.800 | ISO/IEC 15444-1, Rec. ITU-T T.801 | ISO/IEC 15444-2, Rec. ITU-T T.802 | ISO/IEC 15444-3, Rec. ITU-T T.805 | ISO/IEC 15444-6, Rec. ITU-T T.807 | ISO/IEC 15444-8, Rec. ITU-T T.808 | ISO/IEC 15444-9, Rec. ITU-T T.833 | ISO/IEC 29199-3 and ISO/IEC 15444-12. The element names are defined in Annex A.

The element name rule for the marker segment creates an element name related to a two-byte code or a marker segment name identifying the marker segment which is defined in Rec. ITU-T T.800 | ISO/IEC 15444-1, Rec. ITU-T T.801 | ISO/IEC 15444-2, Rec. ITU-T T.807 | ISO/IEC 15444-8, Rec. ITU-T T.809 | ISO/IEC 15444-10, and Rec. ITU-T T.810 | ISO/IEC 15444-11. The two-byte code, non-character and invisible value, is converted to a visible code value for use as an XML element name by using conversion rules defined in 7.3. The details of these element names are defined in Annex B.

The element name rule for the tagged image format creates an element name related to the two byte code of the tag value identifying the tagged maker property which is defined in ISO 12639 and Rec. ITU-T T.832 | ISO/IEC 29199-2. The two byte code, non-character and invisible value, used in the TIFF image and JPEG XR, is converted to a visible code value for use as an XML element name by using conversion rules defined in 7.4.

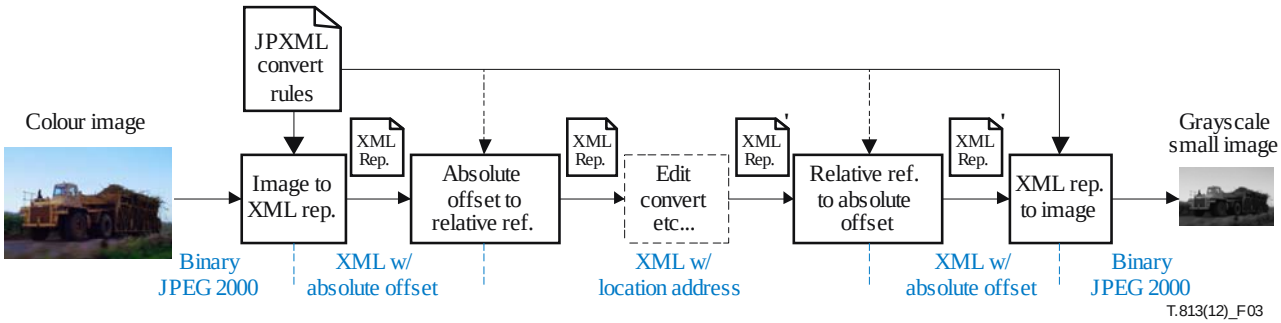


Figure 3 – Block diagram of image converting with JPXML

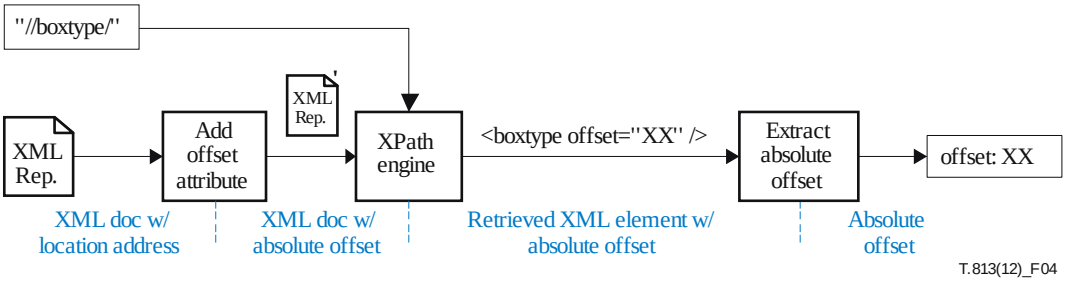


Figure 4 – Example of the process for converting between the XML location path and offset

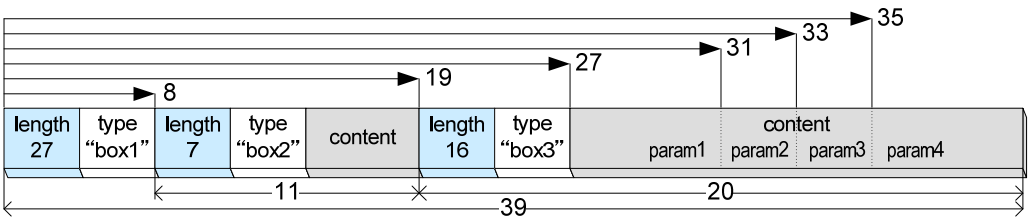
6.3 Access with the JPXML document

The two location representations, an XML location path and an absolute offset, are used to access internal image data with the JPXML document. The XML location path homologizes the target image data to identify the target element in the JPXML document. The absolute offset corresponding to the target element identified by the location path is used for locating the data chunk of the image, and this value shall be used for a binary data access. For converting between the text location and the offset value, conversion rules defined in 8.1 are used.

An absolute offset can be generated from the target location path in several ways. One example of generating a target offset process consists of three steps: 1) set each element's data chunk offset to its element offset attribute, 2) identify a target element from a target location path by using XML tools, and 3) extract the target element offset attribute value as an absolute offset. Figure 4 depicts an example of this process. After this process, the generated absolute offset is the location of the target data chunk from the very start of an image.

A target location path can be generated from a target absolute offset by following three steps: 1) set each element's data chunk offset to its element offset attribute, 2) identify a target element by comparing the element's offset attribute and the target offset, and 3) create a location path from the selected target element. This process creates the location path which identifies the target absolute offset, and this text representation will not suffer from binary data changes in the image.

These processes have the same offset generator for the JPXML document, and these generators use conversion rules defined in 8.1. Figure 5 is an example of a JPXML document with offset calculated by this generator. As shown in this example, the "box2" box offset is eight bytes because each box has eight bytes of data space for the box length and type of storage, and the "box2" is the first child of "box1" box. By using this generated JPXML document, all image data chunks described by the JPXML elements can be accessed by the XML location path.



(a) An example of part of a pseudo box-based format structure

```
 1: <box1 length="39" type="box" offset="0">
 2:   <box2 length="11" type="box" offset="8">
 3:     <content length="3" type="int" offset="16" />
 4:   </box2>
 5:   <box3 length="20" type="box" offset="19">
 6:     <param1   length="4"   type="int"   offset="27"   />
 7:     <param2   length="2"   type="short"   offset="31"   />
 8:     <param3   length="2"   type="short"   offset="33"   />
 9:     <param4 length="4" type="int" offset="35" />
10:   </box3>
11: </box1>
```

(b) Part of a pseudo JPXML document with an offset attribute for the (a) format

```
 1:   <box3 length="16" type="box" offset="19">
 2:     <content length="12" type="hexBinary" offset="27" />
 3:   </box3>
```

(c) Another representation of the box3 element in the (b) document (single-content representation)

Figure 5 – Example of JPXML document accessing

7 Document creation rules

The conversion rules between the JPXML document and the image consist of three parts: the first part is a common conversion rule for file formats and codestreams, the second part is a conversion rule between 4CC and XML element names, and the third part is a conversion rule between marker segment names and XML element names.

7.1 Common rule

All JPXML forward converters shall use the following common forward rules for JPXML conversion. The boxes within a superbox shall be represented in JPXML form when these rules are used for a file format forward conversion. An '`offset`' attribute value shall be converted to a location path by using a JPXML document structure.

1) The namespace of the JPXML document shall be "`http://www.iso.org/jpeg/jpxml/1.0`".
2) The '`jpxml`' shall be used for the JPXML root element, representing one file format or codestream.
3) The root element may have a '`name`' attribute for identifying an image name.
4) All elements shall have '`length`' and '`type`' attributes, representing data byte length and type respectively.
5) The '`type`' attributes value for the box or marker element shall be '`box`' or '`marker`' respectively.
6) A box element with `length=1` attribute shall have a length element, representing the data length in byte.
7) All element may have an '`offset`' attribute, representing the absolute offset to the data chunk in byte.
8) The box element shall have several box elements when the original box has a box inside its content.
9) The element of a box or marker segment may have content elements for representing its properties.
10) The predefined element names defined in later annexes shall be used for element names.
11) The box or marker parameters may be represented in XML form, defined in later annexes, and stored in the JPXML box or marker element.
12) The box or marker parameters may be represented as base64 binary, and stored in the box or marker element.

All JPXML inverse converters shall use the following common inverse rules for JPXML conversion. A box element having one or more box elements shall be converted to a superbox when reconstructing a file format. If an '`offset`' attribute value is the location path using a JPXML document structure, this location path shall be converted to an absolute offset value, and stored in its element attribute.

1) The JPXML document shall have namespace of "`http://www.iso.org/jpeg/jpxml/1.0`".
2) The '`jpxml`' root element shall be converted to one file format or codestream.
3) The '`name`' attribute value in the root element shall be used for a file name.
4) The '`length`' and '`type`' attribute values shall be used for converted data length and type respectively.
5) The element having '`box`' or '`marker`' type shall be converted to a box or marker segment respectively.

6) The node value of the length element in the box element with `length=1` attribute shall be used for its converted data chunk length.

7) The '`offset`' attribute value may be used for its converted data chunk location in the image.

8) All box elements in the box shall be converted into the box content.

9) All child elements in a box or marker element shall be converted into internal contents of the data chunk.

10) All content element shall be combined to a binary data of the parent element.

11) The JPXML content elements defined in later annexes shall be converted to binary data and stored in one of the parent contents of the box or maker.

12) The JPXML content in base64 binary shall be converted to a binary data with the base64 converter and stored in one of the parent contents of the box or maker.

Table 1 – Example of 4CC box type conversions

Box name	Box type	Hex decimal	JPXML element name
JPEG 2000 Signature box	`jP\040\040`	`0x6A50 2020`	`jp__`
JP2 Header box	`jp2h`	`0x6A70 3268`	`jp2h`
Resolution box	`url\040`	`0x7572 6C20`	`url_`
URL box	`res\040`	`0x7265 7320`	`res_`
XML box	`xml\040`	`0x786D 6C20`	`_xml_`

7.2 Element name rule for box format

The JPXML forward converter for file formats translates a box-based format to a JPXML document, and shall use the following forward conversion rule for the 4CC:

1) The JPXML element name shall use a 4CC box type.

2) The alphanumeric characters in 4CC box type shall be directly used for the element name.

3) The space, '`\040`' code, shall be represented with a '_' character for the JPXML element name.

4) The code '`.HH`' (`H`: hexadecimal character = 0, …, 9, A, …, F) shall be used for any other characters.

5) The '_' character at the first character of the element name shall be the escape character.

The JPXML inverse converter to file formats translates a JPXML document to a box-based format, and shall use the following inverse conversion rule for the 4CC.

1) The JPXML element name shall be converted to a 4CC box type, and creates its type box.

2) The '_' character at the first character of the element name may not be removed from the 4CC box type.

3) The alphanumeric name shall be directly used for the 4CC box type.

4) The '_' character in the element name space shall be represented with a '`\040`' code for the 4CC.

5) The '`.HH`' string in the name shall be converted to a '`0xHH`' code character.

7.3 Element name rule for a tagged image format

The JPXML forward converter for tagged image file formats, translates a tag value in a directory entry to a JPXML document element name and shall use the following forward conversion rule.

1) The two bytes tag value shall be represented as the four characters hex string notation, '`HHHH`' (`H`: hexadecimal character = 0, …, 9, A, …, F).

2) The '_' character shall be placed at the front of the four character string, and creates a five characters string ('`_HHHH`').

The JPXML inverse converter to tagged image file formats translates a JPXML document name to a tag value in a directory entry, and shall use the following inverse conversion rule for the 4CC.

1) The first '_' character shall be removed from the five-character string, and shall create a four-character string.

2) The four character-string, '`HHHH`' shall be converted to a '`0xHHHH`' code value

7.4 Element name rule for a marker segment

The JPXML forward converter for marker segments translates JPEG 2000 marker segments to a JPXML document, and shall use the following forward conversion rule.

1) The marker symbol name shall be used for a JPXML codestream element name, or the forward conversion rule for a tagged image file format shall be used for converting the marker symbol code to an element name.

The JPXML inverse converter to marker segments translates a JPXML document to JPEG 2000 marker segments, and shall use the following inverse conversion rule.

1) All elements named with its marker symbol name shall be converted to its named marker symbol code.
2) All elements with a name starting with a '_' character of the five-character string shall be converted to the marker symbol code by using the inverse conversion rule for the tagged image file format.

Table 2 – Example of codestream marker conversions

Codestream marker name	Symbol name	Code	JPXML element name
Start of codestream	SOC	0xFF4F	SOC
Start of tile-part	SOT	0xFF90	SOT
Start of data	SOD	0xFF93	SOD
End of codestream	EOC	0xFFD9	EOC
Start of packet	SOP	0xFF91	SOP

7.5 Element type attributes

All elements have a type attribute which identifies the data type of the element content data. This Recommendation | International Standard defines uses of six data types: box, marker, fourcc, location, hexbyte, and integer and time. The integer and time data types are used for the element content data type as integer and time types, respectively, and these data types are defined in the XML schema Part 2 Recommendation. The box and marker data types are used to identify the box element and the marker element, respectively. The fourcc data type indicates the element content data that shall be the 4CC data type, and the 4CC value will not converted with the rules previously described. The location indicates that the elements can have any location type value. The hexbyte indicates that the element data may be represented as a hexadecimal value in big-endian order. The XML definitions for the box, marker, fourcc, location and hexbyte data types can be defined by the following XML schema's simple type data type definitions:

```
<xs:simpleType name="box">
  <xs:restriction base="xs:string" />
</xs:simpleType>

<xs:simpleType name="marker">
  <xs:restriction base="xs:string" />
</xs:simpleType>

<xs:simpleType name="fourcc">
  <xs:restriction base="xs:string">
    <xs:pattern value="[ a-zA-Z0-9_]{4}"/>
  </xs:restriction>
</xs:simpleType>

<xs:simpleType name="location">
  <xs:restriction base="xs:anyURL" />
</xs:simpleType>

<xs:simpleType name="hexbyte">
  <xs:restriction base="xs:string">
    <xs:pattern value="([a-f0-9][a-f0-9])+"/>
  </xs:restriction>
</xs:simpleType>
```

8 Accessing image data

The XML location path and the JPXML document can be used to access internal binary chunk data in the image data. The following three steps access the internal binary data: the first step identifies the target element by using the location path and the JPXML document, the second step converts the target element to an absolute offset, and the third and final step accesses the internal image data with the target absolute offset position. To achieve the second step, the following rules are employed.

8.1 Rules for location conversion

The following conversion rules are for calculating an absolute offset of the target chunk from the location path point to the target element, which corresponds to the offset. To comply with these rules, there may be two approaches for calculation of the target absolute offset value; the target to root approach and the root to target approach.

1) The absolute offset shall be the sum of length attribute values between the top and the target elements, except for the following elements.
2) For the superbox child elements, the value 8 shall be used for offset counting instead of the superbox length attribute value.
3) The length of the child boxes contained in the parent box, ending at the beginning of the target box, will not be used for offset counting, and the length of the parent box is used for counting.

Figures 6 and 7 are illustrations of two examples of an algorithm for calculating the absolute offset value from the target element using these rules. In these algorithms, the root element shall not be used as target element, because the root element is not part of the image data. The first algorithm calculates all element nodes from the target to the root elements, and the second algorithm calculates all element nodes from the first child of the root to the target elements.

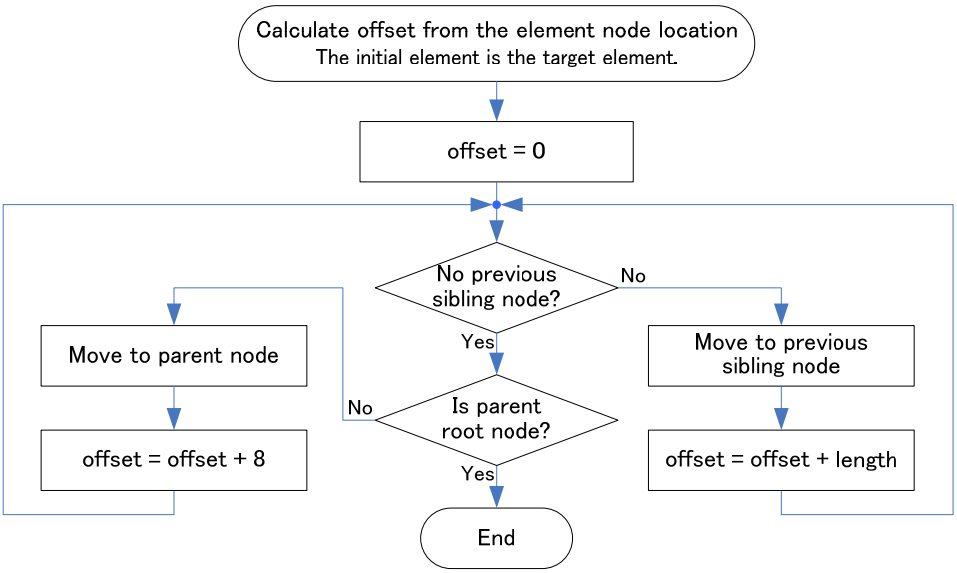


Figure 6 – Example 1 of an algorithm for calculating an offset value from the target element

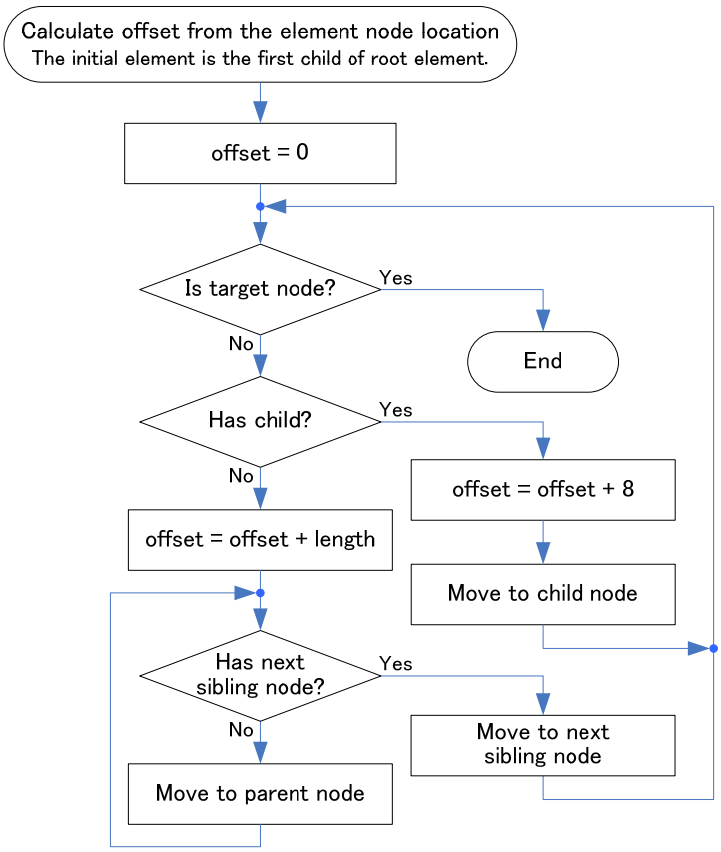


Figure 7 – Example 2 of an algorithm for calculating an offset value from the target element

8.2 Example of location conversion

Figure 8 shows an example of offset calculations for a pseudo JPXML document. The third and fourth columns, labelled "offset 1" and "offset 2", are the result of the target to root approach and the root to target approach, respectively. Their values are calculated by the algorithms depicted in Figures 6 and 7, respectively. In this figure, the offset 1, a+b+d+8, for the "BoxH" box is not the same as the offset 2, a+b+f+g+24. However, the "BoxD" box length, d, is the same as the summation of its content and header lengths, f+g+16, and so these offsets have identical value.

Line	Pseudo JPXM document	offset 1	offset 2
1	<jpxml … >		
2	<BoxA length="a" type="box" />	0	0
3	<BoxB length="b" type="box" />	a	a
4	<BoxC length="c" type="box">	a+b	a+b
5	<BoxD length="d" type="box">	a+b+8	a+b+8
6	<BoxE length="e" type="box">	a+b+16	a+b+16
7	<BoxF length="f" type="box" />	a+b+24	a+b+24
8	<BoxG length="g" type="box" />	a+b+f+24	a+b+f+24
9	</BoxE>		
10	</BoxD>		
11	<BoxH length="h" type="box" />	a+b+d+8	a+b+f+g+24
12	</BoxC>		
13	<BoxI length="i" type="box" />	a+b+c	a+b+f+g+h+24
14	<BoxJ length="j" type="box">	a+b+c+i	a+b+f+g+h+i+24
15	<BoxK length="k" type="box" />	a+b+c+i+8	a+b+f+g+h+i+32
16	<BoxL length="l" type="box">	a+b+c+i+k+8	a+b+f+g+h+i+k+32
17	<length length="8" type="integer" />	a+b+c+i+k+16	a+b+f+g+h+i+k+40
18	<BoxM length="m" type="box" />	a+b+c+i+k+24	a+b+f+g+h+i+k+48
19	</BoxL>		
20	</BoxJ>		
21	</BoxN length="n" type="box" />	a+b+c+j	a+b+f+g+h+i+k+m+48
22	</jpxml>		

Figure 8 – Example of offsets generated for a pseudo JPXML document

Annex A

JPXML elements for box file format

(This annex forms an integral part of this Recommendation | International Standard.)

A.1 Introduction

This annex provides the content element name definitions, and length and type attributes for JPEG 2000 family formats. The box element appearance and structure shall be in conformity with Rec. ITU-T T.800 | ISO/IEC 15444-1, Rec. ITU-T T.801 | ISO/IEC 15444-2, Rec. ITU-T T.802 | ISO/IEC 15444-3, Rec. ITU-T T.805 | ISO/IEC 15444-6, Rec. ITU-T T.807 | ISO/IEC 15444-8, Rec. ITU-T T.808 | ISO/IEC 15444-9 and ISO/IEC 15444-12.

A.2 Box element definitions

This subclause includes many tables of the box content element definitions, and these tables define the content element name, the content data length in bytes, and the content data type. The JPXML data types are described in 7.5. Some tables define the sub-content element names of a defined content element, and some box or content elements have content element definitions of two or more types. For more detailed information of the meaning and structure of box content, refer to Recommendations | International Standards listed in A.1.

A.2.1 Still picture format definitions

The box file formats defined in Rec. ITU-T T.800 | ISO/IEC 15444-1 and Rec. ITU-T T.801 | ISO/IEC 15444-2 are designed to represent a still picture image. Table A.1 illustrates element structures of these file formats. These content element names and types are defined in the following tables.

Table A.1 – Box element structure for a still image

Element name	Box name	Part 1	Part 2	Defined
jP__	*JPEG family signature box*	required	required	1
ftyp	*File type and compatibility box*	required	required	1
rreq	*Reader requirement box*		required	2
jp2h	*JP2 header box (superbox)*	required	optional	1, 2
lbl_	*Label box*		optional	2
ihdr	*Image header box*	required	required	1, 2, 6
bpcc	*Bits per component box*	optional	optional	1, 2
colr	*Colour specification box*	required	required	1, 2, 6
pclr	*Palette box*	optional	optional	1
cmap	*Component mapping box*	optional	optional	1
cdef	*Channel description box*	optional	optional	1
res_	*Resolution box (superbox)*	optional	optional	1
resc	*Capture resolution box*	optional	optional	1
resd	*Display resolution box*	optional	optional	1
jpch	*Codestream header box (superbox)*		optional	2
lbl_	*Label box*		optional	2
ihdr	*Image header box*		required	1, 2, 6
bpcc	*Bits per component box*		optional	2
pclr	*Palette box*		optional	1
cmap	*Component mapping box*		optional	1
jplh	*Compositing layer header box (superbox)*		optional	2
lbl_	*Label box*		optional	2
cgrp	*Colour group box (superbox)*		optional	2
colr	*Colour specification box*		optional	1, 2, 6
opct	*Opacity box (superbox)*		optional	1, 2
cdef	*Channel description box*		optional	1
creg	*Codestream registration box*		optional	2
pxfm	*Pixel format box*		optional	2
res_	*Resolution box (superbox)*		optional	1
dtbl	*Data reference box*		optional	2, 6, 12
ftbl	*Fragment table box (superbox)*		optional	2, 6

Table A.1 – Box element structure for a still image

Element name	Box name	Part 1	Part 2	Defined
flst	*Fragment list box*		optional	2, 6
jp2c	*Contiguous codestream box*	required	optional	1, 2, 6
mdat	*Metadata box*		optional	2
comp	*Composition box*		optional	2
copt	*Composition options box*		optional	2
inst	*Instruction set box*		optional	2
drep	*Desired reproductions box (superbox)*		optional	2
gtso	*Graphics technical standard output box*		optional	2
roid	*ROI description box*		optional	2
cref	*Cross reference box (superbox)*		optional	2, 6
flst	*Fragment list box*		optional	2, 6
asoc	*Association box (superbox)*		optional	2
dxml	*Decomposed XML box*		optional	2
nlst	*Number list box*		optional	2
lbl_	*Label box*		optional	2
bfil	*Binary filter box*		optional	2
chck	*Digital signature box*		optional	2
mp7b	*MPEG-7 binary box*		optional	2
free	*Free space box*		optional	2
xml	*XML container box*	optional	optional	1, 2, 12
jp2i	*IPR box*	optional	optional	1
uuid	*UUID box*	optional	optional	1
uinf	*UUID information box (superbox)*	optional	optional	1
ulst	*UUID list box*	optional	optional	1
url_	*Data entry box*	optional	optional	1
jclx	*Compositing layer extensions box*		optional	2
jlxi	*Compositing layer extensions info box*		optional	2
j2cx	*Multiple codestream box*		optional	2
j2ci	*Multiple codestream info box*		optional	2
grp_	*Grouping box*		optional	2

A.2.1.1 Box element definitions for JPEG 2000 Part 1 format

Table A.2 – Content element name for jP__

content	length	type	loop
signature	12	hexbyte	

Table A.3 – Content element name for ftyp

content	length	type	loop
brand	4	fourcc	
version	4	integer	
compatibility	4	fourcc	↵

Table A.4 – Content element name for jp2h

content	length	type	loop
ihdr	–	box	=1
bpcc	optional	box	≤1
colr	–	box	≥1
pclr	optional	box	≥0
cmap	optional	box	≤1
cdef	–	box	≤1
res_	–	box	≤1

Table A.5 – Content element name for ihdr

content	length	type	loop
height	4	integer	
width	4	integer	
num_components	2	integer	
depth	1	integer	
compression	1	integer	
colour_unknown	1	integer	
ipr	1	integer	

Table A.6 – Content element name for colr

Content	length	type	loop
method	1	integer	
precedence	1	integer	
approx	1	integer	
colour	≥4	hexbyte	

Table A.7 – Content element name for bpcc

content	length	type	loop
depth	1	integer	⌋

Table A.8 – Content element name for pclr

content	length	type	loop
num_entries	2	integer	
num_components	1	integer	
depth	1	integer	⌋
data	num_entries	hexbyte	⌋

Table A.9 – Content element name for cdef

content	length	type	loop
num_entries	2	integer	
index	2	integer	⌉
type	2	integer	⌋
assoc	2	hexbyte	⌋

Table A.10 – Content element name for res_

content	length	type	loop
resc	–	box	≤1
resd	–	box	≤1

Table A.11 – Content element name for resc and resd

content	length	type	loop
vert_num	2	integer	
vert_den	2	integer	
hori_num	2	integer	
hori_den	2	integer	
vert_exp	2	integer	
hori_exp	2	integer	

Table A.12 – Content element name for uuid

content	length	type	loop
id	16	integer	
data	0-65,531	hexbyte	

Table A.13 – Content element name for uinf

content	length	type	loop
ulst	–	box	=1
url_	–	box	=1

Table A.14 – Content element name for ulst

content	length	type	loop
num_entries	2	integer	
data	16	hexbyte	⌋

Table A.15 – Content element name for url_

content	length	type	loop
location	0-65,531	location	

Table A.16 – Content element name for _xml_

content	length	type	loop
text	0-65,531	string	

Table A.17 – Content element name for jp2c

content	length	type	loop
data	≥0	hexbyte	

A.2.1.2 Box element definitions for JPEG 2000 part 2 format

Table A.18 – Content element name for rreq

content	length	type	loop
length	1	integer	
fua_mask	length	hexbyte	
dc_mask	length	hexbyte	
num_std_flags	2	integer	
std_flag	2	integer	⌉
std_mask	length	hexbyte	⌋
num_vender_features	2	integer	
vender_feature	16	integer	⌉
vender_mask	length	hexbyte	⌋

Table A.19 – Content element name for dtbl

content	length	type	loop
num_entries	2	integer	
location	variable	location	

Table A.20 – Content element name for lbl_

content	length	type	loop
label	variable	string	

Table A.21 – Content element name for mp7b

content	length	type	loop
mpeg7	variable	hexbyte	

Table A.22 – Content element name for ftbl

content	length	type	loop
flst	–	box	≥0

Table A.23 – Content element name for free

content	length	type	loop
data	0-65,531	hexbyte	

Table A.24 – Content element name for flst

content	length	type	loop
num_entries	2	integer	
offset	8	integer	⎡
length	4	integer	⎢
data_reference	2	integer	⎣

Table A.25 – Content element name for cref

content	length	type	loop
type	4	integer	=1
flst	–	box	=1

Table A.26 – Content element name for jpch

content	length	type	loop
lbl_	–	box	≥0
ihdr	–	box	
bpcc	–	box	
pclr	–	box	
cmap	–	box	
roid	–	box	

Table A.27 – Content element name for jplh

content	length	type	loop
lbl_	–	box	≥0
cgrp	–	box	≥0
opct	–	box	≥0
cdef	–	box	≥0
creq	–	box	≥0
res_	–	box	≤1

Table A.28 – Content element name for cgrp

content	length	type	loop
colr	0-65,531	box	≥1

Table A.29 – Content element name for colour

colour	length	type	loop
enum_colour	4	integer	
enum_param	variable	hexbyte	

colour	length	type	loop
vender_colour	16	integer	
vender_param	variable	hexbyte	

Table A.30 – Content element name for enum_param

enum_param	length	type	loop
range_L	4	integer	
offset_L	4	integer	
range_a	4	integer	
offset_a	4	integer	
range_b	4	integer	
offset_b	4	integer	
illuminant	4	integer	

enum_param	length	type	loop
range_J	4	integer	
offset_J	4	integer	
range_a	4	integer	
offset_a	4	integer	
range_b	4	integer	
offset_b	4	integer	

Table A.31 – Content element name for opct

content	length	type	loop
type	1	integer	
num_entries	1	integer	
data	1	integer	⎦

Table A.32 – Content element name for drep

content	length	type	loop
gtso	–	box	

Table A.33 – Content element name for gtso

content	length	type	loop
profile	variable	hexbyte	

Table A.34 – Content element name for creg

content	length	type	loop
hori_size	2	integer	
vert_size	2	integer	
cds_num	2	integer	
hori_resolution	1	integer	
vert_resolution	1	integer	
hori_offset	1	integer	
vert_offset	1	integer	

Table A.35 – Content element name for comp

content	length	type	loop
copt	–	box	=1
inst	–	box	

Table A.36 – Content element name for copt

content	length	type	loop
height	4	integer	
width	4	integer	
loop	1	integer	

Table A.37 – Content element name for inst

content	length	type	loop
type	2	integer	
repetition	2	integer	
duration	1	integer	
instruction	variable	hexbyte	

Table A.38 – Content element name for instruction

instruction	length	type	loop
hori_offset	0, 4	integer	
vert_offset	0, 4	integer	
width	0, 4	integer	
height	0, 4	integer	
life	0, 4	integer	
next_use	0, 4	integer	
hori_crop_offset	0, 4	integer	
vert_crop_offset	0, 4	integer	
crop_width	0, 4	integer	
crop_height	0, 4	integer	

Table A.39 – Content element name for roid

content	length	type	loop
num_entries	1	integer	
contained	1	integer	
type	1	integer	
priority	1	integer	
left	4	integer	
top	4	integer	
width	4	integer	
height	4	integer	

Table A.40 – Content element name for asoc

content	length	type	loop
Association box	–	box	=1
Associated box	–	box	⌋

Table A.41 – Content element name for nlst

content	length	type	loop
asoc_num	4	integer	⌋

Table A.42 – Content element name for bfil

content	length	type	loop
type	16	integer	
data	variable	hexbyte	

Table A.43 – Content element name for chck

content	length	type	loop
signature_type	1	integer	
source_type	1	integer	
offset	0, 8	integer	
length	0, 8	integer	
data	length	hexbyte	

A.2.2 Definitions of motion picture format

The box file formats defined in Rec. ITU-T T.802 | ISO/IEC 15444-3 and ISO/IEC 15444-12 are designed to represent a motion picture image. Table A.44 is the box element and element structures for these file formats. These content element names and types are defined in the following tables.

Table A.44 – Box element structure for a motion/video image

Element name	Box name	Part 3	Part 12
jP__	*JPEG family signature box*	required	
ftyp	*File type and compatibility box*	required	required
pdin	*Progressive download information box*		optional
moov	*Movie box (superbox)*	required	required
mvhd	*Movie header box*	required	required
trak	*Track box (superbox)*	required	required
tkhd	*Tack header box*	required	required
tref	*Track reference container box*	optional	optional
edts	*Edit list container box (superbox)*	optional	optional
elst	*Edit list box*	optional	optional
udta	*User-data box (superbox)*	optional	optional
cprt	*Copyright box*	optional	optional
tsel	*Track selection box*	optional	optional
mdia	*Media box (superbox)*	required	required
mdhd	*Media header box*	required	required
hdlr	*Handler box*	required	required
minf	*Media information box (superbox)*	required	required
vmhd	*Video media header box*	optional	optional
smhd	*Sound media header box*	optional	optional
hmhd	*Hint media header box*	optional	optional
dinf	*Data information box (superbox)*	required	required
dref	*Data reference box*	required	required
url_|urn_	*Data entry URL or Data entry URN*	optional	optional

Table A.44 – Box element structure for a motion/video image

Element name	Box name	Part 3	Part 12
url_	*Data entry URL*	optional	optional
urn_	*Data entry URN*	optional	optional
stbl	*Sample table box (superbox)*	required	required
stsd	*Sample descriptions box*	required	required
mjp2	*Video sample entry box*	optional	
jp2h	*JP2 header box (superbox)*	required	
fiel	*Field coding box*	optional	
jp2p	*Motion JPEG 2000 profile box*	optional	
jp2x	*Motion JPEG 2000 prefix box*	optional	
jsub	*Motion JPEG 2000 sub-sampling box*	optional	
orfo	*Motion JPEG 2000 original format box*	optional	optional
raw_	*Audio sample entry box*	optional	
twos	*Audio sample entry box*	optional	
stts	*Time-to-sample box (decoding)*	required	required
ctts	*Time-to-sample box (composition)*		optional
stsc	*Sample-to-chunk box*	required	required
stsz	*Sample sizes box*	required	optional
stz2	*Compact sample size box*		optional
stco | co64	*Chunk offset box (32 bit or 64 bit)*	required	required
stss	*Sync sample table box*		optional
stsh	*Shadow sync sample table box*		optional
padb	*Sample padding bits box*		optional
stdp	*Sample degradation box*		optional
sdtp	*Independent and disposable sample box*		optional
sbgp	*Sample-to-group box*		optional
sgpd	*Sample group description box*		optional
subs	*Sub-sample information box*		optional
meta	*Metadata box*		optional
meta	*Metadata box*		optional
udta	*User-data box (superbox)*	optional	optional
mvex	*Movie extends box (superbox)*	optional	optional
mehd	*Movie extends header box*	optional	optional
trex	*Track extends defaults box*	required	required
ipmc	*IPMP control box*		optional
mdat	*Media data container box*	optional	optional
moof	*Movie fragment box (superbox)*	optional	optional
mfhd	*Movie fragment header box*	required	required
traf	*Track fragment box (superbox)*	optional	optional
tfhd	*Track fragment header box*	required	required
trun	*Track fragment run box*	optional	optional
sdtp	*Independent and disposable sample box*		optional
sbgp	*Sample-to-group box*		optional
subs	*Sub-sample information box*		optional
mfra	*Movie fragment random access box (superbox)*	optional	optional
tfra	*Track fragment random access box*	optional	optional
mfro	*Movie fragment rand. access offset box*	required	required
meta	*Metadata box*		optional
hdlr	*Handler box*		required
dinf	*Data information box (superbox)*		optional
ipmc	*IPMP control box*		optional

Table A.44 – Box element structure for a motion/video image

Element name	Box name	Part 3	Part 12
iloc	*Item location box*		optional
ipro	*Item protection box*		optional
sinf	*Protection scheme information box (superbox)*		optional
frma	*Original format box*		optional
imif	*IPMP information box*		optional
schm	*Scheme type box*		optional
schi	*Scheme information box (superbox)*		optional
iinf	*Item information box*		optional
xml	*XML container box*		optional
bxml	*Binary XML container box*		optional
pitm	*Primary item reference box*		optional
fiin	*File delivery item information box*		optional
paen	*Partition entry box (superbox)*		optional
fpar	*File pattern box*		optional
fecr	*FEC reservoir box*		optional
segr	*File delivery session group box*		optional
gitn	*Group id name box*		optional
tsel	*Track selection box*		optional
dinf	*Data information box (superbox)*		optional
meco	*Additional metadata container box (superbox)*		optional
mere	*Metabox relation box*		optional
meta	*Metadata box*		optional
skip|free	*Free space box*	optional	optional

A.2.2.1 Box element definitions for JPEG 2000 Part 3 format

Table A.45 – Content element name for mjp2

content	length	type	loop
reserved	6	integer	
data_reference	2	integer	
predefined	2	integer	
reserved	2	integer	
predefined	12	integer	
width	2	integer	
height	2	integer	
hori_resolution	4	integer	
vert_resolution	4	integer	
reserved	4	integer	
predefined	2	integer	
name	32	string	
depth	2	integer	
predefined	2	integer	
jp2h	–	box	
fiel	optional	box	
jp2p	optional	box	
jp2x	optional	box	
jsub	optional	box	
orfo	optional	box	

Table A.46 – Content element name for fiel

content	length	type	loop
field_count	1	integer	
field_order	1	integer	

Table A.47 – Content element name for jp2p

content	length	type	loop
version	1	integer	
flag	3	integer	
compatibility	4	integer	

Table A.48 – Content element name for jp2x

content	length	type	loop
codestream	variable	hexbyte	

Table A.49 – Content element name for fiel

content	length	type	loop
field_count	1	integer	
field_order	1	integer	

Table A.50 – Content element name for orfo

content	length	type	loop
original_field_count	1	integer	
original_field_order	1	integer	

Table A.51 – Content element name for jsub

content	length	type	loop
hori_sub	1	integer	
vert_sub	1	integer	
hori_offset	1	integer	
vert_offset	1	integer	

Table A.52 – Content element name for raw_ and twos

content	length	type	loop
reserved	6	integer	
data_reference	2	integer	
reserved	8	integer	
channel_count	2	integer	
sample_size	2	integer	
predefined	2	integer	
reserved	2	integer	
sample_rate	4	integer	

A.2.2.2 Box element definitions for JPEG 2000 part 12 format

Table A.53 – Content element name for mdat

content	length	type	loop
data	variable	hexbyte	

Table A.54 – Content element name for pdin

content	length	type	loop
version	1	integer	
flag	3	integer	
rate	4	integer	⌉
inital_dalay	4	integer	

Table A.55 – Content element name for moov

content	length	type	loop
mvhd	–	box	=1
trak	–	box	≥1
mvex	optional	box	≤1
udta	optional	box	≤1
meta	optional	box	≤1
meco	optional	box	≤1
ipmc	optional	box	≤1

Table A.56 – Content element name for mvhd

content	length	type	loop
version	1	integer	
flag	3	integer	
creation_time	4, 8	integer	
modification_time	4, 8	integer	
time_scale	4	integer	
duration	4, 8	integer	
rate	4	integer	
volume	2	integer	
reserved	10	integer	
matrix	36	hexbyte	
predefined	24	integer	
next_track_id	4	integer	

Table A.57 – Content element name for free and skip

content	length	type	loop
data	variable	hexbyte	

Table A.58 – Content element name for trak

content	length	type	loop
tkhd	–	box	=1
mdia	–	box	=1
tref	optional	box	≤1
edts	optional	box	≤1
udta	optional	box	≤1
meta	optional	box	≤1
meco	optional	box	≤1

Table A.59 – Content element name for tkhd

content	length	type	loop
version	1	integer	
flag	3	integer	
creation_time	4, 8	integer	
modification_time	4, 8	integer	
track_id	4	integer	
reserved	4	integer	
duration	4, 8	time	
reserved	8	integer	
layer	2	integer	
alternate_group	2	integer	
volume	2	integer	
reserved	2	integer	
matrix	36	hexbyte	
width	4	integer	
height	4	integer	

Table A.60 – Content element name for matrix

matrix	length	type	loop
cell	4	integer	

Table A.61 – Content element name for tref

content	length	type	loop
hint|cdsc|hind	–	hexbyte	

Table A.62 – Content element name for hint, cdsc, and hind

hint|cdsc|hind	length	type	loop
track_id	4	integer	⌋

Table A.63 – Content element name for mdia

content	length	type	loop
mdhd	–	box	=1
hdlr	–	box	=1
minf	–	box	=1

Table A.64 – Content element name for mdhd

content	length	type	loop
version	1	integer	
flag	3	integer	
creation_time	4, 8	integer	
modification_time	4, 8	integer	
time_scale	4	integer	
duration	4, 8	integer	
language	2	integer	
predefined	2	integer	

Table A.65 – Content element name for hdlr

content	length	type	loop
version	1	integer	
flag	3	integer	
predefined	4	integer	
type	4	integer 'vide' 'soun' 'hint' 'meta'	
reserved	12	integer	
name	variable	string	

Table A.66 – Content element name for minf

content	length	type	loop
vmhd | smhd | hmhd | nmhd	–	box	=1
stbl	–	box	=1
dinf	–	box	=1

Table A.67 – Content element name for vmhd

content	length	type	loop
version	1	integer	
flag	3	integer	
graphic_mode	2	integer	
op_code	2	integer	

Table A.68 – Content element name for hmhd

content	length	type	loop
version	1	integer	
flag	3	integer	
max_pdu_size	2	integer	
avg_pdu_size	2	integer	
max_bitrate	4	integer	
avg_bitrate	4	integer	
reserved	4	integer	

Table A.69 – Content element name for smhd

content	length	type	loop
version	1	integer	
flag	3	integer	
balance	2	integer	
reserved	2	integer	

Table A.70 – Content element name for stbl

content	length	type	loop
stsd	–	box	=1
stdp	optional	box	≤1
stts	–	box	=1
ctts	optional	box	≤1
stss	optional	box	≤1
stsh	optional	box	≤1
sdtp	optional	box	≤1
stsz | stz2	–	box	=1
stsc	–	box	=1
stco | co64	–	box	=1
padb	optional	box	≤1
subs	optional	box	≤1
sbgp	optional	box	≥0
sgpd	optional	box	≥0

Table A.71 – Content element name for btrt

content	length	type	loop
buffer_size	4	integer	
max_bitrate	4	integer	
avg_bitrate	4	integer	

Table A.72 – Content element name for metx

content	length	type	loop
reserved	6	integer	
data_reference	4	integer	
content_encoding	optional	string	
namespace	variable	string	
location	optional	string	
btrt	optional	box	≥0

Table A.73 – Content element name for mett

content	length	type	loop
reserved	6	integer	
data_reference	4	integer	
content_encoding	optional	string	
mine_format	variable	string	
btrt	optional	box	≥0

Table A.74 – Content element name for pasp

content	length	type	loop
hori_spacing	4	integer	
vert_spacing	4	integer	

Table A.75 – Content element name for clap

content	length	type	loop
width_num	4	integer	
width_den	4	integer	
height_num	4	integer	
height_den	4	integer	
hori_offset_num	4	integer	
hori_offset_den	4	integer	
vert_offset_num	4	integer	
vert_offset_den	4	integer	

Table A.76 – Content element name for Video coding

content	length	type	loop
reserved	6	integer	
data_reference	2	integer	
predefined	2	integer	
reserved	2	integer	
predefined	12	integer	
width	2	integer	
height	2	integer	
hori_resolution	4	integer	
vert_resolution	4	integer	
reserved	4	integer	
predefined	2	integer	
name	32	string	
depth	2	integer	
predefined	2	integer	
clap	optional	box	≤1
pasp	optional	box	≤1

Table A.77 – Content element name for Audio coding

content	length	type	loop
reserved	6	integer	
data_reference	2	integer	
reserved	8	integer	
channel_count	2	integer	
sample_size	2	integer	
predefined	2	integer	
reserved	2	integer	
sample_rate	4	integer	

Table A.78 – Content element name for stsd

content	length	type	loop
version	1	integer	
flag	3	integer	
num_entries	4	integer	
mjp2|raw_|twos	–	box	⌋

Table A.79 – Content element name for stdp

content	length	type	loop
version	1	integer	
flag	3	integer	
priority	2	integer	⌋

Table A.80 – Content element name for stsl

content	length	type	loop
version	1	integer	
flag	3	integer	
constant_flag	1	integer	
scale_method	1	integer	
display_center_x	2	integer	
display_center_y	2	integer	

Table A.81 – Content element name for stts

content	length	type	loop
version	1	integer	
flag	3	integer	
num_entries	4	integer	
sample_count	4	integer	⌈
sample_delta	4	integer	⌋

Table A.82 – Content element name for ctts

content	length	type	loop
version	1	integer	
flag	3	integer	
num_entries	4	integer	
sample_count	4	integer	⌈
sample_offset	4	integer	⌋

Table A.83 – Content element name for stss

content	length	type	loop
version	1	integer	
flag	3	integer	
num_entries	4	integer	
sample_number	4	box	⌋

Table A.84 – Content element name for stsh

content	length	type	loop
version	1	integer	
flag	3	integer	
num_entries	4	integer	
shadowed_sample_number	4	integer	⌈
sync_sample_number	4	integer	⌋

Table A.85 – Content element name for sdtp

content	length	type	loop
version	1	integer	
flag	3	integer	
reserved	2	integer	⌈
sample_depends_on	2	integer	
sample_is_depends_on	2	integer	
sample_has_redundancy	2	integer	⌋

Table A.86 – Content element name for edts

content	length	type	loop
elst	optional	box	≤1

Table A.87 – Content element name for elst

content	length	type	loop
version	1	integer	
flag	3	integer	
num_entries	4	integer	
segment_duration	4, 8	integer	⎡
media_time	4, 8	integer	⎢
media_rate_integer	2	integer	⎢
media_rate_fraction	2	integer	⎣

Table A.88 – Content element name for dinf

content	length	type	loop
dref|url_|urn_	–	box	=1

Table A.89 – Content element name for url_

content	length	type	loop
version	1	integer	
flag	3	integer	
location	variable	string	

Table A.90 – Content element name for urn_

content	length	type	loop
version	1	integer	
flag	3	integer	
name	variable	string	
location	variable	string	

Table A.91 – Content element name for dref

content	length	type	loop
version	1	integer	
flag	3	integer	
num_entries	4	integer	
url_|urn_	–	box	⎦

Table A.92 – Content element name for stsz

content	length	type	loop
version	1	integer	
flag	3	integer	
sample_size	4	integer	
num_entries	4	integer	
entry_size	0, 4	integer	⎦

Table A.93 – Content element name for stz2

content	length	type	loop
version	1	integer	
flag	3	integer	
reserved	3	integer	
field_size	1	integer	
num_entries	4	integer	
entry_size	filed_size	integer	⎦

Table A.94 – Content element name for stsc

content	length	type	loop
version	1	integer	
flag	3	integer	
num_entries	4	integer	
first_chunk	4	integer	⎡
sample_per_chunk	4	integer	⎢
description_index	4	integer	⎣

Table A.95 – Content element name for stco

content	ength	type	loop
version	1	integer	
flag	3	integer	
num_entries	4	integer	
chunk_offset	4	integer	⎦

Table A.96 – Content element name for co64

content	length	type	loop
version	1	integer	
flag	3	integer	
num_entries	4	integer	
chunk_offset	8	integer	⎦

Table A.97 – Content element name for padb

content	length	type	loop
version	1	integer	
flag	3	integer	
pad1	4	integer	⎡
pad2	8	integer	⎣

Table A.98 – Content element name for subs

content	length	type	loop
version	1	integer	
flag	3	integer	
num_entries	4	integer	
sample_delta	4	integer	⎡ ⎤
num_subsample	2	integer	⎢ ⎥
subsample_size	4, 8	integer	⎡ ⎢
subsample_priority	1	integer	⎢ ⎢
discardable	1	integer	⎢ ⎢
reserved	4	integer	⎣ ⎣

Table A.99 – Content element name for mvex

content	length	type	loop
mehd	optional	box	≤1
trex	–	box	=1

Table A.100 – Content element name for mehd

content	length	type	loop
version	1	integer	
flag	3	integer	
fragment_duration	4, 8	integer	

Table A.101 – Content element name for trex

content	length	type	loop
version	1	integer	
flag	3	integer	
track_id	4	integer	
default_sample_index	4	integer	
default_sample_duration	4	integer	
default_sample_size	4	integer	
default_sample_flags	4	integer	

Table A.102 – Content element name for moof

content	length	type	loop
mfhd	–	box	
traf	–	box	

Table A.103 – Content element name for mfhd

content	length	type	loop
version	1	integer	
flag	3	integer	
sequence_number	4	integer	

Table A.104 – Content element name for traf

content	length	type	loop
tfhd	–	box	=1
traf	optional	box	≥0
trun	optional	box	≥0

Table A.105 – Content element name for tfhd

content	length	type	loop
version	1	integer	
flag	3	integer	
track_id	4	integer	
base_data_offset	8	integer	
sample_description_index	4	integer	
default_sample_duration	4	integer	
default_sample_size	4	integer	
default_sample_flags	4	integer	

Table A.106 – Content element name for trun

content	length	type	loop
version	1	integer	
flag	3	integer	
num_entries	4	integer	
data_offset	4	integer	
first_sample_flags	4	integer	
sample_duration	4	integer	⎡
sample_size	4	integer	⎢
sample_flags	4	integer	⎢
sample_time_offset	4	integer	⎣

Table A.107 – Content element name for mfra

content	length	type	loop
tfra	–	box	
mfro	–	box	

Table A.108 – Content element name for tfra

content	length	type	loop
version	1	integer	
flag	3	integer	
track_id	4	integer	
reserved	13/4	integer	
traf_index_size	1/4	integer	
trun_index_size	1/4	integer	
sample_index_size	1/4	integer	
num_entries	4	integer	
time	4, 8	integer	⎡
moof_offset	4, 8	integer	⎢
traf_index	traf_index_size+1	integer	⎢
trun_index	trun_index_size+1	integer	⎢
sample_index	sample_index_size+1	integer	⎣

Table A.109 – Content element name for sbgp

content	length	type	loop
version	1	integer	
flag	3	integer	
type	4	integer	
num_entries	4	integer	
sample_count	4	integer	⎡
group_description_index	4	integer	⎣

Table A.110 – Content element name for sgpd

content	length	type	loop
version	1	integer	
flag	3	integer	
type	4	integer	
default_length	0, 4	integer	
num_entries	4	integer	
description_length	0, 4	integer	⎡
data	variable	hexbyte	⎣

Table A.111 – Content element name for udta

content	length	type	loop
cprt	optional	box	≥0

Table A.112 – Content element name for cprt

content	length	type	loop
version	1	integer	
flag	3	integer	
language	2	integer	
notice	variable	string	

Table A.113 – Content element name for tsel

content	length	type	loop
version	1	integer	
flag	3	integer	
switch_group	4	integer	
attributes	4	integer	

Table A.114 – Content element name for meta

content	length	type	loop
version	1	integer	
flag	3	integer	
xml|bxml	optional	box	≤1
iloc	optional	box	≤1
pitm	optional	box	≤1
ipro	optional	box	≤1
iinf	optional	box	≤1
ipmc	optional	box	≤1
fiim	optional	box	≤1
hdlr	–	box	≤1
dinf	–	box	≤1

Table A.115 – Content element name for _xml_

content	length	type	loop
version	1	integer	
flag	3	integer	
text	variable	string	

Table A.116 – Content element name for bxml

content	length	type	loop
version	1	integer	
flag	3	integer	
data	variable	hexbyte	

Table A.117 – Content element name for iloc

content	length	type	loop
version	1	integer	
flag	3	integer	
offset_size	1/2	integer	
length_size	1/2	integer	
base_offset_size	1/2	integer	
reserved	1/2	integer	
num_entries	2	integer	
item_id	2	integer	⎡
data_reference	2	integer	⎢
base_offset	base_offset_size	integer	⎢
extent_count	2	integer	⎢
extent_offset	offset_size	integer	⎤ ⎢
extent_length	length_size	integer	⎦ ⎣

content	length	type	loop
version	1	integer	
flag	3	integer	
item_id	2	integer	

Table A.118 – Content element name for ipro

content	length	type	loop
version	1	integer	
flag	3	integer	
num_entries	2	integer	
sinf	–	box	⎦

Table A.119 – Content element name for infe

content	length	type	loop
version	1	integer	
flag	3	integer	
item_id	2	integer	
item_protection_index	2	integer	
item_name	variable	string	
content_type	variable	string	
content_encoding	variable	string	
type	0, 4	integer	
data	variable	box	⎦

Table A.120 – Content element name for iinf

content	length	type	loop
version	1	integer	
flag	3	integer	
num_entries	2	integer	
infe	variable	box	⎦

Table A.121– Content element name for meco

content	length	type	loop
mere	optional	box	≥0
meta	–	box	≥1

Table A.122 – Content element name for mere

content	length	type	loop
version	1	integer	
flag	3	integer	
first_metabox_type	4	integer	
second_metabox_type	4	integer	
metabox_relation	1	integer	

Table A.123 – Content element name for sinf

content	length	type	loop
frma	–	box	=1
imif	optional	box	≤1
schm	optional	box	≤1
schi	optional	box	≤1

Table A.124 – Content element name for frma

content	length	type	loop
data_format	4	integer	

Table A.125 – Content element name for imif

content	length	type	loop
version	1	integer	
flag	3	integer	
metabox_relation	1	hexbyte	]

Table A.126 – Content element name for ipmc

content	length	type	loop
version	1	integer	
flag	3	integer	
tool_list	–	hexbyte	
num_entries	1	integer	
metabox_relation	–	hexbyte	]

Table A.127 – Content element name for schm

content	length	type	loop
version	1	integer	
flag	3	integer	
type	4	integer	
version	4	integer	
location	variable	string	

Table A.128 – Content element name for schi

content	length	type	loop
Additional boxes	–	box	]

Table A.129 – Content element name for paen

content	length	type	loop
fpar	–	box	=1
fecr	optional	box	≤1

Table A.130 – Content element name for fiin

content	length	type	loop
version	1	integer	
flag	3	integer	
num_entries	2	integer	
paen	–	box	]
segr	optional	box	≤1
gitn	optional	box	≤1

Table A.131 – Content element name for fpar

content	length	type	loop
version	1	integer	
flag	3	integer	
item_id	2	integer	
packet_payload_size	2	integer	
reserved	1	integer	
FEC_encoding_id	1	integer	
FEC_instance_id	2	integer	
max_source_block_length	2	integer	
symbol_length	2	integer	
max_num_symbols	2	integer	
scheme_specific_info	variable	string	
num_entries	2	integer	
block_count	2	integer	]
block_size	4	integer	]

Table A.132 – Content element name for fecr

content	length	type	loop
version	1	integer	
flag	3	integer	
num_entries	2	integer	
item_id	2	integer	]
symbol_count	4	integer	]

Table A.133 – Content element name for segr

content	length	type	loop
num_entries	2	integer	
num_groups	1	integer	]
group_id	4	integer	]
num_hint_tracks	2	integer	
hint_track_id	4	integer	]]

Table A.134 – Content element name for gitn

content	length	type	loop
version	1	integer	
flag	3	integer	
num_entries	2	integer	
group_id	4	integer	]
group_name	variable	string	]

Table A.135 – Content element name for rtp_

content	length	type	loop
reserved	6	integer	
data_reference	2	integer	
track_version	2	integer	
compatible_version	2	integer	
max_packet_size	4	integer	
tims	–	box	]
tsro	optional	box	
snro	optional	box	]

Table A.136 – Content element name for srtp

content	length	type	loop
reserved	6	integer	
data_reference	2	integer	
track_version	2	integer	
compatible_version	2	integer	
max_packet_size	4	integer	
tims	–	box	]
srpp	–	box	
tsro	optional	box	
snro	optional	box	]

Table A.137 – Content element name for tims

content	length	type	loop
time_scale	4	integer	

Table A.138 – Content element name for tsro and snro

content	length	type	loop
offset	4	integer	

Table A.139 – Content element name for srpp

content	length	type	loop
version	1	integer	
flag	3	integer	
encryption_rtp	4	integer	
encryption_rtcp	4	integer	
integrity_rtp	4	integer	
integrity_rtcp	4	integer	
schm	–	box	=1
schi	–	box	≤1

Table A.140 – Content element name for rtpo

content	length	type	loop
offset	4	integer	

Table A.141 – Content element name for hnti

content	length	type	loop
hnti	–	box	≥0
sdp_	–	box	≥0

Table A.142 – Content element name for rtp_

content	length	type	loop
format	4	integer	
text	variable	string	

Table A.143 – Content element name for sdp_

content	length	type	loop
text	variable	string	

Table A.144 – Content element name for hinf

content	length	type	loop
trpy	optional	box	≤1
nump	optional	box	≤1
tpyl	optional	box	≤1
totl	optional	box	≤1
npck	optional	box	≤1
tpay	optional	box	≤1
maxr	optional	box	≥0
dmed	optional	box	≤1
dimm	optional	box	≤1
drep	optional	box	≤1
tmin	optional	box	≤1
tmax	optional	box	≤1
pmax	optional	box	≤1
dmax	optional	box	≤1
payt	optional	box	≤1

Table A.145 – Content element name for trpy, try1, dmed, dima and drep

content	length	type	loop
bytes_sent	8	integer	

Table A.146 – Content element name for tot1 and tpay

content	length	type	loop
bytes_sent	4	integer	

Table A.147 – Content element name for nump

content	length	type	loop
packets_sent	8	integer	

Table A.148 – Content element name for tmin, tmax and dmax

content	length	type	loop
time	4	integer	

Table A.149 – Content element name for npack

content	length	type	loop
packets_sent	4	integer	

Table A.150 – Content element name for pmax

content	length	type	loop
bytes_sent	4	integer	

Table A.151 – Content element name for maxr

content	length	type	loop
period	4	integer	
bytes_sent	4	integer	

Table A.152 – Content element name for hnti

content	length	type	loop
payload_id	4	integer	
count	1	integer	
rtpmap	count	integer	

Table A.153 – Content element name for fdp_

content	length	type	loop
reserved	6	integer	
data_reference	2	integer	
track_version	2	integer	
compatible_version	2	integer	
partition_entry_id	2	integer	
FEC_overhead	2	integer	
Additional boxes	optional	box	❑

Table A.154 – Content element name for fdsa

content	length	type	loop
fdpa	variable	box	≥1
extr	variable	box	≤1

Table A.155 – Content element name for fdpa

content	length	type	loop
header	3	hexbyte	
extension_size	2	integer	
extension	extension_size	hexbyte	
packet_size	2	integer	
packet	packet_size	hexbyte	

Table A.156 – Content element name for extr

content	length	type	loop
data	variable	hexbyte	

A.2.3 Format definitions for multiple-page documents

The box file formats defined in Rec. ITU-T T.805 | ISO/IEC 15444-6 are designed to represent a multiple-page image such as a document image. These content element names and types are defined in the following tables.

Table A.157 – Box element structure for a document image

Element name	Box name	Part 6	Defined
jP__	*JPEG family signature box*	required	1
ftyp	*File type and compatibility box*	required	1
jp2h	*JP2 header box (superbox)*	optional	1, 2
lbl_	*Label box*	optional	2, 6
ihdr	*Image header box*	optional	1, 2, 6
bpcc	*Bits per component box*	optional	1
colr	*Colour specification box*	optional	1, 2, 6
pclr	*Palette box*	optional	1
cmap	*Component mapping box*	optional	1
cdef	*Channel description box*	optional	1
res_	*Resolution box (superbox)*	optional	1
resc	*Capture resolution box*	optional	1
resd	*Display resolution box*	optional	1
mhdr	*Compound image header box*	required	6
dtbl	*Data reference box*	optional	2, 6
jp2c	*Contiguous codestream box*	optional	1, 2, 6
jp2i	*IPR box*	optional	1
xml_	*XML container box*	optional	1, 2, 12
uuid	*UUID box*	optional	1
uinf	*UUID info box (superbox)*	optional	1
ulst	*UUID list box*	optional	1
url_	*Data entry box*	optional	1
pcol	*Page collection box*	optional	6
ppcl	*Page collection locator box*	required	6
pagt	*Page table box*	required	6
sdat	*Shared data entry box*	optional	6
sref	*Shared data reference box*	optional	6
page	*Page box (superbox) (1)*	required	6
phdr	*Page header box*	required	6
ppcl	*Page collection locator box*	optional	6
sptr	*Spatial transformation box*	optional	6
phtx	*HTX Reference box*	optional	6
htxb	*Hidden text metadata box (superbox)*	optional	6
lobj	*Layout object box (superbox)*	optional	6
lhdr	*Layout object header box*	required	6
bclr	*Base colour box*	optional	6
bcvl	*Base colour value box*	required	6
colr	*Colour specification box*	optional	1, 2, 6
bpcc	*Bits per component box*	optional	1
objc	*Object box (superbox)*	required	6
ohdr	*Object header box*	required	6
scal	*Object scale box*	optional	6
jp2h	*JP2 header box (superbox)*	optional	1, 2, 6
ftbl	*Fragment table box (superbox)*	optional	2, 6
flst	*Fragment list box*	optional	2, 6
cref	*Cross-reference box (superbox)*	optional	2, 6

Table A.157 – Box element structure for a document image

Element name	Box name	Part 6	Defined
flst	*Fragment list box*	optional	2, 6
jp2c	*Contiguous codestream box for page (1)*	optional	1, 2, 6
page	*Page box (superbox) (2)*	optional	6
jp2c	*Contiguous codestream box for page (2)*	optional	1, 2, 6
htxb	*Hidden text metadata box (superbox)*	optional	6

A.2.3.1 Box element definitions for JPEG 2000 Part 6 format

Table A.158 – Content element name for mhdr

content	length	type	loop
num_pages	4	integer	
profile	1	integer	
self_contained	1	integer	
offset	8	integer	
length	4	integer	
mask_coder	1	integer	
image_coder	1	integer	
ipr	1	integer	

Table A.159 – Content element name for pcol

content		type	loop
lbl_	optional	box	≥0
meta	optional	box	≥0
pagt	–	box	=1

Table A.160 – Content element name for ppcl

content	length	type	loop
offset	8	integer	
length	4	integer	
data_reference	2	integer	
index	4	integer	

Table A.161– Content element name for pagt

content	length	type	loop
num_entries	4	integer	
offset	8	integer	]
length	4	integer	
data_reference	2	integer	
flag	1	hexbyte	]

Table A.162 – Content element name for sdal

content	length	type	loop
id	2	integer	
data	variable	hexbyte	

Table A.163 – Content element name for sref

content	length	type	loop
id	2	integer	

Table A.164 – Content element name for page

content	length	type	loop
phdr	–	box	=1
ppcl	–	box	≤1
res_	optional	box	≤1
bclr	optional	box	≤1
meta	optional	box	≥0
lbl_	optional	box	≥0
lobj	–	box	≥1

Table A.165 – Content element name for phdr

content	length	type	loop
num_lobjs	2	integer	
height	4	integer	
width	4	integer	
orientation	2	integer	
colour	2	integer	

Table A.166 – Content element name for lobj

content	length	type	loop
lhdr	–	box	=1
meta	optional	box	≤1
lbl_	optional	box	≥0
objc	–	box	=1
objc	optional	box	≥1

Table A.167 – Content element name for lhdr

content	length	type	loop
id	2	integer	
height	4	integer	
width	4	integer	
top	4	integer	
left	4	integer	
style	1	integer	

Table A.168 – Content element name for objc

content	length	type	loop
ohdr	–	box	=1
bclr	optional	box	≤1
meta	optional	box	≥0
lab_	optional	box	≥0
scal	optional	box	≤1
jp2h	optional	box	≤1

Table A.169 – Content element name for ohdr

content	length	type	loop
type	1	integer	
no_codestream	1	integer	
top	4	integer	
left	4	integer	
offset	8	integer	
length	4	integer	
data_reference	1	integer	

Table A.170 – Content element name for scal

content	length	type	loop
vert_ num	2	integer	
vert_ den	2	integer	
hori_ num	2	integer	
hori_ den	2	integer	

Table A.171 – Content element name for bclr

content	length	type	loop
bcvl	–	box	=1
colr	–	box	=1
bpcc	optional	box	≤1

Table A.172 – Content element name for bcvl

content	length	type	loop
num_components	2	integer	
depth	1	integer	
data	variable	integer	

Table A.173 – Content element name for htxb

content	length	type	loop
lbl_	optional	box	≤1
xml | uuid	–	box	≥1

Table A.174 – Content element name for phtx

content	length	type	loop
type	4	integer	
flst	–	box	=1
lbl_	optional	box	≤1

A.2.4 Security format definitions

The box file formats defined in Rec. ITU-T T.807 | ISO/IEC 15444-8 are for a secure image file format. These content element names and types are defined in the following tables.

Table A.175 – Box element structure for a motion/video image

Element name	Box name	Part 8	defined
moov	*Movie box (superbox)*		12
trak	*Track box (superbox)*		12
mdia	*Media box (superbox)*		12
minf	*Media information box (superbox)*		12
stbl	*Sample table box (superbox)*	use	12
stsd	*Sample descriptions box*	use	12
sces	*Scalable sample description entry box*	optional	8
dces	*Scalable sample description entry box*	optional	8
encv	*Scalable sample description entry box (enc video)*	optional	8
autv	*Scalable sample description entry box (auth video)*	optional	8
enct	*Scalable sample description entry box (enc text)*	optional	8
autt	*Scalable sample description entry box (auth text)*	optional	8
encs	*Scalable sample description entry box (enc sys)*	optional	8
auts	*Scalable sample description entry box (auth sys)*	optional	8
sbgp	*Sample-to-group box*	use	12
sgpd	*Sample group description box*	use	12
attr	*Scalable sample group entry box (media char)*	optional	8
prot	*Scalable sample group entry box (protection)*	optional	8
subs	*Sub-sample information box*		12
meta	*Metadata box*	use	12
meta	*Metadata box*	use	12
meta	*Metadata box*	use	12
iloc	*Item location box*	use	12
ipro	*Item protection box*	use	12

Table A.175 – Box element structure for a motion/video image

Element name	Box name	Part 8	defined
sinf	*Protection scheme information box (superbox)*	use	12
frma	*Original format box*	use	12
schm	*Scheme type box*	use	12
schi	*Scheme information box (superbox)*	use	12
gran	*Granularity box*	optional	8
vall	*Value list box*	optional	8
bcip	*Block cipher box*	optional	8
keyt	*Key template box*	optional	8
scip	*Stream cipher box*	optional	8
keyt	*Key template box*	optional	8
auth	*Authentication box*	optional	8
keyt	*Key template box*	optional	8
iinf	*Item information box*	use	12
ides	*Item description box*	optional	8
dest	*Description type box*	optional	8
desd	*Description data box*	optional	8
vide	*Visual item description entry box*	optional	8
j2ke	*JPEG 2000 item description entry box*	optional	8
icor	*Item correspondence box*	optional	8
meta	*Metadata box*	use	12
gprt	*Generic protected box*	optional	8

A.2.4.1 Box element definitions for JPEG 2000 Part 8 format

Table A.176 – Content element name for gprt

	content	length	type	loop
	type_flag	1/8	integer	
	size_flag	1/8	integer	
	location_flag	1/8	integer	
	reserved	5/8	integer	
if flags ≠ 0	num_entries	0, 4	integer	
	offset_size	0, 4	integer	
	box_length	0, 4	integer	
	box_type	0, 4	integer	
	box_ext_length	0, 8	integer	
	offset	0, offset_size	integer	
else	total_length	0, 4	integer	
	total_ext_length	0, 8	integer	

Table A.177 – Content element name for sces and decs

content	length	type	loop
reserved	6	integer	
data_reference	2	integer	
predefined	2	integer	
reserved	2	integer	
predefined	12	integer	
width	2	integer	
height	2	integer	
hori_resolution	4	integer	
vert_resolution	4	integer	
reserved	4	integer	
predefined	2	integer	
name	32	string	
depth	2	integer	
predefined	2	integer	
clap	optional	box	≤1
pasp	optional	box	≤1
res	1	integer	
layer	1	integer	
cropped_width	4	integer	
cropped_height	4	integer	
startx	0, 4	integer	
starty	0, 4	integer	

Table A.178 – Content element name for gran

content	length	type	loop
granularity	1	integer	

Table A.179 – Content element name for vall

content	length	type	loop
size	1	integer	
num_entries	2	integer	
count	2	integer	]
value	size	integer	]

Table A.180 – Content element name for keyt

content	length	type	loop
size	2	integer	
key_info	1	integer	
gran	optional	box	≤1
vall	–	box	=1

Table A.181 – Content element name for bcip

content	length	type	loop
cipher_id	2	integer	
cipher_mode	6/8	integer	
padding_mode	2/8	integer	
size	1	integer	
keyt	–	box	=1

Table A.182 – Content element name for scip

content	length	type	loop
type	1	integer	
cipher_id	2	integer	
keyt	–	box	=1

Table A.183 – Content element name for schi

content	length	type	loop
type	1	integer	
box_protected	1	integer	
bcip | scip	–	box	=1
gran	–	box	=1
vall	–	box	=1

Table A.184 – Content element name for auth

content	length	type	loop
type	1	integer	
method_id	1	integer	
hash_id	1	integer	
size	2	integer	
keyt	–	box	=1

Table A.185 – Content element name for schi

content	length	type	loop
type	1	integer	
auth	–	box	=1
gran	optional	box	≤1
vall	–	box	=1

Table A.186 – Content element name for ides

content	length	type	loop
num_entries	4	integer	
dest	–	box	]
item_id	4	integer	]
desd	–	box	]

Table A.187 – Content element name for vide

content	length	type	loop
layer_start	2	integer	
layer_count	2	integer	
res_start	2	integer	
res_count	2	integer	
hori_offset	2	integer	
hori_length	2	integer	
vert_offset	2	integer	
vert_length	2	integer	
colour	2	integer	
time_start	2	integer	
time_length	2	integer	

Table A.188 – Content element name for j2ke

content	length	type	loop
vide	optional	box	
tile_start	2	integer	
tile_count	2	integer	
precinct_start	2	integer	
precinct _count	2	integer	
j2k_packet_start	2	integer	
j2k_packet _count	2	integer	

Table A.189 – Content element name for dest

content	length	type	loop
type	4	integer	
version	4	integer	
location	–	string	

Table A.190 – Content element name for desd

content	length	type	loop
Additional box	–	box	]

Table A.191 – Content element name for icor

content	length	type	loop
item_id	2	integer	
desc_id	2	integer	

Table A.192 – Content element name for port and attr

content	length	type	loop
resolution	1	integer	
layer	1	integer	
cropped_width	4	integer	
cropped_height	4	integer	
startx	0, 4	integer	
starty	0, 4	integer	

A.3 Examples of XML schemas

The following examples are XML schemas for box types of JPEG 2000 family format.

A.3.1 Example of an XML schema for a common header

The following example is of common XML schemas for all box types of JPEG 2000 family format.

```
<?xml version="1.0" ?>
<xs:schema  xmlns:xs="http://www.iso.org/jpeg/2001/XMLSchema"
            targetNamespace="http://www.iso.org/jpxml"
            xmlns="http://www.iso.org/jpxml">
  <xs:attributeGroup name="attrs.box">
    <xs:attribute name="length"  type="xs:integer"   use="required" />
    <xs:attribute name="offset"  type="xs:integer"   use="optional" />
    <xs:attribute name="type"    use="required">
      <xs:simpleType>
        <xs:restriction base="xs:string">
          <xs:enumeration value="box" />
        </xs:restriction>
      </xs:simpleType>
    </xs:attribute>
  </xs:attributeGroup>
  <xs:attributeGroup name="attrs.4cc">
    <xs:attribute name="length"  type="xs:integer"   use="required" />
    <xs:attribute name="type"    use="required">
      <xs:simpleType>
        <xs:restriction base="xs:string">
          <xs:enumeration value="fourcc" />
        </xs:restriction>
      </xs:simpleType>
    </xs:attribute>
  </xs:attributeGroup>
  <xs:attributeGroup name="attrs.hex">
    <xs:attribute name="length"  type="xs:integer"   use="required" />
    <xs:attribute name="type"    use="required">
      <xs:simpleType>
        <xs:restriction base="xs:string">
          <xs:enumeration value="hexbyte" />
        </xs:restriction>
      </xs:simpleType>
    </xs:attribute>
  </xs:attributeGroup>
  <xs:attributeGroup name="attrs.int">
    <xs:attribute name="length"  type="xs:integer"   use="required" />
    <xs:attribute name="type"    use="required">
      <xs:simpleType>
        <xs:restriction base="xs:string">
          <xs:enumeration value="integer" />
        </xs:restriction>
      </xs:simpleType>
    </xs:attribute>
  </xs:attributeGroup>
  <xs:attributeGroup name="attrs.str">
    <xs:attribute name="length"  type="xs:integer"   use="required" />
    <xs:attribute name="type"    use="required">
      <xs:simpleType>
        <xs:restriction base="xs:string">
          <xs:enumeration value="string" />
        </xs:restriction>
      </xs:simpleType>
    </xs:attribute>
  </xs:attributeGroup>
  <xs:attributeGroup name="attrs.url">
    <xs:attribute name="length"  type="xs:integer"   use="required" />
    <xs:attribute name="type"    use="required">
      <xs:simpleType>
        <xs:restriction base="xs:string">
          <xs:enumeration value="location" />
        </xs:restriction>
      </xs:simpleType>
    </xs:attribute>
  </xs:attributeGroup>
  <xs:attributeGroup name="attrs.root">
    <xs:attribute name="length"  type="xs:integer"   use="required" />
    <xs:attribute name="name"    type="xs:anyURL"    use="optional" />
  </xs:attributeGroup>
  <!-- add following XML Schemas for the JPXML document. -->
  ...
</xs:schema>
```

A.3.2 Example of an XML schema for a JPEG 2000 Part 1 image (single image)

```
<!-- jpxml part 1 root element -->
<xs:element name="jpxml">
  <xs:complexType>
    <xs:attributeGroup ref="attrs.root" />
    <xs:sequence>
      <xs:element ref="jP  " />
      <xs:element ref="ftyp" />
      <xs:element ref="jp2h" />
      <xs:element ref="jp2i" minOccurs="0" />
      <xs:element ref=" xml " minOccurs="0" />
      <xs:element ref="uuid" minOccurs="0" />
      <xs:element ref="uinf" minOccurs="0" />
      <xs:element ref="jp2c" />
    </xs:sequence>
  </xs:complexType>
</xs:element>

<!-- part 1 box element -->
<xs:element name="jP__">
  <xs:complexType>
    <xs:attributeGroup ref="attrs.box"
                       use="required" />
    <xs:sequence>
      <xs:element ref="signature" />
    </xs:sequence>
  </xs:complexType>
</xs:element>

<xs:element name="ftyp">
  <xs:complexType>
    <xs:attributeGroup ref="attrs.box"
                       use="required" />
    <xs:sequence>
      <xs:element ref="brand" />
      <xs:element ref="version" />
      <xs:element ref="compatibility"
                  maxOccurs="unbounded" />
    </xs:sequence>
  </xs:complexType>
</xs:element>

<xs:element name="jp2h">
  <xs:complexType>
    <xs:attributeGroup ref="attrs.box"
                       use="required" />
    <xs:sequence>
      <xs:element ref="ihdr" />
      <xs:element ref="bpcc" minOccurs="0" />
      <xs:element ref="colr" />
      <xs:element ref="pclr" minOccurs="0" />
      <xs:element ref="cmap" minOccurs="0" />
      <xs:element ref="cdef" minOccurs="0" />
      <xs:element ref="res_" minOccurs="0" />
    </xs:sequence>
  </xs:complexType>
</xs:element>

<xs:element name="ihdr">
  <xs:complexType>
    <xs:attributeGroup ref="attrs.box"
                       use="required" />
    <xs:sequence>
      <xs:element ref="height">
        <xs:attributeGroup ref="attrs.int"
                           use="required" />
      </xs:element>
      <xs:element ref="width" />
      <xs:element ref="num_components" />
      <xs:element ref="depth" />
      <xs:element ref="compression" />
      <xs:element ref="colour unknown" />
      <xs:element ref="ipr" />
    </xs:sequence>
  </xs:complexType>
</xs:element>

<xs:element name="colr">
  <xs:complexType>
    <xs:attributeGroup ref="attrs.box"
                       use="required" />
    <xs:sequence>
      <xs:element ref="method" />
      <xs:element ref="precedence" />
      <xs:element ref="approx" />
      <xs:element ref="colour" />
    </xs:sequence>
  </xs:complexType>
</xs:element>
<xs:element name="bpcc">
  <xs:complexType>
    <xs:attributeGroup ref="attrs.box"
                       use="required" />
    <xs:sequence>
      <xs:element ref="depth"
                  maxOccurs="unbounded" />
    </xs:sequence>
  </xs:complexType>
</xs:element>

<xs:element name="pclr">
  <xs:complexType>
    <xs:attributeGroup ref="attrs.box"
                       use="required" />
    <xs:sequence>
      <xs:element ref="num_entries" />
      <xs:element ref="num_components" />
      <xs:element ref="depth"
                  maxOccurs="unbounded" />
      <xs:element ref="data"
                  maxOccurs="unbounded" />
    </xs:sequence>
  </xs:complexType>
</xs:element>

<xs:element name="cdef">
  <xs:complexType>
    <xs:attributeGroup ref="attrs.box"
                       use="required" />
    <xs:sequence>
      <xs:element ref="num_entries" />
      <xs:sequence maxOccurs="unbounded">
        <xs:element ref="index" />
        <xs:element ref="type" />
        <xs:element ref="assoc" />
      </xs:sequence>
    </xs:sequence>
  </xs:complexType>
</xs:element>

<xs:element name="res_">
  <xs:complexType>
    <xs:attributeGroup ref="attrs.box"
                       use="required" />
    <xs:sequence>
      <xs:element ref="resc" minOccurs="0" />
      <xs:element ref="resd" minOccurs="0" />
    </xs:sequence>
  </xs:complexType>
</xs:element>

<xs:element name="resd">
  <xs:complexType>
    <xs:attributeGroup ref="attrs.box"
                       use="required" />
    <xs:sequence>
      <xs:element ref="vert num" />
      <xs:element ref="vert_den" />
      <xs:element ref="hori num" />
      <xs:element ref="hori den" />
      <xs:element ref="vert exp" />
      <xs:element ref="hori exp" />
    </xs:sequence>
  </xs:complexType>
</xs:element>

<xs:element name="resd">
  <xs:complexType>
    <xs:attributeGroup ref="attrs.box"
                       use="required" />
    <xs:sequence>
      <xs:element ref="vert num" />
      <xs:element ref="vert_den" />
      <xs:element ref="hori num" />
      <xs:element ref="hori_den" />
      <xs:element ref="vert exp" />
      <xs:element ref="hori exp" />
    </xs:sequence>
  </xs:complexType>
```

```
</xs:element>

<xs:element name="uuid">
  <xs:complexType>
    <xs:attributeGroup ref="attrs.box"
                       use="required" />
    <xs:sequence>
      <xs:element ref="id" />
      <xs:element ref="data" />
    </xs:sequence>
  </xs:complexType>
</xs:element>

<xs:element name="uinf">
  <xs:complexType>
    <xs:attributeGroup ref="attrs.box"
                       use="required" />
    <xs:sequence>
      <xs:element ref="ulst" />
      <xs:element ref="url_" />
    </xs:sequence>
  </xs:complexType>
</xs:element>

<xs:element name="ulst">
  <xs:complexType>
    <xs:attributeGroup ref="attrs.box"
                       use="required" />
    <xs:sequence>
      <xs:element ref="num entries" />
      <xs:element ref="uuid" />
    </xs:sequence>
  </xs:complexType>
</xs:element>

<xs:element name="url ">
  <xs:complexType>
    <xs:attributeGroup ref="attrs.box"
                       use="required" />
    <xs:sequence>
      <xs:element ref="location" />
    </xs:sequence>
  </xs:complexType>
</xs:element>

<xs:element name=" xml ">
  <xs:complexType>
    <xs:attributeGroup ref="attrs.box"
                       use="required" />
    <xs:sequence>
      <xs:element ref="text" />
    </xs:sequence>
  </xs:complexType>
</xs:element>

<xs:element name="jp2c">
  <xs:complexType>
    <xs:attributeGroup ref="attrs.box"
                       use="required" />
    <xs:sequence>
      <xs:element ref="data" />
    </xs:sequence>
  </xs:complexType>
</xs:element>

<!-- part 1 content element -->

<xs:element name="signature" type="hexbyte">
  <xs:attributeGroup ref="attrs.hex"
                     use="required" />
</xs:element>

<xs:element name="brand" type="fourcc">
  <xs:attributeGroup ref="attrs.4cc"
                     use="required" />
</xs:element>

<xs:element name="version" type="xs:integer">
  <xs:attributeGroup ref="attrs.int"
                     use="required" />
</xs:element>

<xs:element name="compatibility" type="fourcc">
  <xs:attributeGroup ref="attrs.4cc"
                     use="required" />
</xs:element>
```

```
<xs:element name="width" type="xs:integer">
  <xs:attributeGroup ref="attrs.int"
                     use="required" />
</xs:element>

<xs:element name="num components"
            type="xs:integer">
  <xs:attributeGroup ref="attrs.int"
                     use="required" />
</xs:element>

<xs:element name="depth" type="xs:integer">
  <xs:attributeGroup ref="attrs.int"
                     use="required" />
</xs:element>

<xs:element name="compression"
            type="xs:integer">
  <xs:attributeGroup ref="attrs.int"
                     use="required" />
</xs:element>

<xs:element name="colour unknown"
            type="xs:integer">
  <xs:attributeGroup ref="attrs.int"
                     use="required" />
</xs:element>

<xs:element name="ipr" type="xs:integer">
  <xs:attributeGroup ref="attrs.int"
                     use="required" />
</xs:element>

<xs:element name="method" type="xs:integer">
  <xs:attributeGroup ref="attrs.int"
                     use="required" />
</xs:element>

<xs:element name="precedence" type="xs:integer">
  <xs:attributeGroup ref="attrs.int"
                     use="required" />
</xs:element>

<xs:element name="approx" type="xs:integer">
  <xs:attributeGroup ref="attrs.int"
                     use="required" />
</xs:element>

<xs:element name="colour" type="hexbyte">
  <xs:attributeGroup ref="attrs.hex"
                     use="required" />
</xs:element>

<xs:element name="num_entries"
            type="xs:integer">
  <xs:attributeGroup ref="attrs.int"
                     use="required" />
</xs:element>

<xs:element name="depth" type="xs:integer">
  <xs:attributeGroup ref="attrs.int"
                     use="required" />
</xs:element>

<xs:element name="data" type="hexbyte">
  <xs:attributeGroup ref="attrs.hex"
                     use="required" />
</xs:element>
<xs:element name="index" type="xs:integer">
  <xs:attributeGroup ref="attrs.int"
                     use="required" />
</xs:element>

<xs:element name="type" type="xs:integer">
  <xs:attributeGroup ref="attrs.int"
                     use="required" />
</xs:element>

<xs:element name="assoc" type="hexbyte">
  <xs:attributeGroup ref="attrs.hex"
                     use="required" />
</xs:element>

<xs:element name="vert_num" type="xs:integer">
  <xs:attributeGroup ref="attrs.int"
                     use="required" />
</xs:element>
```

```
<xs:element name="vert_den" type="xs:integer">
  <xs:attributeGroup ref="attrs.int"
                     use="required" />
</xs:element>

<xs:element name="hori_num" type="xs:integer">
  <xs:attributeGroup ref="attrs.int"
                     use="required" />
</xs:element>

<xs:element name="hori_den" type="xs:integer">
  <xs:attributeGroup ref="attrs.int"
                     use="required" />
</xs:element>

<xs:element name="vert_exp" type="xs:integer">
  <xs:attributeGroup ref="attrs.int"
                     use="required" />
</xs:element>
```

```
<xs:element name="hori_exp" type="xs:integer">
  <xs:attributeGroup ref="attrs.int"
                     use="required" />
</xs:element>

<xs:element name="id" type="xs:integer">
  <xs:attributeGroup ref="attrs.int"
                     use="required" />
</xs:element>

<xs:element name="location" type="location">
  <xs:attributeGroup ref="attrs.url"
                     use="required" />
</xs:element>

<xs:element name="text" type="xs:string">
  <xs:attributeGroup ref="attrs.str"
                     use="required" />
</xs:element>
```

A.3.3 Example of an XML schema for a JPEG 2000 Part 2 image (single/layered image)

```
<!-- jpxml part 2 root element -->
<xs:element name="jpxml">
  <xs:complexType>
    <xs:attributeGroup ref="attrs.root" />
    <xs:sequence>
      <xs:element ref="jP  " />
      <xs:element ref="ftyp" />
      <xs:element ref="rreq" />
      <xs:element ref="jp2h" />
      <xs:element ref="jpch" minOccurs="0" />
      <xs:element ref="jplh" minOccurs="0" />
      <xs:element ref="dtbl" minOccurs="0" />
      <xs:element ref="ftbl" minOccurs="0" />
      <xs:element ref="jp2i" minOccurs="0" />
      <xs:element ref="jp2c" minOccurs="0" />
      <xs:element ref="mdat" minOccurs="0" />
      <xs:element ref="comp" minOccurs="0" />
      <xs:element ref="drep" minOccurs="0" />
      <xs:element ref="roid" minOccurs="0" />
      <xs:element ref="cref" minOccurs="0" />
      <xs:element ref="asoc" minOccurs="0" />
      <xs:element ref="bfil" minOccurs="0" />
      <xs:element ref="chck" minOccurs="0" />
      <xs:element ref="mp7b" minOccurs="0" />
      <xs:element ref="free" minOccurs="0" />
      <xs:element ref=" xml " minOccurs="0" />
      <xs:element ref="uuid" minOccurs="0" />
      <xs:element ref="uinf" minOccurs="0" />
      <xs:element ref="jclx" minOccurs="0" />
      <xs:element ref="j2cx" minOccurs="0" />
      <xs:element ref="grp " minOccurs="0" />
    </xs:sequence>
  </xs:complexType>
</xs:element>

<!-- part 1 box element -->
…
<!-- part 2 box element -->
<xs:element name="rreq">
  <xs:complexType>
    <xs:attributeGroup ref="attrs.box"
                       use="required" />
    <xs:sequence>
      <xs:element ref="length" />
      <xs:element ref="fua mask" />
      <xs:element ref="dc mask" />
      <xs:element ref="num std flags" />
      <xs:sequence maxOccurs="unbounded">
        <xs:element ref="std_flag" />
        <xs:element ref="std mask" />
      </xs:sequence>
      <xs:element ref="num vender features" />
      <xs:sequence maxOccurs="unbounded">
        <xs:element ref="vender_feature" />
        <xs:element ref="vender mask" />
      </xs:sequence>
    </xs:sequence>
  </xs:complexType>
</xs:element>

<xs:element name="dtbl">
```

```
  <xs:complexType>
    <xs:attributeGroup ref="attrs.box"
                       use="required" />
    <xs:sequence>
      <xs:element ref="num_entries" />
      <xs:sequence maxOccurs="unbounded">
        <xs:element ref="location" />
      </xs:sequence>
    </xs:sequence>
  </xs:complexType>
</xs:element>

<xs:element name="ftbl">
  <xs:complexType>
    <xs:attributeGroup ref="attrs.box"
                       use="required" />
    <xs:sequence maxOccurs="unbounded">
      <xs:element ref="flst" />
    </xs:sequence>
  </xs:complexType>
</xs:element>

<xs:element name="flst">
  <xs:complexType>
    <xs:attributeGroup ref="attrs.box"
                       use="required" />
    <xs:sequence>
      <xs:element ref="num entries" />
      <xs:sequence maxOccurs="unbounded">
        <xs:element ref="offset" />
        <xs:element ref="length" />
        <xs:element ref="data_reference" />
      </xs:sequence>
    </xs:sequence>
  </xs:complexType>
</xs:element>

<xs:element name="cref">
  <xs:complexType>
    <xs:attributeGroup ref="attrs.box"
                       use="required" />
    <xs:sequence>
      <xs:element ref="type" />
      <xs:element ref="flst" />
    </xs:sequence>
  </xs:complexType>
</xs:element>

<xs:element name="jpch">
  <xs:complexType>
    <xs:attributeGroup ref="attrs.box"
                       use="required" />
    <xs:sequence>
      <xs:element ref="lbl_" minOccurs="0" />
      <xs:element ref="ihdr" />
      <xs:element ref="bpcc" minOccurs="0" />
      <xs:element ref="pclr" minOccurs="0" />
      <xs:element ref="cmap" minOccurs="0" />
      <xs:element ref="roid" minOccurs="0" />
    </xs:sequence>
  </xs:complexType>
</xs:element>
```

```
<xs:element name="jplh">
  <xs:complexType>
    <xs:attributeGroup ref="attrs.box"
                       use="required" />
    <xs:sequence>
      <xs:element ref="lbl " minOccurs="0" />
      <xs:element ref="cgrp" minOccurs="0" />
      <xs:element ref="opct" minOccurs="0" />
      <xs:element ref="cdef" minOccurs="0" />
      <xs:element ref="creq" minOccurs="0" />
      <xs:element ref="res " minOccurs="0" />
    </xs:sequence>
  </xs:complexType>
</xs:element>

<xs:element name="colour">
  <xs:complexType>
    <xs:attributeGroup ref="attrs.hex"
                       use="required" />
    <xs:choice>
      <xs:sequence>
        <xs:element ref="enum_colour" />
        <xs:element ref="enum param" />
      </xs:sequence>
      <xs:sequence>
        <xs:element ref="vender colour" />
        <xs:element ref="vendar_param" />
      </xs:sequence>
    </xs:choice>
  </xs:complexType>
</xs:element>

<xs:element name="enum param">
  <xs:complexType>
    <xs:attributeGroup ref="attrs.hex"
                       use="required" />
    <xs:choice>
      <xs:sequence>
        <xs:element ref="range_L" />
        <xs:element ref="offset L" />
        <xs:element ref="range a" />
        <xs:element ref="offset_a" />
        <xs:element ref="range b" />
        <xs:element ref="offset_b" />
        <xs:element ref="illuminant" />
      </xs:sequence>
      <xs:sequence>
        <xs:element ref="range J" />
        <xs:element ref="offset_J" />
        <xs:element ref="range a" />
        <xs:element ref="offset a" />
        <xs:element ref="range_b" />
        <xs:element ref="offset b" />
      </xs:sequence>
    </xs:choice>
  </xs:complexType>
</xs:element>

<xs:element name="cgrp">
  <xs:complexType>
    <xs:attributeGroup ref="attrs.box"
                       use="required" />
    <xs:sequence maxOccurs="unbounded">
      <xs:element ref="colr" />
    </xs:sequence>
  </xs:complexType>
</xs:element>

<xs:element name="opct">
  <xs:complexType>
    <xs:attributeGroup ref="attrs.box"
                       use="required" />
    <xs:sequence>
      <xs:element ref="type" />
      <xs:element ref="num entries" />
      <xs:element ref="data" />
    </xs:sequence>
  </xs:complexType>
</xs:element>

<xs:element name="creg">
  <xs:complexType>
    <xs:attributeGroup ref="attrs.box"
                       use="required" />
    <xs:sequence>
      <xs:element ref="hori_size" />
      <xs:element ref="vert_size" />
      <xs:element ref="cds num" />
      <xs:element ref="hori resolution" />
      <xs:element ref="vert resolution" />
      <xs:element ref="hori offset" />
      <xs:element ref="vert_offset" />
      <xs:element ref="num entries" />
    </xs:sequence>
  </xs:complexType>
</xs:element>

<xs:element name="free">
  <xs:complexType>
    <xs:attributeGroup ref="attrs.box"
                       use="required" />
    <xs:sequence>
      <xs:element ref="data" />
    </xs:sequence>
  </xs:complexType>
</xs:element>

<xs:element name="comp">
  <xs:complexType>
    <xs:attributeGroup ref="attrs.box"
                       use="required" />
    <xs:sequence>
      <xs:element ref="copt" />
      <xs:sequence maxOccurs="unbounded">
        <xs:element ref="inst" />
      </xs:sequence>
    </xs:sequence>
  </xs:complexType>
</xs:element>

<xs:element name="copt">
  <xs:complexType>
    <xs:attributeGroup ref="attrs.box"
                       use="required" />
    <xs:sequence>
      <xs:element ref="height" />
      <xs:element ref="width" />
      <xs:element ref="loop" />
    </xs:sequence>
  </xs:complexType>
</xs:element>

<xs:element name="inst">
  <xs:complexType>
    <xs:attributeGroup ref="attrs.box"
                       use="required" />
    <xs:sequence>
      <xs:element ref="type" />
      <xs:element ref="repeition" />
      <xs:element ref="duration" />
      <xs:element ref="instruction" />
    </xs:sequence>
  </xs:complexType>
</xs:element>

<xs:element name="instruction" type="hexbyte">
  <xs:attributeGroup ref="attrs.hex"
                     use="required" />
    <xs:sequence>
      <xs:element ref="hori offset" />
      <xs:element ref="vert_offset" />
      <xs:element ref="width" />
      <xs:element ref="height" />
      <xs:element ref="life" />
      <xs:element ref="next use" />
      <xs:element ref="vert_crop_offset" />
      <xs:element ref="hori crop offset" />
      <xs:element ref="crop_width" />
      <xs:element ref="crop height" />
    </xs:sequence>
</xs:element>

<xs:element name="nlst">
  <xs:complexType>
    <xs:attributeGroup ref="attrs.box"
                       use="required" />
    <xs:sequence maxOccurs="unbounded">
      <xs:element ref="asoc_num" />
    </xs:sequence>
  </xs:complexType>
</xs:element>

<xs:element name="lbl ">
  <xs:complexType>
```

```
    <xs:attributeGroup ref="attrs.box"
                       use="required" />
    <xs:sequence>
      <xs:element ref="label" />
    </xs:sequence>
  </xs:complexType>
</xs:element>

<xs:element name="bfil">
  <xs:complexType>
    <xs:attributeGroup ref="attrs.box"
                       use="required" />
    <xs:sequence>
      <xs:element ref="type" />
      <xs:element ref="data" />
    </xs:sequence>
  </xs:complexType>
</xs:element>

<xs:element name="drep">
  <xs:complexType>
    <xs:attributeGroup ref="attrs.box"
                       use="required" />
    <xs:sequence>
      <xs:element ref="gtso" />
    </xs:sequence>
  </xs:complexType>
</xs:element>

<xs:element name="gtso">
  <xs:complexType>
    <xs:attributeGroup ref="attrs.box"
                       use="required" />
    <xs:sequence>
      <xs:element ref="profile" />
    </xs:sequence>
  </xs:complexType>
</xs:element>

<xs:element name="roid">
  <xs:attributeGroup ref="attrs.box"
                     use="required" />
    <xs:sequence>
      <xs:element ref="num_entries" />
      <xs:element ref="contained" />
      <xs:element ref="type" />
      <xs:element ref="priority" />
      <xs:element ref="left" />
      <xs:element ref="top" />
      <xs:element ref="width" />
      <xs:element ref="height" />
    </xs:sequence>
</xs:element>

<xs:element name="chck">
  <xs:attributeGroup ref="attrs.box"
                     use="required" />
    <xs:sequence>
```

```
      <xs:element ref="signature type" />
      <xs:element ref="source type" />
      <xs:element ref="offset" />
      <xs:element ref="length" />
      <xs:element ref="data" />
    </xs:sequence>
</xs:element>

<xs:element name="mp7b">
  <xs:attributeGroup ref="attrs.box"
                     use="required" />
    <xs:sequence>
      <xs:element ref="data" />
    </xs:sequence>
</xs:element>

<!-- part 1 content element -->
 …
<!-- part 3 content element -->
<xs:element name="length"
              type="xs:integer">
  <xs:attributeGroup ref="attrs.int"
                     use="required" />
</xs:element>

<xs:element name="offset" type="xs:integer">
  <xs:attributeGroup ref="attrs.int"
                     use="required" />
</xs:element>

<xs:element name="data reference"
             type="xs:integer">
  <xs:attributeGroup ref="attrs.int"
                     use="required" />
</xs:element>

<xs:element name="enum_colour"
             type="xs:integer">
  <xs:attributeGroup ref="attrs.int"
                     use="required" />
</xs:element>

<xs:element name="enum param" type="hexbyte">
  <xs:attributeGroup ref="attrs.hex"
                     use="required" />
</xs:element>

<xs:element name="vender colour"
             type="xs:integer">
  <xs:attributeGroup ref="attrs.int"
                     use="required" />
</xs:element>

<xs:element name="vendar_param" type="hexbyte">
  <xs:attributeGroup ref="attrs.hex"
                     use="required" />
</xs:element>
```

A.3.4 Example of an XML schema for a JPEG 2000 Part 3 image (motion image)

```
<!-- jpxml part 3 root element -->
<xs:element name="jpxml">
  <xs:complexType>
    <xs:attributeGroup ref="attrs.root" />
    <xs:sequence>
      <xs:element ref="jP  " />
      <xs:element ref="ftyp" />
      <xs:element ref="pdin" minOccurs="0" />
      <xs:element ref="moov" />
      <xs:element ref="mdat" minOccurs="0" />
      <xs:element ref="moof" minOccurs="0" />
      <xs:element ref="mfra" minOccurs="0" />
      <xs:element ref="meta" minOccurs="0" />
      <xs:element ref="meco" minOccurs="0" />
      <xs:element ref="skip" minOccurs="0" />
      <xs:element ref="free" minOccurs="0" />
    </xs:sequence>
  </xs:complexType>
</xs:element>

<!-- part 1 box element -->
<!-- part 12 box element -->
…
<!-- part 3 box element -->
```

```
<xs:element name="mjp2">
  <xs:complexType>
    <xs:attributeGroup ref="attrs.box"
                       use="required" />
    <xs:sequence>
      <xs:element ref="reserved" />
      <xs:element ref="data_reference" />
      <xs:element ref="predefined" />
      <xs:element ref="reserved" />
      <xs:element ref="predefined" />
      <xs:element ref="width" />
      <xs:element ref="height" />
      <xs:element ref="hori resolution" />
      <xs:element ref="vert resolution" />
      <xs:element ref="reserved" />
      <xs:element ref="predefined" />
      <xs:element ref="name" />
      <xs:element ref="depth" />
      <xs:element ref="predefined" />
      <xs:element ref="length" />
      <xs:element ref="jp2h" />
      <xs:element ref="fiel" minOccurs="0" />
      <xs:element ref="jp2p" minOccurs="0" />
      <xs:element ref="jp2x" minOccurs="0" />
      <xs:element ref="jsub" minOccurs="0" />
```

```
      <xs:element ref="orfo" minOccurs="0" />
    </xs:sequence>
  </xs:complexType>
</xs:element>

<xs:element name="fiel">
  <xs:complexType>
    <xs:attributeGroup ref="attrs.box"
                       use="required" />
    <xs:sequence>
      <xs:element ref="field count" />
      <xs:element ref="field_order" />
    </xs:sequence>
  </xs:complexType>
</xs:element>

<xs:element name="orfo">
  <xs:complexType>
    <xs:attributeGroup ref="attrs.box"
                       use="required" />
    <xs:sequence>
      <xs:element ref="original_field_count" />
      <xs:element ref="original field order" />
    </xs:sequence>
  </xs:complexType>
</xs:element>

<xs:element name="jp2p">
  <xs:complexType>
    <xs:attributeGroup ref="attrs.box"
                       use="required" />
    <xs:sequence>
      <xs:element ref="version" />
      <xs:element ref="flag" />
      <xs:element ref="compatibility" />
    </xs:sequence>
  </xs:complexType>
</xs:element>

<xs:element name="jp2x">
  <xs:complexType>
    <xs:attributeGroup ref="attrs.box"
                       use="required" />
    <xs:sequence>
      <xs:element ref="data" />
    </xs:sequence>
  </xs:complexType>
</xs:element>

<xs:element name="jsub">
  <xs:complexType>
    <xs:attributeGroup ref="attrs.box"
                       use="required" />
    <xs:sequence>
      <xs:element ref="hori sub" />
      <xs:element ref="vert sub" />
      <xs:element ref="hori offset" />
      <xs:element ref="vert offset" />
    </xs:sequence>
  </xs:complexType>
</xs:element>

<xs:element name="raw ">
  <xs:complexType>
    <xs:attributeGroup ref="attrs.box"
                       use="required" />
    <xs:sequence>
      <xs:element ref="reserved" />
      <xs:element ref="data_reference" />
      <xs:element ref="reserved" />
      <xs:element ref="channel_count" />
      <xs:element ref="sample size" />
      <xs:element ref="predefined" />
      <xs:element ref="reserved" />
      <xs:element ref="sample rate" />
    </xs:sequence>
  </xs:complexType>
</xs:element>

<!-- part 3 content element -->
<xs:element name="reserved" type="xs:integer">
  <xs:attributeGroup ref="attrs.int"
                     use="required" />
</xs:element>
```

```
<xs:element name="predefined"
            type="xs:integer">
  <xs:attributeGroup ref="attrs.int"
                     use="required" />
</xs:element>

<xs:element name="hori resolution"
            type="xs:integer">
  <xs:attributeGroup ref="attrs.int"
                     use="required" />
</xs:element>

<xs:element name="vert resolution"
            type="xs:integer">
  <xs:attributeGroup ref="attrs.int"
                     use="required" />
</xs:element>

<xs:element name="name" type="xs:integer">
  <xs:attributeGroup ref="attrs.int"
                     use="required" />
</xs:element>

<xs:element name="field_count"
            type="xs:integer">
  <xs:attributeGroup ref="attrs.int"
                     use="required" />
</xs:element>

<xs:element name="field order"
            type="xs:integer">
  <xs:attributeGroup ref="attrs.int"
                     use="required" />
</xs:element>

<xs:element name="original_field_count"
            type="xs:integer">
  <xs:attributeGroup ref="attrs.int"
                     use="required" />
</xs:element>

<xs:element name="original field order"
            type="xs:integer">
  <xs:attributeGroup ref="attrs.int"
                     use="required" />
</xs:element>

<xs:element name="version" type="xs:integer">
  <xs:attributeGroup ref="attrs.int"
                     use="required" />
</xs:element>

<xs:element name="flag" type="xs:integer">
  <xs:attributeGroup ref="attrs.int"
                     use="required" />
</xs:element>

<xs:element name="hori_sub" type="xs:integer">
  <xs:attributeGroup ref="attrs.int"
                     use="required" />
</xs:element>

<xs:element name="vert_sub" type="xs:integer">
  <xs:attributeGroup ref="attrs.int"
                     use="required" />
</xs:element>

<xs:element name="channel_count"
            type="xs:integer">
  <xs:attributeGroup ref="attrs.int"
                     use="required" />
</xs:element>

<xs:element name="sample size"
            type="xs:integer">
  <xs:attributeGroup ref="attrs.int"
                     use="required" />
</xs:element>
<xs:element name="sample rate"
            type="xs:integer">
  <xs:attributeGroup ref="attrs.int"
                     use="required" />
</xs:element>
```

A.3.5 Example of an XML schema for a JPEG 2000 Part 6 image (document image)

```
<!-- jpxml part 6 root element -->
<xs:element name="jpxml">
  <xs:complexType>
    <xs:attributeGroup ref="attrs.root" />
    <xs:sequence>
      <xs:element ref="jP__" />
      <xs:element ref="ftyp" />
      <xs:element ref="jp2h" minOccurs="0" />
      <xs:element ref="mhdr" />
      <xs:element ref="pcol" minOccurs="0" />
      <xs:element ref="page" />
      <xs:element ref="sdat" minOccurs="0" />
      <xs:element ref="dtbl" minOccurs="0" />
      <xs:element ref="ftbl" minOccurs="0" />
      <xs:element ref="cref" minOccurs="0" />
      <xs:element ref="jp2i" minOccurs="0" />
      <xs:element ref="jp2c" minOccurs="0" />
      <xs:element ref="_xml_" minOccurs="0" />
      <xs:element ref="uuid" minOccurs="0" />
      <xs:element ref="uinf" minOccurs="0" />
      <xs:element ref="free" minOccurs="0" />
      <xs:element ref="htxb" minOccurs="0" />
    </xs:sequence>
  </xs:complexType>
</xs:element>

<!-- part 1 box element -->
<!-- part 2 box element -->
…
<!-- part 6 box element -->
<xs:element name="mhdr">
  <xs:complexType>
    <xs:attributeGroup ref="attrs.box"
                       use="required" />
    <xs:sequence>
      <xs:element ref="num pages" />
      <xs:element ref="profile" />
      <xs:element ref="self_contained" />
      <xs:element ref="offset" />
      <xs:element ref="length" />
      <xs:element ref="mask coder" />
      <xs:element ref="image coder" />
      <xs:element ref="ipr" />
    </xs:sequence>
  </xs:complexType>
</xs:element>

<xs:element name="pcol">
  <xs:complexType>
    <xs:attributeGroup ref="attrs.box"
                       use="required" />
    <xs:sequence>
      <xs:element ref="lbl_"
                  minOccurs="0" />
      <xs:element ref="meta"
                  minOccurs="0" />
      <xs:element ref="pagt" />
    </xs:sequence>
  </xs:complexType>
</xs:element>

<xs:element name="ppcl">
  <xs:complexType>
    <xs:attributeGroup ref="attrs.box"
                       use="required" />
    <xs:sequence>
      <xs:element ref="offset" />
      <xs:element ref="length" />
      <xs:element ref="data reference" />
      <xs:element ref="index" />
    </xs:sequence>
  </xs:complexType>
</xs:element>

<xs:element name="pagt">
  <xs:complexType>
    <xs:attributeGroup ref="attrs.box"
                       use="required" />
    <xs:sequence>
      <xs:element ref="num_entries" />
      <xs:sequence maxOccurs="unbounded">
        <xs:element ref="offset" />
        <xs:element ref="length" />
        <xs:element ref="data reference" />
        <xs:element ref="flag" />
      </xs:sequence>
    </xs:sequence>
  </xs:complexType>
</xs:element>

<xs:element name="sdal">
  <xs:complexType>
    <xs:attributeGroup ref="attrs.box"
                       use="required" />
    <xs:sequence>
      <xs:element ref="id" />
      <xs:element ref="data" />
    </xs:sequence>
  </xs:complexType>
</xs:element>

<xs:element name="sref">
  <xs:complexType>
    <xs:attributeGroup ref="attrs.box"
                       use="required" />
    <xs:sequence>
      <xs:element ref="id" />
    </xs:sequence>
  </xs:complexType>
</xs:element>

<xs:element name="page">
  <xs:complexType>
    <xs:attributeGroup ref="attrs.box"
                       use="required" />
    <xs:sequence>
      <xs:element ref="phdr" />
      <xs:element ref="ppcl"
                   minOccurs="0" />
      <xs:element ref="res "
                   minOccurs="0" />
      <xs:element ref="bclr"
                   minOccurs="0" />
      <xs:element ref="meta"
                   minOccurs="0" />
      <xs:element ref="lbl_"
                   minOccurs="0" />
      <xs:element ref="lobj"
                   maxOccurs="unbounded" />
    </xs:sequence>
  </xs:complexType>
</xs:element>

<xs:element name="phdr">
  <xs:complexType>
    <xs:attributeGroup ref="attrs.box"
                       use="required" />
    <xs:sequence>
      <xs:element ref="num lobjs" />
      <xs:element ref="height" />
      <xs:element ref="width" />
      <xs:element ref="orientation" />
      <xs:element ref="colour" />
    </xs:sequence>
  </xs:complexType>
</xs:element>

<xs:element name="lobj">
  <xs:complexType>
    <xs:attributeGroup ref="attrs.box"
                       use="required" />
    <xs:sequence>
      <xs:element ref="ihdr" />
      <xs:element ref="meta"
                   minOccurs="0" />
      <xs:element ref="lbl "
                   minOccurs="0" />
      <xs:element ref="objc"
                   maxOccurs="unbounded" />
    </xs:sequence>
  </xs:complexType>
</xs:element>

<xs:element name="lhdr">
  <xs:complexType>
    <xs:attributeGroup ref="attrs.box"
```

```
                        use="required" />
    <xs:sequence>
      <xs:element ref="id" />
      <xs:element ref="height" />
      <xs:element ref="width" />
      <xs:element ref="top" />
      <xs:element ref="left" />
      <xs:element ref="style" />
    </xs:sequence>
  </xs:complexType>
</xs:element>

<xs:element name="objc">
  <xs:complexType>
    <xs:attributeGroup ref="attrs.box"
                        use="required" />
    <xs:sequence>
      <xs:element ref="ohdr" />
      <xs:element ref="bclr"
                  minOccurs="0" />
      <xs:element ref="meta"
                  minOccurs="0" />
      <xs:element ref="lbl "
                  minOccurs="0" />
      <xs:element ref="scal"
                  minOccurs="0" />
      <xs:element ref="jp2h"
                  minOccurs="0" />
    </xs:sequence>
  </xs:complexType>
</xs:element>

<xs:element name="lhdr">
  <xs:complexType>
    <xs:attributeGroup ref="attrs.box"
                        use="required" />
    <xs:sequence>
      <xs:element ref="type" />
      <xs:element ref="no_codestream" />
      <xs:element ref="top" />
      <xs:element ref="left" />
      <xs:element ref="offset" />
      <xs:element ref="length" />
      <xs:element ref="data_reference" />
    </xs:sequence>
  </xs:complexType>
</xs:element>

<xs:element name="scal">
  <xs:complexType>
    <xs:attributeGroup ref="attrs.box"
                        use="required" />
    <xs:sequence>
      <xs:element ref="vert_num" />
      <xs:element ref="vert den" />
      <xs:element ref="hori num" />
      <xs:element ref="hori den" />
    </xs:sequence>
  </xs:complexType>
</xs:element>

<xs:element name="bclr">
  <xs:complexType>
    <xs:attributeGroup ref="attrs.box"
                        use="required" />
    <xs:sequence>
      <xs:element ref="bcvl" />
      <xs:element ref="colr"
                  minOccurs="0" />
      <xs:element ref="bpcc"
                  minOccurs="0" />
    </xs:sequence>
  </xs:complexType>
</xs:element>

<xs:element name="bcvl">
  <xs:complexType>
    <xs:attributeGroup ref="attrs.box"
                        use="required" />
    <xs:sequence>
      <xs:element ref="num_components" />
      <xs:element ref="depth" />
      <xs:element ref="data" />
    </xs:sequence>
  </xs:complexType>
</xs:element>
```

```
<xs:element name="htxb">
  <xs:complexType>
    <xs:attributeGroup ref="attrs.box"
                        use="required" />
    <xs:sequence>
      <xs:element ref="lbl "
                  minOccurs="0" />
      <xs:sequence maxOccurs="unbounded">
        <xs:choice>
          <xs:element ref="_xml_" />
          <xs:element ref="uuid" />
        </xs:choice>
      </xs:sequence>
    </xs:sequence>
  </xs:complexType>
</xs:element>

<xs:element name="phtx">
  <xs:complexType>
    <xs:attributeGroup ref="attrs.box"
                        use="required" />
    <xs:sequence>
      <xs:element ref="type" />
      <xs:element ref="flst" />
      <xs:element ref="lbl_"
                  minOccurs="0" />
    </xs:sequence>
  </xs:complexType>
</xs:element>

<!-- part 1 content element -->
<!-- part 2 content element -->
…
<!-- part 6 content element -->
<xs:element name="profile" type="xs:integer">
  <xs:attributeGroup ref="attrs.int"
                      use="required" />
</xs:element>

<xs:element name="self contained"
type="xs:integer">
  <xs:attributeGroup ref="attrs.int"
                      use="required" />
</xs:element>

<xs:element name="offset" type="xs:integer">
  <xs:attributeGroup ref="attrs.int"
                      use="required" />
</xs:element>

<xs:element name="mask coder" type="xs:integer">
  <xs:attributeGroup ref="attrs.int"
                      use="required" />
</xs:element>

<xs:element name="image coder" type="xs:integer">
  <xs:attributeGroup ref="attrs.int"
                      use="required" />
</xs:element>

<xs:element name="flag" type="hexbyte">
  <xs:attributeGroup ref="attrs.hex"
                      use="required" />
</xs:element>

<xs:element name="num_lobjs"
             type="xs:integer">
  <xs:attributeGroup ref="attrs.int"
                      use="required" />
</xs:element>

<xs:element name="orientation"
             type="xs:integer">
  <xs:attributeGroup ref="attrs.int"
                      use="required" />
</xs:element>

<xs:element name="top" type="xs:integer">
  <xs:attributeGroup ref="attrs.int"
                      use="required" />
</xs:element>

<xs:element name="left" type="xs:integer">
  <xs:attributeGroup ref="attrs.int"
                      use="required" />
</xs:element>
```

```
<xs:element name="style" type="xs:integer">
  <xs:attributeGroup ref="attrs.int"
                     use="required" />
</xs:element>
```

```
<xs:element name="no codestream"
type="xs:integer">
  <xs:attributeGroup ref="attrs.int"
                     use="required" />
</xs:element>
```

A.3.6 Example of an XML schema for a JPEG 2000 Part 8 image (security)

```
<!-- part 1 box element -->
<!-- part 8 box element -->
<xs:element name="gprt">
  <xs:complexType>
    <xs:attributeGroup ref="attrs.box"
                       use="required" />
    <xs:sequence>
      <xs:element ref="type_flag" />
      <xs:element ref="size flag" />
      <xs:element ref="location flag" />
      <xs:element ref="reserved" />
      <xs:choice>
        <xs:sequence>
          <xs:element ref="num entries" />
          <xs:sequence maxOccurs="unbounded">
            <xs:element ref="offset_size" />
            <xs:element ref="box length" />
            <xs:element ref="box_type" />
            <xs:element ref="box ext length" />
            <xs:element ref="offset" />
          </xs:sequence>
        </xs:sequence>
        <xs:sequence>
          <xs:element ref="total length" />
          <xs:element ref="total ext length" />
        </xs:sequence>
      </xs:choice>
    </xs:sequence>
  </xs:complexType>
</xs:element>

<!-- part 12 box element -->
<!-- followings need part 12 -->
<xs:element name="schi">
  <xs:complexType>
    <xs:attributeGroup ref="attrs.box"
                       use="required" />
    <xs:sequence>
      <xs:element ref="type" />
      <xs:choice>
        <!-- for Decryption -->
        <xs:sequence>
          <xs:element ref="box_protected" />
          <xs:choice>
            <xs:element ref="bcip" />
            <xs:element ref="scip" />
          </xs:choice>
          <xs:element ref="gran" />
        </xs:sequence>
        <!-- for Authentication -->
        <xs:sequence>
          <xs:element ref="auth" />
          <xs:element ref="gran"
                      minOccurs="0" />
        </xs:sequence>
      </xs:choice>
      <xs:element ref="vall" />
    </xs:sequence>
  </xs:complexType>
</xs:element>

<xs:element name="bcip">
  <xs:complexType>
    <xs:attributeGroup ref="attrs.box"
                       use="required" />
    <xs:sequence>
      <xs:element ref="cipher id" />
      <xs:element ref="cipher mode" />
      <xs:element ref="padding mode" />
      <xs:element ref="size" />
      <xs:element ref="keyt" />
    </xs:sequence>
  </xs:complexType>
</xs:element>

<xs:element name="scip">
  <xs:complexType>
    <xs:attributeGroup ref="attrs.box"
                       use="required" />
    <xs:sequence>
      <xs:element ref="type" />
      <xs:element ref="cipher id" />
      <xs:element ref="keyt" />
    </xs:sequence>
  </xs:complexType>
</xs:element>

<xs:element name="auth">
  <xs:complexType>
    <xs:attributeGroup ref="attrs.box"
                       use="required" />
    <xs:sequence>
      <xs:element ref="type" />
      <xs:element ref="method id" />
      <xs:element ref="hash_id" />
      <xs:element ref="size" />
      <xs:element ref="keyt" />
    </xs:sequence>
  </xs:complexType>
</xs:element>

<xs:element name="keyt">
  <xs:complexType>
    <xs:attributeGroup ref="attrs.box"
                       use="required" />
    <xs:sequence>
      <xs:element ref="size" />
      <xs:element ref="key info" />
      <xs:element ref="gran"
                  minOccurs="0" />
      <xs:element ref="vall" />
    </xs:sequence>
  </xs:complexType>
</xs:element>

<xs:element name="gran">
  <xs:complexType>
    <xs:attributeGroup ref="attrs.box"
                       use="required" />
    <xs:sequence>
      <xs:element ref="granularity" />
    </xs:sequence>
  </xs:complexType>
</xs:element>

<xs:element name="vall">
  <xs:complexType>
    <xs:attributeGroup ref="attrs.box"
                       use="required" />
    <xs:sequence>
      <xs:element ref="size" />
      <xs:element ref="num_entries" />
      <xs:element ref="count"
              maxOccurs="unbounded" />
      <xs:element ref="value"
              maxOccurs="unbounded" />
    </xs:sequence>
  </xs:complexType>
</xs:element>

<xs:element name="ides">
  <xs:complexType>
    <xs:attributeGroup ref="attrs.box"
                       use="required" />
    <xs:sequence>
      <xs:element ref="num entries" />
      <xs:sequence maxOccurs="unbounded">
        <xs:element ref="dest" />
        <xs:element ref="item_id" />
        <xs:element ref="desd" />
      </xs:sequence>
    </xs:sequence>
  </xs:complexType>
</xs:element>
```

```
<xs:element name="dest">
  <xs:complexType>
    <xs:attributeGroup ref="attrs.box"
                       use="required" />
    <xs:sequence>
      <xs:element ref="type" />
      <xs:element ref="version" />
      <xs:element ref="location" />
    </xs:sequence>
  </xs:complexType>
</xs:element>

<xs:element name="vide">
  <xs:complexType>
    <xs:attributeGroup ref="attrs.box"
                       use="required" />
    <xs:sequence>
      <xs:element ref="layer start" />
      <xs:element ref="layer count" />
      <xs:element ref="res_start" />
      <xs:element ref="res count" />
      <xs:element ref="hori_offset" />
      <xs:element ref="hori lenght" />
      <xs:element ref="vert offset" />
      <xs:element ref="vert_length" />
      <xs:element ref="colour" />
      <xs:element ref="time_start" />
      <xs:element ref="time length" />
    </xs:sequence>
  </xs:complexType>
</xs:element>

<xs:element name="j2ke">
  <xs:complexType>
    <xs:attributeGroup ref="attrs.box"
                       use="required" />
    <xs:sequence>
      <xs:element ref="vide"
                  minOccurs="0" />
      <xs:element ref="tile start" />
      <xs:element ref="tile count" />
      <xs:element ref="precinct_start" />
      <xs:element ref="precinct count" />
      <xs:element ref="j2k_packet_start" />
      <xs:element ref="j2k packet count" />
    </xs:sequence>
  </xs:complexType>
</xs:element>

<xs:element name="icor">
  <xs:complexType>
    <xs:attributeGroup ref="attrs.box"
                       use="required" />
    <xs:sequence>
      <xs:element ref="item id" />
      <xs:element ref="desc id" />
    </xs:sequence>
  </xs:complexType>
</xs:element>

<xs:element name="port">
  <xs:complexType>
    <xs:attributeGroup ref="attrs.box"
                       use="required" />
    <xs:sequence>
      <xs:element ref="resolution" />
      <xs:element ref="layer" />
      <xs:element ref="cropped width" />
      <xs:element ref="cropped_height" />
      <xs:sequence minOccurs="0" >
        <xs:element ref="startx" />
        <xs:element ref="srarty" />
      </xs:sequence>
    </xs:sequence>
  </xs:complexType>
</xs:element>

<xs:element name="attr">
  <xs:complexType>
    <xs:attributeGroup ref="attrs.box"
                       use="required" />
    <xs:sequence>
      <xs:element ref="resolution" />
      <xs:element ref="layer" />
      <xs:element ref="cropped width" />
      <xs:element ref="cropped_height" />
      <xs:sequence minOccurs="0" >
        <xs:element ref="startx" />
        <xs:element ref="srarty" />
      </xs:sequence>
    </xs:sequence>
  </xs:complexType>
</xs:element>

<xs:element name="sces">
  <xs:complexType>
    <xs:attributeGroup ref="attrs.box"
                       use="required" />
    <xs:sequence>
      <xs:element ref="reserved" />
      <xs:element ref="data reference" />
      <xs:element ref="predefined" />
      <xs:element ref="reserved" />
      <xs:element ref="predefined" />
      <xs:element ref="width" />
      <xs:element ref="height" />
      <xs:element ref="hori_resolution" />
      <xs:element ref="vert resolution" />
      <xs:element ref="reserved" />
      <xs:element ref="predefined" />
      <xs:element ref="name" />
      <xs:element ref="depth" />
      <xs:element ref="predefined" />
      <xs:element ref="clap" minOccurs="0" />
      <xs:element ref="pasp" minOccurs="0" />
      <xs:element ref="resolution" />
      <xs:element ref="layer" />
      <xs:element ref="cropped width" />
      <xs:element ref="cropped_height" />
      <xs:sequence minOccurs="0" >
        <xs:element ref="startx" />
        <xs:element ref="srarty" />
      </xs:sequence>
    </xs:sequence>
  </xs:complexType>
</xs:element>

<xs:element name="dces">
  <xs:complexType>
    <xs:attributeGroup ref="attrs.box"
                       use="required" />
    <xs:sequence>
      <xs:element ref="reserved" />
      <xs:element ref="data_reference" />
      <xs:element ref="predefined" />
      <xs:element ref="reserved" />
      <xs:element ref="predefined" />
      <xs:element ref="width" />
      <xs:element ref="height" />
      <xs:element ref="hori resolution" />
      <xs:element ref="vert_resolution" />
      <xs:element ref="reserved" />
      <xs:element ref="predefined" />
      <xs:element ref="name" />
      <xs:element ref="depth" />
      <xs:element ref="predefined" />
      <xs:element ref="clap" minOccurs="0" />
      <xs:element ref="pasp" minOccurs="0" />
      <xs:element ref="resolution" />
      <xs:element ref="layer" />
      <xs:element ref="cropped_width" />
      <xs:element ref="cropped height" />
      <xs:sequence minOccurs="0" >
        <xs:element ref="startx" />
        <xs:element ref="srarty" />
      </xs:sequence>
    </xs:sequence>
  </xs:complexType>
</xs:element>

<!-- part 1 content element -->
<!-- part 8 content element -->
<xs:element ref="type_flag" type="xs:integer">
  <xs:attributeGroup ref="attrs.int"
                     use="required" />
</xs:element>

<xs:element ref="size_flag" type="xs:integer">
  <xs:attributeGroup ref="attrs.int"
                     use="required" />
</xs:element>

<xs:element ref="location flag"
              type="xs:integer">
```

```
  <xs:attributeGroup ref="attrs.int"
                     use="required" />
</xs:element>

<xs:element ref="reserved" type="xs:integer">
  <xs:attributeGroup ref="attrs.int"
                     use="required" />
</xs:element>

<xs:element ref="offset_size"
type="xs:integer">
  <xs:attributeGroup ref="attrs.int"
                     use="required" />
</xs:element>

<xs:element ref="box length" type="xs:integer">
  <xs:attributeGroup ref="attrs.int"
                     use="required" />
</xs:element>

<xs:element ref="box type" type="xs:integer">
  <xs:attributeGroup ref="attrs.int"
                     use="required" />
</xs:element>

<xs:element ref="box ext length"
type="xs:integer">
  <xs:attributeGroup ref="attrs.int"
                     use="required" />
</xs:element>

<xs:element ref="offset" type="xs:integer">
  <xs:attributeGroup ref="attrs.int"
                     use="required" />
</xs:element>

<xs:element ref="total_length"
             type="xs:integer">
  <xs:attributeGroup ref="attrs.int"
                     use="required" />
</xs:element>

<xs:element ref="total ext length"
type="xs:integer">
  <xs:attributeGroup ref="attrs.int"
                     use="required" />
</xs:element>

<!-- part 12 box element -->
<!-- followings need part 12 -->
<xs:element ref="box protected"
              type="xs:integer">
  <xs:attributeGroup ref="attrs.int"
                     use="required" />
</xs:element>

<xs:element ref="cipher id" type="xs:integer">
  <xs:attributeGroup ref="attrs.int"
                     use="required" />
</xs:element>

<xs:element ref="method id" type="xs:integer">
  <xs:attributeGroup ref="attrs.int"
                     use="required" />
</xs:element>

<xs:element ref="hash id" type="xs:integer">
  <xs:attributeGroup ref="attrs.int"
                     use="required" />
</xs:element>

<xs:element ref="cipher mode"
              type="xs:integer">
  <xs:attributeGroup ref="attrs.int"
                     use="required" />
</xs:element>

<xs:element ref="padding mode"
              type="xs:integer">
  <xs:attributeGroup ref="attrs.int"
                     use="required" />
</xs:element>

<xs:element ref="size" type="xs:integer">
  <xs:attributeGroup ref="attrs.int"
                     use="required" />
</xs:element>
```

```
<xs:element ref="key info" type="xs:integer">
  <xs:attributeGroup ref="attrs.int"
                     use="required" />
</xs:element>

<xs:element ref="granularity"
             type="xs:integer">
  <xs:attributeGroup ref="attrs.int"
                     use="required" />
</xs:element>

<xs:element ref="num entries" type="xs:integer">
  <xs:attributeGroup ref="attrs.int"
                     use="required" />
</xs:element>

<xs:element ref="count" type="xs:integer">
  <xs:attributeGroup ref="attrs.int"
                     use="required" />
</xs:element>

<xs:element ref="value" type="xs:integer">
  <xs:attributeGroup ref="attrs.int"
                     use="required" />
</xs:element>

<xs:element ref="item id" type="xs:integer">
  <xs:attributeGroup ref="attrs.int"
                     use="required" />
</xs:element>

<xs:element ref="layer start"
             type="xs:integer">
  <xs:attributeGroup ref="attrs.int"
                     use="required" />
</xs:element>

<xs:element ref="layer_count"
             type="xs:integer">
  <xs:attributeGroup ref="attrs.int"
                     use="required" />
</xs:element>

<xs:element ref="res start" type="xs:integer">
  <xs:attributeGroup ref="attrs.int"
                     use="required" />
</xs:element>

<xs:element ref="res count" type="xs:integer">
  <xs:attributeGroup ref="attrs.int"
                     use="required" />
</xs:element>

<xs:element ref="hori offset"
             type="xs:integer">
  <xs:attributeGroup ref="attrs.int"
                     use="required" />
</xs:element>

<xs:element ref="hori lenght"
             type="xs:integer">
  <xs:attributeGroup ref="attrs.int"
                     use="required" />
</xs:element>

<xs:element ref="vert offset"
             type="xs:integer">
  <xs:attributeGroup ref="attrs.int"
                     use="required" />
</xs:element>

<xs:element ref="vert length"
              type="xs:integer">
  <xs:attributeGroup ref="attrs.int"
                     use="required" />
</xs:element>

<xs:element ref="time_start" type="xs:integer">
  <xs:attributeGroup ref="attrs.int"
                     use="required" />
</xs:element>

<xs:element ref="time_length"
             type="xs:integer">
  <xs:attributeGroup ref="attrs.int"
                     use="required" />
```

```
</xs:element>

<xs:element ref="tile start" type="xs:integer">
  <xs:attributeGroup ref="attrs.int"
                     use="required" />
</xs:element>

<xs:element ref="tile count" type="xs:integer">
  <xs:attributeGroup ref="attrs.int"
                     use="required" />
</xs:element>

<xs:element ref="precinct start"
             type="xs:integer">
  <xs:attributeGroup ref="attrs.int"
                     use="required" />
</xs:element>

<xs:element ref="precinct count"
             type="xs:integer">
  <xs:attributeGroup ref="attrs.int"
                     use="required" />
</xs:element>

<xs:element ref="j2k_packet_start"
             type="xs:integer">
  <xs:attributeGroup ref="attrs.int"
                     use="required" />
</xs:element>

<xs:element ref="j2k packet count"
             type="xs:integer">
  <xs:attributeGroup ref="attrs.int"
                     use="required" />
</xs:element>

<xs:element ref="item_id" type="xs:integer">
  <xs:attributeGroup ref="attrs.int"
                     use="required" />
</xs:element>

<xs:element ref="desc_id" type="xs:integer">
  <xs:attributeGroup ref="attrs.int"
                     use="required" />
</xs:element>

<xs:element ref="reserved" type="xs:integer">
  <xs:attributeGroup ref="attrs.int"
                     use="required" />
</xs:element>
```

```
<xs:element ref="predefined" type="xs:integer">
  <xs:attributeGroup ref="attrs.int"
                     use="required" />
</xs:element>

<xs:element ref="hori_resolution"
             type="xs:integer">
  <xs:attributeGroup ref="attrs.int"
                     use="required" />
</xs:element>

<xs:element ref="vert resolution"
             type="xs:integer">
  <xs:attributeGroup ref="attrs.int"
                     use="required" />
</xs:element>

<xs:element ref="resolution" type="xs:integer">
  <xs:attributeGroup ref="attrs.int"
                     use="required" />
</xs:element>

<xs:element ref="layer" type="xs:integer">
  <xs:attributeGroup ref="attrs.int"
                     use="required" />
</xs:element>

<xs:element ref="cropped width"
             type="xs:integer">
  <xs:attributeGroup ref="attrs.int"
                     use="required" />
</xs:element>

<xs:element ref="cropped height"
             type="xs:integer">
  <xs:attributeGroup ref="attrs.int"
                     use="required" />
</xs:element>

<xs:element ref="startx" type="xs:integer">
  <xs:attributeGroup ref="attrs.int"
                     use="required" />
</xs:element>

<xs:element ref="srarty" type="xs:integer">
  <xs:attributeGroup ref="attrs.int"
                     use="required" />
</xs:element>
```

A.3.7 Example of an XML schema for a JPEG 2000 Part 12 image (motion image)

```
<!-- part 12 box element -->
<xs:element name="mdat">
  <xs:complexType>
    <xs:attributeGroup ref="attrs.box"
                       use="required" />
    <xs:sequence>
      <xs:element ref="data" />
    </xs:sequence>
  </xs:complexType>
</xs:element>

<xs:element name="free">
  <xs:complexType>
    <xs:attributeGroup ref="attrs.box"
                       use="required" />
    <xs:sequence>
      <xs:element ref="data" />
    </xs:sequence>
  </xs:complexType>
</xs:element>

<xs:element name="skip">
  <xs:complexType>
    <xs:attributeGroup ref="attrs.box"
                       use="required" />
    <xs:sequence>
      <xs:element ref="data" />
    </xs:sequence>
  </xs:complexType>
</xs:element>

<xs:element name="pdin">
  <xs:complexType>
```

```
    <xs:attributeGroup ref="attrs.box"
                       use="required" />
    <xs:sequence>
      <xs:element ref="version" />
      <xs:element ref="flag" />
      <xs:sequence maxOccurs="unbounded">
        <xs:element ref="rate" />
        <xs:element ref="inital_delay" />
      </xs:sequence>
    </xs:sequence>
  </xs:complexType>
</xs:element>

<xs:element name="moov">
  <xs:complexType>
    <xs:attributeGroup ref="attrs.box"
                       use="required" />
    <xs:sequence>
      <xs:element ref="mvhd" />
      <xs:element ref="trak"
                  maxOccurs="unbounded" />
      <xs:element ref="mvex" minOccurs="0" />
      <xs:element ref="udta" minOccurs="0" />
      <xs:element ref="meta" minOccurs="0" />
      <xs:element ref="meco" minOccurs="0" />
      <xs:element ref="ipmc" minOccurs="0" />
    </xs:sequence>
  </xs:complexType>
</xs:element>

<xs:element name="mvhd">
  <xs:complexType>
    <xs:attributeGroup ref="attrs.box"
```

```
                        use="required" />
    <xs:sequence>
      <xs:element ref="version" />
      <xs:element ref="flag" />
      <xs:element ref="creation_time" />
      <xs:element ref="modification time" />
      <xs:element ref="time_scale" />
      <xs:element ref="duration" />
      <xs:element ref="rate" />
      <xs:element ref="volume" />
      <xs:element ref="reserved" />
      <xs:element ref="matrix" />
      <xs:element ref="predefined" />
      <xs:element ref="next track id" />
    </xs:sequence>
  </xs:complexType>
</xs:element>

<xs:element name="trak">
  <xs:complexType>
    <xs:attributeGroup ref="attrs.box"
                        use="required" />
    <xs:sequence>
      <xs:element ref="tkhd" />
      <xs:element ref="mdia" />
      <xs:element ref="tref" minOccurs="0" />
      <xs:element ref="edts" minOccurs="0" />
      <xs:element ref="udta" minOccurs="0" />
      <xs:element ref="meta" minOccurs="0" />
      <xs:element ref="meco" minOccurs="0" />
    </xs:sequence>
  </xs:complexType>
</xs:element>

<xs:element name="tkhd">
  <xs:complexType>
    <xs:attributeGroup ref="attrs.box"
                        use="required" />
    <xs:sequence>
      <xs:element ref="version" />
      <xs:element ref="flag" />
      <xs:element ref="creation_time" />
      <xs:element ref="modification time" />
      <xs:element ref="track_id" />
      <xs:element ref="reserved" />
      <xs:element ref="duration" />
      <xs:element ref="reserved" />
      <xs:element ref="layer" />
      <xs:element ref="alternate_group" />
      <xs:element ref="volume" />
      <xs:element ref="reserved" />
      <xs:element ref="matrix" />
      <xs:element ref="width" />
      <xs:element ref="height" />
    </xs:sequence>
  </xs:complexType>
</xs:element>

<xs:element name="tref">
  <xs:complexType>
    <xs:attributeGroup ref="attrs.box"
                        use="required" />
    <xs:sequence>
      <xs:choice>
        <xs:element ref="hint" />
        <xs:element ref="cdsc" />
        <xs:element ref="hind" />
      </xs:choice>
    </xs:sequence>
  </xs:complexType>
</xs:element>

<xs:element name="hint">
  <xs:complexType>
    <xs:attributeGroup ref="attrs.box"
                        use="required" />
    <xs:sequence>
      <xs:element ref="track id" />
    </xs:sequence>
  </xs:complexType>
</xs:element>

<xs:element name="cdsc">
  <xs:complexType>
    <xs:attributeGroup ref="attrs.box"
                        use="required" />
    <xs:sequence>
      <xs:element ref="track id" />
    </xs:sequence>
  </xs:complexType>
</xs:element>

<xs:element name="hind">
  <xs:complexType>
    <xs:attributeGroup ref="attrs.box"
                        use="required" />
    <xs:sequence>
      <xs:element ref="track id" />
    </xs:sequence>
  </xs:complexType>
</xs:element>

<xs:element name="mdia">
  <xs:complexType>
    <xs:attributeGroup ref="attrs.box"
                        use="required" />
    <xs:sequence>
      <xs:element ref="mdhd" />
      <xs:element ref="hdlr" />
      <xs:element ref="minf" />
    </xs:sequence>
  </xs:complexType>
</xs:element>

<xs:element name="mdhd">
  <xs:complexType>
    <xs:attributeGroup ref="attrs.box"
                        use="required" />
    <xs:sequence>
      <xs:element ref="version" />
      <xs:element ref="flag" />
      <xs:element ref="creation time" />
      <xs:element ref="modification time" />
      <xs:element ref="time_scale" />
      <xs:element ref="duration" />
      <xs:element ref="language" />
      <xs:element ref="predefined" />
    </xs:sequence>
  </xs:complexType>
</xs:element>

<xs:element name="hdlr">
  <xs:complexType>
    <xs:attributeGroup ref="attrs.box"
                        use="required" />
    <xs:sequence>
      <xs:element ref="version" />
      <xs:element ref="flag" />
      <xs:element ref="predefined" />
      <xs:element ref="hdlr type" />
      <xs:element ref="reserved" />
      <xs:element ref="name" />
    </xs:sequence>
  </xs:complexType>
</xs:element>

<xs:element name="minf">
  <xs:complexType>
    <xs:attributeGroup ref="attrs.box"
                        use="required" />
    <xs:sequence>
      <xs:choice>
        <xs:element ref="vmhd" />
        <xs:element ref="smhd" />
        <xs:element ref="hmhd" />
        <xs:element ref="nmhd" />
      </xs:choice>
      <xs:element ref="stbl" />
      <xs:element ref="dinf" />
    </xs:sequence>
  </xs:complexType>
</xs:element>

<xs:element name="vmhd">
  <xs:complexType>
    <xs:attributeGroup ref="attrs.box"
                        use="required" />
    <xs:sequence>
      <xs:element ref="version" />
      <xs:element ref="flag" />
      <xs:element ref="graphic_mode" />
      <xs:element ref="op code" />
    </xs:sequence>
  </xs:complexType>
```

```
</xs:element>

<xs:element name="smhd">
  <xs:complexType>
    <xs:attributeGroup ref="attrs.box"
                       use="required" />
    <xs:sequence>
      <xs:element ref="version" />
      <xs:element ref="flag" />
      <xs:element ref="balance" />
      <xs:element ref="reserved" />
    </xs:sequence>
  </xs:complexType>
</xs:element>

<xs:element name="hmhd">
  <xs:complexType>
    <xs:attributeGroup ref="attrs.box"
                       use="required" />
    <xs:sequence>
      <xs:element ref="version" />
      <xs:element ref="flag" />
      <xs:element ref="max pdu size" />
      <xs:element ref="avg pdu size" />
      <xs:element ref="max_bitrate" />
      <xs:element ref="avg bitrate" />
      <xs:element ref="reserved" />
    </xs:sequence>
  </xs:complexType>
</xs:element>

<xs:element name="stbl">
  <xs:complexType>
    <xs:attributeGroup ref="attrs.box"
                       use="required" />
    <xs:sequence>
      <xs:element ref="stsd" />
      <xs:element ref="stdp" minOccurs="0" />
      <xs:element ref="ctts" />
      <xs:element ref="stss" minOccurs="0" />
      <xs:element ref="stsh" minOccurs="0" />
      <xs:element ref="sdtp" minOccurs="0" />
      <xs:choice>
        <xs:element ref="stsz" />
        <xs:element ref="stz2" />
      </xs:choice>
      <xs:element ref="stsc" />
      <xs:choice>
        <xs:element ref="stco" />
        <xs:element ref="co64" />
      </xs:choice>
      <xs:element ref="padb" minOccurs="0" />
      <xs:element ref="subs" minOccurs="0" />
      <xs:element ref="sbgp" minOccurs="0"
                  maxOccurs="unbounded" />
      <xs:element ref="sgpd" minOccurs="0"
                  maxOccurs="unbounded" />
    </xs:sequence>
  </xs:complexType>
</xs:element>

<xs:element name="btrt">
  <xs:complexType>
    <xs:attributeGroup ref="attrs.box"
                       use="required" />
    <xs:sequence>
      <xs:element ref="buffer size" />
      <xs:element ref="max bitrate" />
      <xs:element ref="avg_bitrate" />
    </xs:sequence>
  </xs:complexType>
</xs:element>

<xs:element name="metx">
  <xs:complexType>
    <xs:attributeGroup ref="attrs.box"
                       use="required" />
    <xs:sequence>
      <xs:element ref="reserved" />
      <xs:element ref="data reference" />
      <xs:element ref="content_encoding" />
      <xs:element ref="namespace" />
      <xs:element ref="location" />
      <xs:element ref="btrt"
                  minOccurs="0"
                  maxOccurs="unbounded" />
    </xs:sequence>
  </xs:complexType>
</xs:element>

<xs:element name="mett">
  <xs:complexType>
    <xs:attributeGroup ref="attrs.box"
                       use="required" />
    <xs:sequence>
      <xs:element ref="reserved" />
      <xs:element ref="data_reference" />
      <xs:element ref="content encoding" />
      <xs:element ref="mine_format" />
      <xs:element ref="btrt"
                  minOccurs="0"
                  maxOccurs="unbounded" />
    </xs:sequence>
  </xs:complexType>
</xs:element>

<xs:element name="pasp">
  <xs:complexType>
    <xs:attributeGroup ref="attrs.box"
                       use="required" />
    <xs:sequence>
      <xs:element ref="hori_spacing" />
      <xs:element ref="vert spacing" />
    </xs:sequence>
  </xs:complexType>
</xs:element>

<xs:element name="clap">
  <xs:complexType>
    <xs:attributeGroup ref="attrs.box"
                       use="required" />
    <xs:sequence>
      <xs:element ref="width num" />
      <xs:element ref="width_den" />
      <xs:element ref="height num" />
      <xs:element ref="height_den" />
      <xs:element ref="hori offset num" />
      <xs:element ref="hori offset den" />
      <xs:element ref="vert_offset_num" />
      <xs:element ref="vert offset den" />
    </xs:sequence>
  </xs:complexType>
</xs:element>

<xs:element name="stsd">
  <xs:complexType>
    <xs:attributeGroup ref="attrs.box"
                       use="required" />
    <xs:sequence>
      <xs:element ref="version" />
      <xs:element ref="flag" />
      <xs:element ref="num entries" />
      <xs:sequence maxOccurs="unbounded">
        <xs:element ref="mjp2" />
        <xs:element ref="raw " />
        <xs:element ref="twos" />
      </xs:sequence>
    </xs:sequence>
  </xs:complexType>
</xs:element>

<xs:element name=VideoCoding4CC>
  <xs:complexType>
    <xs:attributeGroup ref="attrs.box"
                       use="required" />
    <xs:sequence>
      <xs:element ref="reserved" />
      <xs:element ref="data_reference" />
      <xs:element ref="predefined" />
      <xs:element ref="reserved" />
      <xs:element ref="predefined" />
      <xs:element ref="width" />
      <xs:element ref="height" />
      <xs:element ref="hori resolution" />
      <xs:element ref="vert resolution" />
      <xs:element ref="reserved" />
      <xs:element ref="predefined" />
      <xs:element ref="name" />
      <xs:element ref="depth" />
      <xs:element ref="predefined" />
      <xs:element ref="clap" minOccurs="0" />
      <xs:element ref="pasp" minOccurs="0" />
    </xs:sequence>
  </xs:complexType>
```

```
</xs:element>

<xs:element name="AudioCoding4CC">
  <xs:complexType>
    <xs:attributeGroup ref="attrs.box"
                       use="required" />
    <xs:sequence>
      <xs:element ref="reserved" />
      <xs:element ref="data reference" />
      <xs:element ref="reserved" />
      <xs:element ref="channel count" />
      <xs:element ref="sample_size" />
      <xs:element ref="predefined" />
      <xs:element ref="reserved" />
      <xs:element ref="sample_rate" />
    </xs:sequence>
  </xs:complexType>
</xs:element>

<xs:element name="stdp">
  <xs:complexType>
    <xs:attributeGroup ref="attrs.box"
                       use="required" />
    <xs:sequence>
      <xs:element ref="version" />
      <xs:element ref="flag" />
      <xs:element ref="priority" />
    </xs:sequence>
  </xs:complexType>
</xs:element>

<xs:element name="stts">
  <xs:complexType>
    <xs:attributeGroup ref="attrs.box"
                       use="required" />
    <xs:sequence>
      <xs:element ref="version" />
      <xs:element ref="flag" />
      <xs:element ref="num_entries" />
      <xs:sequence maxOccurs="unbounded">
        <xs:element ref="sample count" />
        <xs:element ref="sample_delta" />
      </xs:sequence>
    </xs:sequence>
  </xs:complexType>
</xs:element>

<xs:element name="ctts">
  <xs:complexType>
    <xs:attributeGroup ref="attrs.box"
                       use="required" />
    <xs:sequence>
      <xs:element ref="version" />
      <xs:element ref="flag" />
      <xs:element ref="num entries" />
      <xs:sequence maxOccurs="unbounded">
        <xs:element ref="sample count" />
        <xs:element ref="sample offset" />
      </xs:sequence>
    </xs:sequence>
  </xs:complexType>
</xs:element>

<xs:element name="stss">
  <xs:complexType>
    <xs:attributeGroup ref="attrs.box"
                       use="required" />
    <xs:sequence>
      <xs:element ref="version" />
      <xs:element ref="flag" />
      <xs:element ref="num_entries" />
      <xs:sequence maxOccurs="unbounded">
        <xs:element ref="sample number" />
      </xs:sequence>
    </xs:sequence>
  </xs:complexType>
</xs:element>

<xs:element name="stsh">
  <xs:complexType>
    <xs:attributeGroup ref="attrs.box"
                       use="required" />
    <xs:sequence>
      <xs:element ref="version" />
      <xs:element ref="flag" />
      <xs:element ref="num_entries" />
      <xs:sequence maxOccurs="unbounded">
        <xs:element
                  ref="shadowed sample number"
/>
        <xs:element ref="sync sample number" />
      </xs:sequence>
    </xs:sequence>
  </xs:complexType>
</xs:element>

<xs:element name="sdtp">
  <xs:complexType>
    <xs:attributeGroup ref="attrs.box"
                       use="required" />
    <xs:sequence>
      <xs:element ref="version" />
      <xs:element ref="flag" />
      <xs:sequence maxOccurs="unbounded">
        <xs:element ref="reserved" />
        <xs:element ref="sample depends on" />
        <xs:element ref="sample_is_depends_on"
/>
        <xs:element ref="sample_has_redundancy"
/>
      </xs:sequence>
    </xs:sequence>
  </xs:complexType>
</xs:element>

<xs:element name="edts">
  <xs:complexType>
    <xs:attributeGroup ref="attrs.box"
                       use="required" />
    <xs:sequence>
      <xs:element ref="elst"
                  minOccurs="0" />
    </xs:sequence>
  </xs:complexType>
</xs:element>

<xs:element name="elst">
  <xs:complexType>
    <xs:attributeGroup ref="attrs.box"
                       use="required" />
    <xs:sequence>
      <xs:element ref="version" />
      <xs:element ref="flag" />
      <xs:element ref="num_entries" />
      <xs:sequence maxOccurs="unbounded">
        <xs:element ref="duration" />
        <xs:element ref="media time" />
        <xs:element ref="media rate integer" />
        <xs:element ref="media_rate_fraction" />
      </xs:sequence>
    </xs:sequence>
  </xs:complexType>
</xs:element>

<xs:element name="dinf">
  <xs:complexType>
    <xs:attributeGroup ref="attrs.box"
                       use="required" />
    <xs:sequence>
      <xs:choice>
        <xs:element ref="dref" />
        <xs:element ref="url " />
        <xs:element ref="urn_" />
      </xs:choice>
    </xs:sequence>
  </xs:complexType>
</xs:element>

<xs:element name="url ">
  <xs:complexType>
    <xs:attributeGroup ref="attrs.box"
                       use="required" />
    <xs:sequence>
      <xs:element ref="version" />
      <xs:element ref="flag" />
      <xs:element ref="location" />
    </xs:sequence>
  </xs:complexType>
</xs:element>

<xs:element name="urn_">
  <xs:complexType>
    <xs:attributeGroup ref="attrs.box"
                       use="required" />
```

```
    <xs:sequence>
      <xs:element ref="version" />
      <xs:element ref="flag" />
      <xs:element ref="name" />
      <xs:element ref="location" />
    </xs:sequence>
  </xs:complexType>
</xs:element>

<xs:element name="dref">
  <xs:complexType>
    <xs:attributeGroup ref="attrs.box"
                       use="required" />
    <xs:sequence>
      <xs:element ref="version" />
      <xs:element ref="flag" />
      <xs:element ref="num_entries" />
      <xs:sequence maxOccurs="unbounded">
        <xs:choice>
          <xs:element ref="url_" />
          <xs:element ref="urn " />
        </xs:choice>
      </xs:sequence>
    </xs:sequence>
  </xs:complexType>
</xs:element>

<xs:element name="stsz">
  <xs:complexType>
    <xs:attributeGroup ref="attrs.box"
                       use="required" />
    <xs:sequence>
      <xs:element ref="version" />
      <xs:element ref="flag" />
      <xs:element ref="sample size" />
      <xs:element ref="num entries" />
      <xs:element ref="entry_size"
                  minOccurs="0"
                  maxOccurs="unbounded" />
    </xs:sequence>
  </xs:complexType>
</xs:element>

<xs:element name="stz2">
  <xs:complexType>
    <xs:attributeGroup ref="attrs.box"
                       use="required" />
    <xs:sequence>
      <xs:element ref="version" />
      <xs:element ref="flag" />
      <xs:element ref="reserved" />
      <xs:element ref="field_size" />
      <xs:element ref="num entries" />
      <xs:element ref="entry_size"
                  minOccurs="0"
                  maxOccurs="unbounded" />
    </xs:sequence>
  </xs:complexType>
</xs:element>

<xs:element name="stsc">
  <xs:complexType>
    <xs:attributeGroup ref="attrs.box"
                       use="required" />
    <xs:sequence>
      <xs:element ref="version" />
      <xs:element ref="flag" />
      <xs:element ref="num entries" />
      <xs:sequence maxOccurs="unbounded">
        <xs:element ref="first chunk" />
        <xs:element ref="sample_per_chunk" />
        <xs:element ref="description index" />
      </xs:sequence>
    </xs:sequence>
  </xs:complexType>
</xs:element>

<xs:element name="stco">
  <xs:complexType>
    <xs:attributeGroup ref="attrs.box"
                       use="required" />
    <xs:sequence>
      <xs:element ref="version" />
      <xs:element ref="flag" />
      <xs:element ref="num entries" />
      <xs:element ref="chunk_offset"
                  maxOccurs="unbounded" />
    </xs:sequence>
  </xs:complexType>
</xs:element>

<xs:element name="co64">
  <xs:complexType>
    <xs:attributeGroup ref="attrs.box"
                       use="required" />
    <xs:sequence>
      <xs:element ref="version" />
      <xs:element ref="flag" />
      <xs:element ref="num_entries" />
      <xs:element ref="chunk offset"
                  maxOccurs="unbounded" />
    </xs:sequence>
  </xs:complexType>
</xs:element>

<xs:element name="padb">
  <xs:complexType>
    <xs:attributeGroup ref="attrs.box"
                       use="required" />
    <xs:sequence>
      <xs:element ref="version" />
      <xs:element ref="flag" />
      <xs:element ref="pad1" />
      <xs:element ref="pad2" />
    </xs:sequence>
  </xs:complexType>
</xs:element>

<xs:element name="subs">
  <xs:complexType>
    <xs:attributeGroup ref="attrs.box"
                       use="required" />
    <xs:sequence>
      <xs:element ref="version" />
      <xs:element ref="flag" />
      <xs:element ref="num_entries" />
      <xs:sequence maxOccurs="unbounded">
        <xs:element ref="sample delta" />
        <xs:element ref="num_subsample" />
        <xs:sequence maxOccurs="unbounded">
          <xs:element ref="subsample_size" />
          <xs:element ref="subsample priority"
/>
          <xs:element ref="discardable" />
          <xs:element ref="reserved" />
        </xs:sequence>
      </xs:sequence>
    </xs:sequence>
  </xs:complexType>
</xs:element>

<xs:element name="mvex">
  <xs:complexType>
    <xs:attributeGroup ref="attrs.box"
                       use="required" />
    <xs:sequence>
      <xs:element ref="mehd"
                  minOccurs="0" />
      <xs:element ref="trex" />
    </xs:sequence>
  </xs:complexType>
</xs:element>

<xs:element name="mehd">
  <xs:complexType>
    <xs:attributeGroup ref="attrs.box"
                       use="required" />
    <xs:sequence>
      <xs:element ref="version" />
      <xs:element ref="flag" />
      <xs:element ref="duration" />
    </xs:sequence>
  </xs:complexType>
</xs:element>

<xs:element name="trex">
  <xs:complexType>
    <xs:attributeGroup ref="attrs.box"
                       use="required" />
    <xs:sequence>
      <xs:element ref="version" />
      <xs:element ref="flag" />
      <xs:element ref="track id" />
      <xs:element ref="sample_index" />
```

```
      <xs:element ref="sample duration" />
      <xs:element ref="sample size" />
      <xs:element ref="sample flags" />
    </xs:sequence>
  </xs:complexType>
</xs:element>

<xs:element name="moof">
  <xs:complexType>
    <xs:attributeGroup ref="attrs.box"
                       use="required" />
    <xs:sequence>
      <xs:element ref="mfhd" />
      <xs:element ref="traf" />
    </xs:sequence>
  </xs:complexType>
</xs:element>

<xs:element name="mfhd">
  <xs:complexType>
    <xs:attributeGroup ref="attrs.box"
                       use="required" />
    <xs:sequence>
      <xs:element ref="version" />
      <xs:element ref="flag" />
      <xs:element ref="sequence number" />
    </xs:sequence>
  </xs:complexType>
</xs:element>

<xs:element name="traf">
  <xs:complexType>
    <xs:attributeGroup ref="attrs.box"
                       use="required" />
    <xs:sequence>
      <xs:element ref="tfhd" />
      <xs:element ref="traf" minOccurs="0" />
      <xs:element ref="trun" minOccurs="0" />
    </xs:sequence>
  </xs:complexType>
</xs:element>

<xs:element name="tfhd">
  <xs:complexType>
    <xs:attributeGroup ref="attrs.box"
                       use="required" />
    <xs:sequence>
      <xs:element ref="version" />
      <xs:element ref="flag" />
      <xs:element ref="track id" />
      <xs:element ref="base delta offset" />
      <xs:element ref="sample_index" />
      <xs:element ref="sample duration" />
      <xs:element ref="sample_size" />
      <xs:element ref="sample flags" />
    </xs:sequence>
  </xs:complexType>
</xs:element>

<xs:element name="trun">
  <xs:complexType>
    <xs:attributeGroup ref="attrs.box"
                       use="required" />
    <xs:sequence>
      <xs:element ref="version" />
      <xs:element ref="flag" />
      <xs:element ref="track id" />
      <xs:element ref="num entries" />
      <xs:element ref="delta_offset" />
      <xs:element ref="first sample flags" />
      <xs:sequence maxOccurs="unbounded">
        <xs:element ref="sample duration" />
        <xs:element ref="sample size" />
        <xs:element ref="sample_flags" />
        <xs:element ref="sample time offset"
/>
      </xs:sequence>
    </xs:sequence>
  </xs:complexType>
</xs:element>

<xs:element name="mfra">
  <xs:complexType>
    <xs:attributeGroup ref="attrs.box"
                       use="required" />
    <xs:sequence>
      <xs:element ref="tfra" />
      <xs:element ref="mfro" />
    </xs:sequence>
  </xs:complexType>
</xs:element>

<xs:element name="tfra">
  <xs:complexType>
    <xs:attributeGroup ref="attrs.box"
                       use="required" />
    <xs:sequence>
      <xs:element ref="version" />
      <xs:element ref="flag" />
      <xs:element ref="track id" />
      <xs:element ref="reserved" />
      <xs:element ref="traf_index_size" />
      <xs:element ref="trun index size" />
      <xs:element ref="sample_index_size" />
      <xs:element ref="num entries" />
      <xs:sequence maxOccurs="unbounded">
        <xs:element ref="time" />
        <xs:element ref="moof offset" />
        <xs:element ref="traf_index" />
        <xs:element ref="trun index" />
        <xs:element ref="sample index" />
      </xs:sequence>
    </xs:sequence>
  </xs:complexType>
</xs:element>

<xs:element name="sbgp">
  <xs:complexType>
    <xs:attributeGroup ref="attrs.box"
                       use="required" />
    <xs:sequence>
      <xs:element ref="version" />
      <xs:element ref="flag" />
      <xs:element ref="type" />
      <xs:element ref="num entries" />
      <xs:sequence maxOccurs="unbounded">
        <xs:element ref="sample count" />
        <xs:element
               ref="group_description_index" />
      </xs:sequence>
    </xs:sequence>
  </xs:complexType>
</xs:element>

<xs:element name="sbgd">
  <xs:complexType>
    <xs:attributeGroup ref="attrs.box"
                       use="required" />
    <xs:sequence>
      <xs:element ref="version" />
      <xs:element ref="flag" />
      <xs:element ref="type" />
      <xs:element ref="default length" />
      <xs:element ref="num entries" />
      <xs:sequence maxOccurs="unbounded">
        <xs:element ref="description_length" />
        <xs:element ref="data" />
      </xs:sequence>
    </xs:sequence>
  </xs:complexType>
</xs:element>

<xs:element name="udta">
  <xs:complexType>
    <xs:attributeGroup ref="attrs.box"
                       use="required" />
    <xs:sequence>
      <xs:element ref="cprt"
                  minOccurs="0"
                  maxOccurs="unbounded" />
    </xs:sequence>
  </xs:complexType>
</xs:element>

<xs:element name="cprt">
  <xs:complexType>
    <xs:attributeGroup ref="attrs.box"
                       use="required" />
    <xs:sequence>
      <xs:element ref="version" />
      <xs:element ref="flag" />
      <xs:element ref="language" />
      <xs:element ref="notice" />
    </xs:sequence>
```

```
  </xs:complexType>
</xs:element>

<xs:element name="tsel">
  <xs:complexType>
    <xs:attributeGroup ref="attrs.box"
                       use="required" />
    <xs:sequence>
      <xs:element ref="version" />
      <xs:element ref="flag" />
      <xs:element ref="switch group" />
      <xs:element ref="attributes" />
    </xs:sequence>
  </xs:complexType>
</xs:element>

<xs:element name="meta">
  <xs:complexType>
    <xs:attributeGroup ref="attrs.box"
                       use="required" />
    <xs:sequence>
      <xs:element ref="version" />
      <xs:element ref="flag" />
      <xs:choice minOccurs="0">
        <xs:element ref="_xml_" />
        <xs:element ref="bxml" />
      </xs:choice>
      <xs:sequence>
        <xs:element ref="iloc" minOccurs="0"
/>
      <xs:element ref="pitm" minOccurs="0" />
      <xs:element ref="ipro" minOccurs="0" />
      <xs:element ref="iinf" minOccurs="0" />
      <xs:element ref="ipmc" minOccurs="0" />
      <xs:element ref="fiim" minOccurs="0" />
      <xs:element ref="hdlr" minOccurs="0" />
      <xs:element ref="dinf" minOccurs="0" />
      </xs:sequence>
    </xs:sequence>
  </xs:complexType>
</xs:element>

<xs:element name=" xml ">
  <xs:complexType>
    <xs:attributeGroup ref="attrs.box"
                       use="required" />
    <xs:sequence>
      <xs:element ref="version" />
      <xs:element ref="flag" />
      <xs:element ref="text" />
    </xs:sequence>
  </xs:complexType>
</xs:element>

<xs:element name="bxml">
  <xs:complexType>
    <xs:attributeGroup ref="attrs.box"
                       use="required" />
    <xs:sequence>
      <xs:element ref="version" />
      <xs:element ref="flag" />
      <xs:element ref="data" />
    </xs:sequence>
  </xs:complexType>
</xs:element>

<xs:element name="iloc">
  <xs:complexType>
    <xs:attributeGroup ref="attrs.box"
                       use="required" />
    <xs:sequence>
      <xs:element ref="version" />
      <xs:element ref="flag" />
      <xs:choice>
        <xs:sequence>
          <xs:element ref="offset_size" />
          <xs:element ref="length size" />
          <xs:element ref="base offset size"
/>
          <xs:element ref="reserved" />
          <xs:element ref="num_entries" />
          <xs:sequence maxOccurs="unbounded">
            <xs:element ref="item id" />
            <xs:element ref="data_reference"
/>
            <xs:element ref="base_offset" />
            <xs:element ref="sample_index" />
            <xs:element ref="num extents" />
            <xs:sequence maxOccurs="unbounded">
              <xs:element ref="extent offset" />
              <xs:element ref="extent length" />
            </xs:sequence>
          </xs:sequence>
        </xs:sequence>
        <xs:element ref="item id" />
      </xs:choice>
    </xs:sequence>
  </xs:complexType>
</xs:element>

<xs:element name="ipro">
  <xs:complexType>
    <xs:attributeGroup ref="attrs.box"
                       use="required" />
    <xs:sequence>
      <xs:element ref="version" />
      <xs:element ref="flag" />
      <xs:element ref="num entries" />
      <xs:sequence maxOccurs="unbounded">
        <xs:element ref="sinf" />
      </xs:sequence>
    </xs:sequence>
  </xs:complexType>
</xs:element>

<xs:element name="infe">
  <xs:complexType>
    <xs:attributeGroup ref="attrs.box"
                       use="required" />
    <xs:sequence>
      <xs:element ref="version" />
      <xs:element ref="flag" />
      <xs:element ref="item id" />
      <xs:element ref="item_protection_index" />
      <xs:element ref="item name" />
      <xs:element ref="content_type" />
      <xs:element ref="content encoding" />
      <xs:element ref="type" />
      <xs:element ref="data" />
    </xs:sequence>
  </xs:complexType>
</xs:element>

<xs:element name="iinf">
  <xs:complexType>
    <xs:attributeGroup ref="attrs.box"
                       use="required" />
    <xs:sequence>
      <xs:element ref="version" />
      <xs:element ref="flag" />
      <xs:element ref="num_entries" />
      <xs:element ref="infe"
                   maxOccurs="unbounded" />
    </xs:sequence>
  </xs:complexType>
</xs:element>

<xs:element name="meco">
  <xs:complexType>
    <xs:attributeGroup ref="attrs.box"
                       use="required" />
    <xs:sequence>
      <xs:element ref="mere"
                   minOccurs="0"
                   maxOccurs="unbounded" />
      <xs:element ref="meta"
                   maxOccurs="unbounded" />
    </xs:sequence>
  </xs:complexType>
</xs:element>

<xs:element name="sinf">
  <xs:complexType>
    <xs:attributeGroup ref="attrs.box"
                       use="required" />
    <xs:sequence>
      <xs:element ref="frma" />
      <xs:element ref="imif" minOccurs="0" />
      <xs:element ref="schm" minOccurs="0" />
      <xs:element ref="schi" minOccurs="0" />
    </xs:sequence>
  </xs:complexType>
</xs:element>
```

```
<xs:element name="frma">
  <xs:complexType>
    <xs:attributeGroup ref="attrs.box"
                       use="required" />
    <xs:sequence>
      <xs:element ref="data format" />
    </xs:sequence>
  </xs:complexType>
</xs:element>

<xs:element name="imif">
  <xs:complexType>
    <xs:attributeGroup ref="attrs.box"
                       use="required" />
    <xs:sequence>
      <xs:element ref="version" />
      <xs:element ref="flag" />
      <xs:element ref="metabox relation" />
    </xs:sequence>
  </xs:complexType>
</xs:element>

<xs:element name="ipmc">
  <xs:complexType>
    <xs:attributeGroup ref="attrs.box"
                       use="required" />
    <xs:sequence>
      <xs:element ref="version" />
      <xs:element ref="flag" />
      <xs:element ref="tool list" />
      <xs:element ref="num entries" />
      <xs:element ref="metabox_relaion" />
    </xs:sequence>
  </xs:complexType>
</xs:element>

<xs:element name="schm">
  <xs:complexType>
    <xs:attributeGroup ref="attrs.box"
                       use="required" />
    <xs:sequence>
      <xs:element ref="version" />
      <xs:element ref="flag" />
      <xs:element ref="schema_type" />
      <xs:element ref="schema version" />
      <xs:element ref="schema location" />
    </xs:sequence>
  </xs:complexType>
</xs:element>

<xs:element name="schi">
  <xs:complexType>
    <xs:attributeGroup ref="attrs.box"
                       use="required" />
    <xs:sequence>
      <xs:element ref="fpar" />
      <xs:element ref="fecr" minOccurs="0" />
    </xs:sequence>
  </xs:complexType>
</xs:element>

<xs:element name="fiin">
  <xs:complexType>
    <xs:attributeGroup ref="attrs.box"
                       use="required" />
    <xs:sequence>
      <xs:element ref="version" />
      <xs:element ref="flag" />
      <xs:element ref="num_entries" />
      <xs:element ref="paen"
                  maxOccurs="unbounded" />
      <xs:element ref="segr" minOccurs="0" />
      <xs:element ref="gitn" minOccurs="0" />
    </xs:sequence>
  </xs:complexType>
</xs:element>

<xs:element name="fpar">
  <xs:complexType>
    <xs:attributeGroup ref="attrs.box"
                       use="required" />
    <xs:sequence>
      <xs:element ref="version" />
      <xs:element ref="flag" />
      <xs:element ref="packet payload size" />
      <xs:element ref="reserved" />
      <xs:element ref="FEC_encoding_id" />
      <xs:element ref="FEC instance id" />
      <xs:element ref="max source block length"
/>
      <xs:element ref="symbol length" />
      <xs:element ref="max_num_symbols" />
      <xs:element ref="scheme specific info" />
      <xs:element ref="num_entries" />
      <xs:sequence maxOccurs="unbounded">
        <xs:element ref="block count" />
        <xs:element ref="block_size" />
      </xs:sequence>
    </xs:sequence>
  </xs:complexType>
</xs:element>

<xs:element name="segr">
  <xs:complexType>
    <xs:attributeGroup ref="attrs.box"
                       use="required" />
    <xs:sequence>
      <xs:element ref="version" />
      <xs:element ref="flag" />
      <xs:element ref="num entries" />
      <xs:sequence maxOccurs="unbounded">
        <xs:element ref="num_group_ids" />
        <xs:sequence maxOccurs="unbounded">
          <xs:element ref="group_id" />
        </xs:sequence>
        <xs:element ref="num hint track ids" />
        <xs:sequence maxOccurs="unbounded">
          <xs:element ref="hint track id" />
        </xs:sequence>
      </xs:sequence>
    </xs:sequence>
  </xs:complexType>
</xs:element>

<xs:element name="gitn">
  <xs:complexType>
    <xs:attributeGroup ref="attrs.box"
                       use="required" />
    <xs:sequence>
      <xs:element ref="version" />
      <xs:element ref="flag" />
      <xs:element ref="num entries" />
      <xs:sequence maxOccurs="unbounded">
        <xs:element ref="group_id" />
        <xs:element ref="group name" />
      </xs:sequence>
    </xs:sequence>
  </xs:complexType>
</xs:element>

<xs:element name="rtp_">
  <xs:complexType>
    <xs:attributeGroup ref="attrs.box"
                       use="required" />
    <xs:sequence>
      <xs:element ref="reserved" />
      <xs:element ref="data reference" />
      <xs:element ref="track version" />
      <xs:element ref="compatible version" />
      <xs:element ref="max packet size" />
      <xs:sequence maxOccurs="unbounded">
        <xs:element ref="tims" />
        <xs:element ref="tsro"
                    minOccurs="0" />
        <xs:element ref="snro"
                    minOccurs="0" />
      </xs:sequence>
    </xs:sequence>
  </xs:complexType>
</xs:element>

<xs:element name="srtp">
  <xs:complexType>
    <xs:attributeGroup ref="attrs.box"
                       use="required" />
    <xs:sequence>
      <xs:element ref="reserved" />
      <xs:element ref="data_reference" />
      <xs:element ref="track version" />
      <xs:element ref="compatible version" />
      <xs:element ref="max_packet_size" />
      <xs:sequence maxOccurs="unbounded">
        <xs:element ref="tims" />
        <xs:element ref="srpp" />
```

```
        <xs:element ref="tsro" minOccurs="0"
/>
        <xs:element ref="snro" minOccurs="0"
/>
      </xs:sequence>
    </xs:sequence>
  </xs:complexType>
</xs:element>

<xs:element name="tims">
  <xs:complexType>
    <xs:attributeGroup ref="attrs.box"
                       use="required" />
    <xs:sequence>
      <xs:element ref="time_scale" />
    </xs:sequence>
  </xs:complexType>
</xs:element>

<xs:element name="tsro">
  <xs:complexType>
    <xs:attributeGroup ref="attrs.box"
                       use="required" />
    <xs:sequence>
      <xs:element ref="offset" />
    </xs:sequence>
  </xs:complexType>
</xs:element>

<xs:element name="snro">
  <xs:complexType>
    <xs:attributeGroup ref="attrs.box"
                       use="required" />
    <xs:sequence>
      <xs:element ref="offset" />
    </xs:sequence>
  </xs:complexType>
</xs:element>

<xs:element name="srpp">
  <xs:complexType>
    <xs:attributeGroup ref="attrs.box"
                       use="required" />
    <xs:sequence>
      <xs:element ref="version" />
      <xs:element ref="flag" />
      <xs:element ref="encryption_rtp" />
      <xs:element ref="encryption rtcp" />
      <xs:element ref="integrity_rtp" />
      <xs:element ref="integrity rtcp" />
      <xs:element ref="schm" />
      <xs:element ref="schi"
                  minOccurs="0" />
    </xs:sequence>
  </xs:complexType>
</xs:element>

<xs:element name="rtpo">
  <xs:complexType>
    <xs:attributeGroup ref="attrs.box"
                       use="required" />
    <xs:sequence>
      <xs:element ref="offset" />
    </xs:sequence>
  </xs:complexType>
</xs:element>

<xs:element name="hinti">
  <xs:complexType>
    <xs:attributeGroup ref="attrs.box"
                       use="required" />
    <xs:sequence>
      <xs:choice>
        <xs:element ref="rtp_" />
        <xs:element ref="sdp " />
      </xs:choice>
    </xs:sequence>
  </xs:complexType>
</xs:element>

<xs:element name="rtp_">
  <xs:complexType>
    <xs:attributeGroup ref="attrs.box"
                       use="required" />
    <xs:sequence>
      <xs:element ref="format" />
      <xs:element ref="text" />
    </xs:sequence>
  </xs:complexType>
</xs:element>

<xs:element name="sdp_">
  <xs:complexType>
    <xs:attributeGroup ref="attrs.box"
                       use="required" />
    <xs:sequence>
      <xs:element ref="text" />
    </xs:sequence>
  </xs:complexType>
</xs:element>

<xs:element name="hinf">
  <xs:complexType>
    <xs:attributeGroup ref="attrs.box"
                       use="required" />
    <xs:sequence>
      <xs:element ref="trpy" minOccurs="0" />
      <xs:element ref="nump" minOccurs="0" />
      <xs:element ref="tpyl" minOccurs="0" />
      <xs:element ref="totl" minOccurs="0" />
      <xs:element ref="npck" minOccurs="0" />
      <xs:element ref="tpay" minOccurs="0" />
      <xs:element ref="maxr" minOccurs="0"
                   maxOccurs="unbounded" />
      <xs:element ref="dmed" minOccurs="0" />
      <xs:element ref="dimm" minOccurs="0" />
      <xs:element ref="drep" minOccurs="0" />
      <xs:element ref="tmin" minOccurs="0" />
      <xs:element ref="tmax"  minOccurs="0" />
      <xs:element ref="pmax" minOccurs="0" />
      <xs:element ref="dmax" minOccurs="0" />
      <xs:element ref="payt" minOccurs="0" />
    </xs:sequence>
  </xs:complexType>
</xs:element>

<xs:element name="trpy">
  <xs:complexType>
    <xs:attributeGroup ref="attrs.box"
                       use="required" />
    <xs:sequence>
      <xs:element ref="bytes sent" />
    </xs:sequence>
  </xs:complexType>
</xs:element>

<xs:element name="tryl">
  <xs:complexType>
    <xs:attributeGroup ref="attrs.box"
                       use="required" />
    <xs:sequence>
      <xs:element ref="bytes sent" />
    </xs:sequence>
  </xs:complexType>
</xs:element>

<xs:element name="dmed">
  <xs:complexType>
    <xs:attributeGroup ref="attrs.box"
                       use="required" />
    <xs:sequence>
      <xs:element ref="bytes sent" />
    </xs:sequence>
  </xs:complexType>
</xs:element>

<xs:element name="dima">
  <xs:complexType>
    <xs:attributeGroup ref="attrs.box"
                       use="required" />
    <xs:sequence>
      <xs:element ref="bytes sent" />
    </xs:sequence>
  </xs:complexType>
</xs:element>

<xs:element name="drep">
  <xs:complexType>
    <xs:attributeGroup ref="attrs.box"
                       use="required" />
    <xs:sequence>
      <xs:element ref="bytes sent" />
    </xs:sequence>
  </xs:complexType>
```

```
</xs:element>

<xs:element name="totl">
  <xs:complexType>
    <xs:attributeGroup ref="attrs.box"
                       use="required" />
    <xs:sequence>
      <xs:element ref="bytes sent" />
    </xs:sequence>
  </xs:complexType>
</xs:element>

<xs:element name="tpay">
  <xs:complexType>
    <xs:attributeGroup ref="attrs.box"
                       use="required" />
    <xs:sequence>
      <xs:element ref="bytes sent" />
    </xs:sequence>
  </xs:complexType>
</xs:element>

<xs:element name="nump">
  <xs:complexType>
    <xs:attributeGroup ref="attrs.box"
                       use="required" />
    <xs:sequence>
      <xs:element ref="packets sent" />
    </xs:sequence>
  </xs:complexType>
</xs:element>

<xs:element name="npack">
  <xs:complexType>
    <xs:attributeGroup ref="attrs.box"
                       use="required" />
    <xs:sequence>
      <xs:element ref="packets sent" />
    </xs:sequence>
  </xs:complexType>
</xs:element>

<xs:element name="tmin">
  <xs:complexType>
    <xs:attributeGroup ref="attrs.box"
                       use="required" />
    <xs:sequence>
      <xs:element ref="time" />
    </xs:sequence>
  </xs:complexType>
</xs:element>

<xs:element name="tmax">
  <xs:complexType>
    <xs:attributeGroup ref="attrs.box"
                       use="required" />
    <xs:sequence>
      <xs:element ref="time" />
    </xs:sequence>
  </xs:complexType>
</xs:element>

<xs:element name="dmax">
  <xs:complexType>
    <xs:attributeGroup ref="attrs.box"
                       use="required" />
    <xs:sequence>
      <xs:element ref="time" />
    </xs:sequence>
  </xs:complexType>
</xs:element>

<xs:element name="pmax">
  <xs:complexType>
    <xs:attributeGroup ref="attrs.box"
                       use="required" />
    <xs:sequence>
      <xs:element ref="bytes sent" />
    </xs:sequence>
  </xs:complexType>
</xs:element>

<xs:element name="maxr">
  <xs:complexType>
    <xs:attributeGroup ref="attrs.box"
                       use="required" />
    <xs:sequence>
      <xs:element ref="period" />
      <xs:element ref="bytes sent" />
    </xs:sequence>
  </xs:complexType>
</xs:element>

<xs:element name="hnti">
  <xs:complexType>
    <xs:attributeGroup ref="attrs.box"
                       use="required" />
    <xs:sequence>
      <xs:element ref="payload_id" />
      <xs:element ref="count" />
      <xs:element ref="rtpmap" />
    </xs:sequence>
  </xs:complexType>
</xs:element>

<xs:element name="fdp ">
  <xs:complexType>
    <xs:attributeGroup ref="attrs.box"
                       use="required" />
    <xs:sequence>
      <xs:element ref="reserved" />
      <xs:element ref="data_reference" />
      <xs:element ref="track version" />
      <xs:element ref="compatible_version" />
      <xs:element ref="partition entry id" />
      <xs:element ref="FEC overhead" />
    </xs:sequence>
  </xs:complexType>
</xs:element>

<xs:element name="fdsa">
  <xs:complexType>
    <xs:attributeGroup ref="attrs.box"
                       use="required" />
    <xs:sequence>
      <xs:element ref="fdpa"
                   maxOccurs="unbounded" />
      <xs:element ref="extr"
                   minOccurs="0" />
    </xs:sequence>
  </xs:complexType>
</xs:element>

<xs:element name="fdpa">
  <xs:complexType>
    <xs:attributeGroup ref="attrs.box"
                       use="required" />
    <xs:sequence>
      <xs:element ref="header" />
      <xs:element ref="extension size" />
      <xs:element ref="extension" />
      <xs:element ref="packet size" />
      <xs:element ref="packet" />
    </xs:sequence>
  </xs:complexType>
</xs:element>

<xs:element name="extr">
  <xs:complexType>
    <xs:attributeGroup ref="attrs.box"
                       use="required" />
    <xs:sequence>
      <xs:element ref="data" />
    </xs:sequence>
  </xs:complexType>
</xs:element>

<!-- part 12 content element -->

<xs:element name="matrix">
  <xs:complexType>
    <xs:attributeGroup ref="attrs.hexbyte"
                       use="required" />
    <xs:sequence minOccurs="9" maxOccurs="9">
      <xs:element ref="cell" />
    </xs:sequence>
  </xs:complexType>
</xs:element>

<xs:element name="cell" type="xs:integer">
  <xs:attributeGroup ref="attrs.int"
                     use="required" />
</xs:element>
```

```
<xs:element name="version" type="xs:integer">
  <xs:attributeGroup ref="attrs.int"
                     use="required" />
</xs:element>

<xs:element name="flag" type="xs:integer">
  <xs:attributeGroup ref="attrs.int"
                     use="required" />
</xs:element>

<xs:element name="rate" type="xs:integer">
  <xs:attributeGroup ref="attrs.int"
                     use="required" />
</xs:element>

<xs:element name="data" type="hexbyte">
  <xs:attributeGroup ref="attrs.hex"
                     use="required" />
</xs:element>

<xs:element name="data reference"
             type="xs:integer">
  <xs:attributeGroup ref="attrs.int"
                     use="required" />
</xs:element>

<xs:element name="data_format"
             type="xs:integer">
  <xs:attributeGroup ref="attrs.int"
                     use="required" />
</xs:element>
             <xs:element name="inital_delay"
type="xs:integer">
  <xs:attributeGroup ref="attrs.int"
                     use="required" />
</xs:element>

<xs:element name="creation time"
             type="xs:integer">
  <xs:attributeGroup ref="attrs.int"
                     use="required" />
</xs:element>

<xs:element name="modification_time"
             type="xs:integer">
  <xs:attributeGroup ref="attrs.int"
                     use="required" />
</xs:element>

<xs:element name="time scale"
             type="xs:integer">
  <xs:attributeGroup ref="attrs.int"
                     use="required" />
</xs:element>

<xs:element name="duration"
             type="xs:integer">
  <xs:attributeGroup ref="attrs.int"
                     use="required" />
</xs:element>

<xs:element name="volume" type="xs:integer">
  <xs:attributeGroup ref="attrs.int"
                     use="required" />
</xs:element>

<xs:element name="reserved" type="xs:integer">
  <xs:attributeGroup ref="attrs.int"
                     use="required" />
</xs:element>

<xs:element name="predefined"
type="xs:integer">
  <xs:attributeGroup ref="attrs.int"
                     use="required" />
</xs:element>

<xs:element name="next track id"
             type="xs:integer">
  <xs:attributeGroup ref="attrs.int"
                     use="required" />
</xs:element>

<xs:element name="track_id"
             type="xs:integer">
  <xs:attributeGroup ref="attrs.int"
                     use="required" />
</xs:element>

<xs:element name="layer" type="xs:integer">
  <xs:attributeGroup ref="attrs.int"
                     use="required" />
</xs:element>

<xs:element name="alternate group"
             type="xs:integer">
  <xs:attributeGroup ref="attrs.int"
                     use="required" />
</xs:element>

<xs:element name="width" type="xs:integer">
  <xs:attributeGroup ref="attrs.int"
                     use="required" />
</xs:element>

<xs:element name="height" type="xs:integer">
  <xs:attributeGroup ref="attrs.int"
                     use="required" />
</xs:element>

<xs:element name="language"
             type="xs:integer">
  <xs:attributeGroup ref="attrs.int"
                     use="required" />
</xs:element>

<xs:element name="hdlr type"
type="xs:integer">
  <xs:attributeGroup ref="attrs.int"
                     use="required" />
</xs:element>

<xs:element name="name" type="xs:string">
  <xs:attributeGroup ref="attrs.str"
                     use="required" />
</xs:element>

<xs:element name="balance" type="xs:integer">
  <xs:attributeGroup ref="attrs.int"
                     use="required" />
</xs:element>

<xs:element name="max pdu size"
             type="xs:integer">
  <xs:attributeGroup ref="attrs.int"
                     use="required" />
</xs:element>

<xs:element name="avg_pdu_size"
             type="xs:integer">
  <xs:attributeGroup ref="attrs.int"
                     use="required" />
</xs:element>

<xs:element name="max bitrate"
             type="xs:integer">
  <xs:attributeGroup ref="attrs.int"
                     use="required" />
</xs:element>

<xs:element name="avg_bitrate"
             type="xs:integer">
  <xs:attributeGroup ref="attrs.int"
                     use="required" />
</xs:element>

<xs:element name="buffer size"
             type="xs:integer">
  <xs:attributeGroup ref="attrs.int"
                     use="required" />
</xs:element>

<xs:element name="content_encoding"
             type="xs:integer">
  <xs:attributeGroup ref="attrs.int"
                     use="required" />
</xs:element>

<xs:element name="namespace" type="xs:string">
  <xs:attributeGroup ref="attrs.str"
                     use="required" />
</xs:element>

<xs:element name="location" type="xs:string">
```

```
  <xs:attributeGroup ref="attrs.str"
                     use="required" />
</xs:element>

<xs:element name="mine_format"
            type="xs:string">
  <xs:attributeGroup ref="attrs.str"
                     use="required" />
</xs:element>

<xs:element name="hori spacing"
            type="xs:integer">
  <xs:attributeGroup ref="attrs.int"
                     use="required" />
</xs:element>

<xs:element name="vert_spacing"
            type="xs:integer">
  <xs:attributeGroup ref="attrs.int"
                     use="required" />
</xs:element>

<xs:element name="width num"
            type="xs:integer">
  <xs:attributeGroup ref="attrs.int"
                     use="required" />
</xs:element>

<xs:element name="width den"
            type="xs:integer">
  <xs:attributeGroup ref="attrs.int"
                     use="required" />
</xs:element>

<xs:element name="height num"
            type="xs:integer">
  <xs:attributeGroup ref="attrs.int"
                     use="required" />
</xs:element>

<xs:element name="height den"
            type="xs:integer">
  <xs:attributeGroup ref="attrs.int"
                     use="required" />
</xs:element>

<xs:element name="hori_offset_num"
            type="xs:integer">
  <xs:attributeGroup ref="attrs.int"
                     use="required" />
</xs:element>

<xs:element name="hori offset den"
            type="xs:integer">
  <xs:attributeGroup ref="attrs.int"
                     use="required" />
</xs:element>

<xs:element name="vert_offset_num"
            type="xs:integer">
  <xs:attributeGroup ref="attrs.int"
                     use="required" />
</xs:element>

<xs:element name="vert offset den"
            type="xs:integer">
  <xs:attributeGroup ref="attrs.int"
                     use="required" />
</xs:element>

<xs:element name="hori_resolution"
            type="xs:integer">
  <xs:attributeGroup ref="attrs.int"
                     use="required" />
</xs:element>

<xs:element name="vert resolution"
            type="xs:integer">
  <xs:attributeGroup ref="attrs.int"
                     use="required" />
</xs:element>

<xs:element name="depth" type="xs:integer">
  <xs:attributeGroup ref="attrs.int"
                     use="required" />
</xs:element>
```

```
<xs:element name="channel count"
            type="xs:integer">
  <xs:attributeGroup ref="attrs.int"
                     use="required" />
</xs:element>

<xs:element name="sample_size"
            type="xs:integer">
  <xs:attributeGroup ref="attrs.int"
                     use="required" />
</xs:element>

<xs:element name="sample rate"
            type="xs:integer">
  <xs:attributeGroup ref="attrs.int"
                     use="required" />
</xs:element>

<xs:element name="priority" type="xs:integer">
  <xs:attributeGroup ref="attrs.int"
                     use="required" />
</xs:element>

<xs:element name="sample delta"
            type="xs:integer">
  <xs:attributeGroup ref="attrs.int"
                     use="required" />
</xs:element>

<xs:element name="num entries"
            type="xs:integer">
  <xs:attributeGroup ref="attrs.int"
                     use="required" />
</xs:element>

<xs:element name="sample count"
            type="xs:integer">
  <xs:attributeGroup ref="attrs.int"
                     use="required" />
</xs:element>

<xs:element name="sample_offset"
            type="xs:integer">
  <xs:attributeGroup ref="attrs.int"
                     use="required" />
</xs:element>

<xs:element name="sample number"
            type="xs:integer">
  <xs:attributeGroup ref="attrs.int"
                     use="required" />
</xs:element>

<xs:element name="shadowed_sample_number"
            type="xs:integer">
  <xs:attributeGroup ref="attrs.int"
                     use="required" />
</xs:element>

<xs:element name="sync sample number"
            type="xs:integer">
  <xs:attributeGroup ref="attrs.int"
                     use="required" />
</xs:element>

<xs:element name="sample_depends_on"
            type="xs:integer">
  <xs:attributeGroup ref="attrs.int"
                     use="required" />
</xs:element>

<xs:element name="sample is depends on"
            type="xs:integer">
  <xs:attributeGroup ref="attrs.int"
                     use="required" />
</xs:element>

<xs:element name="sample has redundancy"
            type="xs:integer">
  <xs:attributeGroup ref="attrs.int"
                     use="required" />
</xs:element>

<xs:element name="media_time"
            type="xs:integer">
  <xs:attributeGroup ref="attrs.int"
                     use="required" />
```

```
</xs:element>

<xs:element name="media rate integer"
            type="xs:integer">
  <xs:attributeGroup ref="attrs.int"
                     use="required" />
</xs:element>

<xs:element name="media rate fraction"
            type="xs:integer">
  <xs:attributeGroup ref="attrs.int"
                     use="required" />
</xs:element>

<xs:element name="field_size"
            type="xs:integer">
  <xs:attributeGroup ref="attrs.int"
                     use="required" />
</xs:element>

<xs:element name="first chunk"
            type="xs:integer">
  <xs:attributeGroup ref="attrs.int"
                     use="required" />
</xs:element>

<xs:element name="sample_per_chunk"
            type="xs:integer">
  <xs:attributeGroup ref="attrs.int"
                     use="required" />
</xs:element>

<xs:element name="description index"
            type="xs:integer">
  <xs:attributeGroup ref="attrs.int"
                     use="required" />
</xs:element>

<xs:element name="chunk_offset"
            type="xs:integer">
  <xs:attributeGroup ref="attrs.int"
                     use="required" />
</xs:element>

<xs:element name="pad1" type="xs:integer">
  <xs:attributeGroup ref="attrs.int"
                     use="required" />
</xs:element>

<xs:element name="pad2" type="xs:integer">
  <xs:attributeGroup ref="attrs.int"
                     use="required" />
</xs:element>

<xs:element name="num subsample"
            type="xs:integer">
  <xs:attributeGroup ref="attrs.int"
                     use="required" />
</xs:element>

<xs:element name="subsample size"
            type="xs:integer">
  <xs:attributeGroup ref="attrs.int"
                     use="required" />
</xs:element>

<xs:element name="subsample priority"
            type="xs:integer">
  <xs:attributeGroup ref="attrs.int"
                     use="required" />
</xs:element>

<xs:element name="discardable"
            type="xs:integer">
  <xs:attributeGroup ref="attrs.int"
                     use="required" />
</xs:element>

<xs:element name="sample_index"
            type="xs:integer">
  <xs:attributeGroup ref="attrs.int"
                     use="required" />
</xs:element>

<xs:element name="sample duration"
            type="xs:integer">
  <xs:attributeGroup ref="attrs.int"
                     use="required" />
</xs:element>

<xs:element name="sample flags"
            type="xs:integer">
  <xs:attributeGroup ref="attrs.int"
                     use="required" />
</xs:element>

<xs:element name="sequence_number"
            type="xs:integer">
  <xs:attributeGroup ref="attrs.int"
                     use="required" />
</xs:element>

<xs:element name="base delta offset"
            type="xs:integer">
  <xs:attributeGroup ref="attrs.int"
                     use="required" />
</xs:element>

<xs:element name="delta_offset"
            type="xs:integer">
  <xs:attributeGroup ref="attrs.int"
                     use="required" />
</xs:element>

<xs:element name="first sample flags"
            type="xs:integer">
  <xs:attributeGroup ref="attrs.int"
                     use="required" />
</xs:element>

<xs:element name="sample time offset"
            type="xs:integer">
  <xs:attributeGroup ref="attrs.int"
                     use="required" />
</xs:element>

<xs:element name="traf index size"
            type="xs:integer">
  <xs:attributeGroup ref="attrs.int"
                     use="required" />
</xs:element>

<xs:element name="trun index size"
            type="xs:integer">
  <xs:attributeGroup ref="attrs.int"
                     use="required" />
</xs:element>

<xs:element name="sample_index_size"
            type="xs:integer">
  <xs:attributeGroup ref="attrs.int"
                     use="required" />
</xs:element>

<xs:element name="moof offset"
            type="xs:integer">
  <xs:attributeGroup ref="attrs.int"
                     use="required" />
</xs:element>

<xs:element name="traf_index"
            type="xs:integer">
  <xs:attributeGroup ref="attrs.int"
                     use="required" />
</xs:element>

<xs:element name="trun index"
            type="xs:integer">
  <xs:attributeGroup ref="attrs.int"
                     use="required" />
</xs:element>

<xs:element name="type" type="xs:integer">
  <xs:attributeGroup ref="attrs.int"
                     use="required" />
</xs:element>

<xs:element name="group_description_index"
            type="xs:integer">
  <xs:attributeGroup ref="attrs.int"
                     use="required" />
</xs:element>

<xs:element name="type" type="xs:integer">
```

```
  <xs:attributeGroup ref="attrs.int"
                     use="required" />
</xs:element>

<xs:element name="default_length"
            type="xs:integer">
  <xs:attributeGroup ref="attrs.int"
                     use="required" />
</xs:element>

<xs:element name="description length"
            type="xs:integer">
  <xs:attributeGroup ref="attrs.int"
                     use="required" />
</xs:element>

<xs:element name="language" type="xs:integer">
  <xs:attributeGroup ref="attrs.int"
                     use="required" />
</xs:element>

<xs:element name="notice" type="xs:integer">
  <xs:attributeGroup ref="attrs.int"
                     use="required" />
</xs:element>

<xs:element name="switch_group"
            type="xs:integer">
  <xs:attributeGroup ref="attrs.int"
                     use="required" />
</xs:element>

<xs:element name="attributes"
            type="xs:integer">
  <xs:attributeGroup ref="attrs.int"
                     use="required" />
</xs:element>

<xs:element name="offset_size"
            type="xs:integer">
  <xs:attributeGroup ref="attrs.int"
                     use="required" />
</xs:element>

<xs:element name="length size"
            type="xs:integer">
  <xs:attributeGroup ref="attrs.int"
                     use="required" />
</xs:element>

<xs:element name="base offset size"
            type="xs:integer">
  <xs:attributeGroup ref="attrs.int"
                     use="required" />
</xs:element>

<xs:element name="item id" type="xs:integer">
  <xs:attributeGroup ref="attrs.int"
                     use="required" />
</xs:element>

<xs:element name="base offset"
            type="xs:integer">
  <xs:attributeGroup ref="attrs.int"
                     use="required" />
</xs:element>

<xs:element name="sample index"
            type="xs:integer">
  <xs:attributeGroup ref="attrs.int"
                     use="required" />
</xs:element>

<xs:element name="num_extents"
            type="xs:integer">
  <xs:attributeGroup ref="attrs.int"
                     use="required" />
</xs:element>

<xs:element name="extent offset"
            type="xs:integer">
  <xs:attributeGroup ref="attrs.int"
                     use="required" />
</xs:element>

<xs:element name="extent_length"
            type="xs:integer">
```

```
  <xs:attributeGroup ref="attrs.int"
                     use="required" />
</xs:element>

<xs:element name="item_protection_index"
            type="xs:integer">
  <xs:attributeGroup ref="attrs.int"
                     use="required" />
</xs:element>

<xs:element name="item name"
            type="xs:integer">
  <xs:attributeGroup ref="attrs.int"
                     use="required" />
</xs:element>

<xs:element name="content_type"
            type="xs:integer">
  <xs:attributeGroup ref="attrs.int"
                     use="required" />
</xs:element>

<xs:element name="content encoding"
            type="xs:integer">
  <xs:attributeGroup ref="attrs.int"
                     use="required" />
</xs:element>

<xs:element name="metabox relation"
            type="xs:integer">
  <xs:attributeGroup ref="attrs.int"
                     use="required" />
</xs:element>

<xs:element name="tool list" type="hexbyte">
  <xs:attributeGroup ref="attrs.hex"
                     use="required" />
</xs:element>

<xs:element name="metabox relation"
            type="hexbyte">
  <xs:attributeGroup ref="attrs.hex"
                     use="required" />
</xs:element>

<xs:element name="schema type"
            type="xs:integer">
  <xs:attributeGroup ref="attrs.int"
                     use="required" />
</xs:element>

<xs:element name="schema_version"
            type="xs:integer">
  <xs:attributeGroup ref="attrs.int"
                     use="required" />
</xs:element>

<xs:element name="schema location"
            type="xs:string">
  <xs:attributeGroup ref="attrs.str"
                     use="required" />
</xs:element>

<xs:element name="packet_payload_size"
            type="xs:integer">
  <xs:attributeGroup ref="attrs.int"
                     use="required" />
</xs:element>

<xs:element name="FEC encoding id"
            type="xs:integer">
  <xs:attributeGroup ref="attrs.int"
                     use="required" />
</xs:element>

<xs:element name="FEC_instance_id"
            type="xs:integer">
  <xs:attributeGroup ref="attrs.int"
                     use="required" />
</xs:element>

<xs:element name="max source block length"
            type="xs:integer">
  <xs:attributeGroup ref="attrs.int"
                     use="required" />
</xs:element>
```

```
<xs:element name="symbol length"
            type="xs:integer">
  <xs:attributeGroup ref="attrs.int"
                     use="required" />
</xs:element>

<xs:element name="max_num_symbols"
            type="xs:integer">
  <xs:attributeGroup ref="attrs.int"
                     use="required" />
</xs:element>

<xs:element name="scheme specific info"
            type="xs:integer">
  <xs:attributeGroup ref="attrs.int"
                     use="required" />
</xs:element>

<xs:element name="block count"
            type="xs:integer">
  <xs:attributeGroup ref="attrs.int"
                     use="required" />
</xs:element>

<xs:element name="block_size"
            type="xs:integer">
  <xs:attributeGroup ref="attrs.int"
                     use="required" />
</xs:element>

<xs:element name="num group ids"
            type="xs:integer">
  <xs:attributeGroup ref="attrs.int"
                     use="required" />
</xs:element>

<xs:element name="num_hint_track_ids"
            type="xs:integer">
  <xs:attributeGroup ref="attrs.int"
                     use="required" />
</xs:element>

<xs:element name="hint track id"
            type="xs:integer">
  <xs:attributeGroup ref="attrs.int"
                     use="required" />
</xs:element>

<xs:element name="group_id"
            type="xs:integer">
  <xs:attributeGroup ref="attrs.int"
                     use="required" />
</xs:element>

<xs:element name="group name"
            type="xs:string">
  <xs:attributeGroup ref="attrs.str"
                     use="required" />
</xs:element>

<xs:element name="track version"
            type="xs:integer">
  <xs:attributeGroup ref="attrs.int"
                     use="required" />
</xs:element>

<xs:element name="compatible version"
            type="xs:integer">
  <xs:attributeGroup ref="attrs.int"
                     use="required" />
</xs:element>

<xs:element name="max packet size"
            type="xs:integer">
  <xs:attributeGroup ref="attrs.int"
                     use="required" />
</xs:element>

<xs:element name="max_packet_size"
            type="xs:integer">
  <xs:attributeGroup ref="attrs.int"
                     use="required" />
</xs:element>

<xs:element name="offset" type="xs:integer">
  <xs:attributeGroup ref="attrs.int"
                     use="required" />
</xs:element>

<xs:element name="encryption rtp"
            type="xs:integer">
  <xs:attributeGroup ref="attrs.int"
                     use="required" />
</xs:element>

<xs:element name="encryption rtcp"
            type="xs:integer">
  <xs:attributeGroup ref="attrs.int"
                     use="required" />
</xs:element>

<xs:element name="integrity_rtp"
            type="xs:integer">
  <xs:attributeGroup ref="attrs.int"
                     use="required" />
</xs:element>

<xs:element name="integrity rtcp"
            type="xs:integer">
  <xs:attributeGroup ref="attrs.int"
                     use="required" />
</xs:element>

<xs:element name="offset" type="xs:integer">
  <xs:attributeGroup ref="attrs.int"
                     use="required" />
</xs:element>

<xs:element name="format" type="xs:integer">
  <xs:attributeGroup ref="attrs.int"
                     use="required" />
</xs:element>

<xs:element name="text" type="xs:string">
  <xs:attributeGroup ref="attrs.str"
                     use="required" />
</xs:element>

<xs:element name="bytes_sent"
            type="xs:integer">
  <xs:attributeGroup ref="attrs.int"
                     use="required" />
</xs:element>

<xs:element name="packets sent"
            type="xs:integer">
  <xs:attributeGroup ref="attrs.int"
                     use="required" />
</xs:element>

<xs:element name="time" type="xs:integer">
  <xs:attributeGroup ref="attrs.int"
                     use="required" />
</xs:element>

<xs:element name="period" type="xs:integer">
  <xs:attributeGroup ref="attrs.int"
                     use="required" />
</xs:element>

<xs:element name="bytes_sent"
            type="xs:integer">
  <xs:attributeGroup ref="attrs.int"
                     use="required" />
</xs:element>

<xs:element name="payload id"
            type="xs:integer">
  <xs:attributeGroup ref="attrs.int"
                     use="required" />
</xs:element>

<xs:element name="count" type="xs:integer">
  <xs:attributeGroup ref="attrs.int"
                     use="required" />
</xs:element>

<xs:element name="rtpmap" type="xs:integer">
  <xs:attributeGroup ref="attrs.int"
                     use="required" />
</xs:element>

<xs:element name="partition entry id"
            type="xs:integer">
```

```
   <xs:attributeGroup ref="attrs.int"
                      use="required" />
</xs:element>

<xs:element name="FEC_overhead"
             type="xs:integer">
  <xs:attributeGroup ref="attrs.int"
                      use="required" />
</xs:element>

<xs:element name="header"
             type="xs:integer">
  <xs:attributeGroup ref="attrs.int"
                      use="required" />
</xs:element>

<xs:element name="extension_size"
             type="xs:integer">
  <xs:attributeGroup ref="attrs.int"
                      use="required" />
```

```
</xs:element>

<xs:element name="extension"
             type="hexbyte">
  <xs:attributeGroup ref="attrs.hex"
                      use="required" />
</xs:element>

<xs:element name="packet size"
             type="xs:integer">
  <xs:attributeGroup ref="attrs.int"
                      use="required" />
</xs:element>

<xs:element name="packet" type="hexbyte">
  <xs:attributeGroup ref="attrs.hex"
                      use="required" />
</xs:element>
```

Annex B

JPXML elements for codestream marker segments

(This annex forms an integral part of this Recommendation | International Standard.)

B.1 Introduction

This annex provides the predefined marker element name, content element names, and length and type attributes for the JPEG 2000 family codestream. The marker element appearance and structure shall be in conformity with Rec. ITU-T T.800 | ISO/IEC 15444-1, Rec. ITU-T T.801 | ISO/IEC 15444-2, Rec. ITU-T T.807 | ISO/IEC 15444-8, Rec. ITU-T T.809 | ISO/IEC 15444-10 and Rec. ITU-T T.810 | ISO/IEC 15444-11.

B.2 JPEG 2000 codestream marker element definitions

This subclause includes many tables of the marker's content element definitions, and these tables define the content element name, the content data length in byte, and the content data type. The JPXML data types are described in 7.5. Some tables define the sub-content element names of a defined content element, and some marker or content elements have content element definitions of two or more types. For more detailed information of the meaning and structure of this marker segment content, see each Recommendation | International Standard cited in clause B.1.

B.2.1 JPEG 2000 Part 1 marker elements

The JPEG 2000 part 1 marker elements shall be used as the element names described in Table B.1. These marker content element names are defined in Tables B.2 to B.19.

Table B.1 – JPEG 2000 Part 1 markers

Marker name	Element name	Code	Size	Main header	Tile-part header
Delimiting markers and marker segments					
Start of codestream	SOC	0xFF4F	2	required[a]	not allowed
Start of tile-part	SOT	0xFF90	12	not allowed	required
Start of data	SOD	0xFF93	2	not allowed	last marker
End of codestream	EOC	0xFFD9	2	not allowed	not allowed
Fixed information marker segments					
Image and tile size	SIZ	0xFF51	43 to 49,192	required	not allowed
Functional marker segments					
Coding style default	COD	0xFF52	14 to 47	required	optional
Coding style component	COC	0xFF53	11 to 45	optional	optional
Region-of-interest	RGN	0xFF5E	7 to 8	optional	optional
Quantization default	QCD	0xFF5C	6 to 199	required	optional
Quantization component	QCC	0xFF5D	7 to 201	optional	optional
Progression order change	POC	0xFF5F	11 to 65,537	optional	optional
Pointer marker segments					
Tile-part lengths	TLM	0xFF55	8 to 65,537	optional	not allowed
Packet length, main header	PLM	0xFF57	6 to 65,537	optional	not allowed
Packet length, tile-part header	PLT	0xFF58	6 to 65,537	not allowed	optional
Packed packet headers, main header	PPM	0xFF60	9 to 65,537	optional	not allowed
Packed packet headers, tile-part header	PPT	0xFF61	6 to 65,537	not allowed	optional
In-bit-stream markers and marker segments					
Start of packet	SOP	0xFF91	6	not allowed	not allowed optional in-bitstream
End of packet header	EPH	0xFF92	2	optional inside PPM	optional inside PPT or in-bitstream
Informational marker segments					
Component registration	CRG	0xFF63	8 to 65,536	optional	not allowed
Comment	COM	0xFF64	7 to 65,537	optional	optional

Table B.2 – Content element names for SOT

content	length	type	loop
Isot	2	integer	
Psot	4	integer	
TPsot	1	integer	
TNsot	1	integer	

Table B.3 – Content element names for SIZ

content	length	type	loop
Rsiz	2	integer	
Xsiz	4	integer	
Ysiz	4	integer	
OXsiz	4	integer	
OYsiz	4	integer	
XTsiz	4	integer	
YTsiz	4	integer	
XTOsiz	4	integer	
YTOsiz	4	integer	
Csiz	2	integer	
Ssiz	1	integer	]
XRsiz	1	integer	
YRsiz	1	integer	

Table B.4 – Content element names for COD

content	length	type	loop
Scod	1	hexbyte	
SGcod	4	hexbyte	
SPcod	5 – 43	hexbyte	]

Table B.5 – Content element names for COC

content	length	type	loop
Ccoc	1, 2	hexbyte	
Scoc	1	hexbyte	
SPcoc	5 – 43	hexbyte	]

Table B.6 – Content element names for SGcod

SGcod	length	type	loop
progression	1	integer	
num_layers	2	integer	
colour_conv	1	integer	

Table B.7 – Content element names for SPcod/SPcoc

SPcod/SPcoc	length	type	loop
num_levels	1	integer	
xcb	1	integer	
ycb	1	integer	
style	1	integer	
wavelet	1	integer	
ppy	1/2	integer	]
ppx	1/2	integer	

Table B.8 – Content element names for RGN

content	length	type	loop
Crgn	1, 2	integer	
Srgn	1	integer	
SPrgn	1	integer	

Table B.9 – Content element names for QCD

content	length	type	loop
Sqcd	1	integer	
SPqcd	1, 2	integer	]

Table B.10 – Content element names for QCC

content	length	type	loop
Cqcc	1	integer	
Sqcc	1	integer	
SPqcc	1, 2	integer	]

Table B.11 – Content element names for POC

content	length	type	loop
RSpoc	1	integer	]
CSpoc	1, 2	integer	
LYEpoc	2	integer	
REpoc	1	integer	
CEpoc	1, 2	integer	
Ppoc	1	integer	

Table B.12 – Content element names for TLM

content	length	type	loop
Ztlm	1	integer	
Stlm	1	integer	
Ttlm	0 – 2	integer	]
Ptlm	0 – 1	integer	

Table B.13 – Content element names for PLM

content	length	type	loop
Zplm	1	integer	]
Nplm	1	integer	]
Iplm	0-65,531	integer	

Table B.14 – Content element names for PLT

content	length	type	loop
Zplt	1	integer	
Iplt	0-65,531	integer	]

Table B.15 – Content element names for PPM

content	length	type	loop
Zppm	1	integer	
Nppm	2	integer	]
Ippm	0-65,531	integer	]

Table B.16 – Content element names for PPT

content	length	type	loop
Zppt	1	integer	
Ippt	0-65,531	integer	]

Table B.17 – Content element names for SOP

content	length	type	loop
Nsop	2	integer	

Table B.18– Content element names for CRG

content	length	type	loop
Xcrg	2	integer	
Ycrg	2	integer	

Table B.19 – Content element names for COM

content	length	type	loop
Rcom	1	integer	
Ccom	1-65,531	string	

B.2.2 JPEG 2000 Part 2 marker elements

The extended and additional JPEG 2000 Part 2 marker elements shall be used as the element names described in Table B.20. These marker content element names are defined in Tables B.21 to B.38.

Table B.20 – JPEG 2000 Part 2 extended and additional markers

Marker name	Element name	Code	Size	Main header	Tile-part header
Start of tile-part (part 1 extended)	SOT	0xFF90	14	not allowed	required
Coding style default (part 1 extended)	COD	0xFF52	14 to 47	required	optional
Coding style component (part 1 extended)	COC	0xFF53	11 to 45	optional	optional
Region-of-interest (part 1 extended)	RGN	0xFF5E	7 to 24	optional	optional
Variable DC offset	DCO	0xFF70	7 to 32,772	optional	optional
Visual masking	VMS	0xFF71	11	optional	optional
Downsampling factor style	DFS	0xFF72	7 to 65,537	optional	optional
Arbitrary decomposition style	ADS	0xFF73	5 to 65,537	optional	optional
Arbitrary transformation kernels	ATK	0xFF79	11 to 65,537	optional	optional
Component bit depth	CBD	0xFF78	7 to 16,390	optional	optional
Multiple component transformation definition	MCT	0xFF74	8 to 65,537	optional	optional
Multiple component collection	MCC	0xFF75	7 to 65,537	optional	optional
Multiple component transformation ordering	MCO	0xFF77	3 to 260	optional	optional
Non-linearity point transformation	NLT	0xFF76	14 to 65,537	optional	optional
Quantization default, precinct	QPD	0xFF5A	8 to 103	optional	optional
Quantization component, precinct	QPC	0xFF5B	7 to 201	optional	optional

Table B.21 – Content element names for SOT

content	length	type	loop
Isot	2	integer	
Psot	**4**	**integer**	
TPsot	2	integer	
TNsot	2	integer	

Table B.22 – Content element names for COD

content	length	type	loop
Scod	1	hexbyte	
SGcod	4	hexbyte	
SPcod	7 – 45	hexbyte	

Table B.23 – Content element names for COC

content	length	type	loop
Ccoc	1, 2	hexbyte	
Scoc	1	hexbyte	
SPcoc	7 – 45	hexbyte	

Table B.24 – Content element names for Spcod/SPcoc

SPcod/SPcoc	length	type	loop
num_levels	1	integer	
xcb	1	integer	
ycb	1	integer	
style	1	integer	
wavelet	1	integer	
sso	2	integer	
ppy	1/2	integer	
ppx	1/2	integer	

Table B.25 – Content element names for RGN

content	length	type	loop
Crgn	1, 2	integer	
Srgn	1	integer	
SPrgn	17	integer	

Table B.26 – Content element names for SPrgn

SPrgn	length	type	loop
shift	1	integer	
XArgn	4	integer	
YArgn	4	integer	
XBrgn	4	integer	
YBrgn	4	integer	

Table B.27 – Content element names for DCO

content	length	type	loop
Sdco	1	integer	
SPdco	1	integer	❑

Table B.28 – Content element names for VMS

content	length	type	loop
Cvms	2	integer	
Svm	1	integer	
Wvms	1	integer	
Rvm	1	integer	
Avms	1	integer	
Bvms	1	integer	

Table B.29 – Content element names for DFS

content	length	type	loop
Sdfs	2	integer	
Idfs	1	integer	
Ddfs	0-65,530	integer	

Table B.30 – Content element names for ADS

content	length	type	loop
Sads	1	integer	
IOads	1	integer	
DOads	0-65,530	string	
ISads	1	integer	
DSads	0-65,530	string	

Table B.31 – Content element names for ATK

content	length	type	loop
Satk	2	integer	
Katk	0, 1, 2, 4, 8, 16	integer	
Natk	1	integer	
Oatk	0, 1	integer	
Eatk	0, 1	string	
Batk	0, 1, 2, 4, 8, 16	integer	
LCatk	1	integer	
Aatk	1, 2, 4, 8, 16	integer	❑

Table B.32 – Content element names for CBD

content	length	type	loop
Ncbd	2	integer	
BDcbd	1	integer	❑

Table B.33 – Content element names for MCO

content	length	type	loop
Nmco	1	integer	
Imco	1	integer	❑

Table B.34 – Content element names for NLT

content	length	type	loop
Cnlt	2	integer	
BDnlt	1	integer	
Tnlt	1	integer	
STnlt	1	integer	❑

Table B.35 – Content element names for QPD

content	length	type	loop
PLqpd	2	integer	
PPqpd	1, 2	integer	
Sqpd	1	integer	
SPqpd	1	integer	❑

Table B.36 – Content element names for QPC

content	length	type	loop
Cqpc	1, 2	integer	
PLqpc	2	integer	
PPqpc	1, 2	integer	
Sqpc	1	integer	
SPqpc	1	integer	❑

Table B.37 – Content element names for MCT

content	length	type	loop
Zmct	2	integer	
Imct	2	integer	
Ymct	0, 2	integer	
SPmct	2	integer	❑

Table B.38 – Content element names for MCC

content	length	type	loop
Zmcc	2	integer	
Imcc	1	integer	
Ymcc	0, 2	integer	
Qmcc	1	integer	
Xmcc	1	integer	
Nmcc	2	integer	
Cmcc	1, 2	integer	❑
Mmcc	2	integer	
Wmcc	1, 2	integer	❑
Tmcc	3	integer	
Omcc	4	integer	

B.2.3 JPEG 2000 Part 8 marker elements

The extended and additional JPEG 2000 Part 8 marker elements shall be used as the element names described in Table B.39. These marker content element names are defined in Tables B.40 to B.55.

Table B.39 – JPEG 2000 Part 8 additional marker

Marker name	Element name	Code	Size	Main header	Tile-part header
Main security marker	SEC	0xFF65	4 to 65,537	optional	optional
In-codestream security marker	INSEC	0xFF94	4 to 65,537	not allowed	not allowed optional in-bitstream

Table B.40 – Content element names for SEC

content	length	type	loop
Zsec	8 -	integer	
Psec	0, 7 -	hexbyte	
Tool	0 -	hexbyte	❑

Table B.41 – Content element names for INSEC

content	length	type	loop
index	8 -	integer	
Rinsec	0, 7 -	hexbyte	
APinsec	variable	hexbyte	

Table B.42 – Content element names for Psec

Psec	length	type	loop
Fpsec	1 -	hexbyte	
Ntools	1+n	integer	
Imax	1+n	integer	
Ptrlcp	0, 4	hexbyte	

Table B.43 – Content element names for Tool

Tool	length	type	loop
type	1+n	integer	
index	1+n	integer	
IDtool	1+n	hexbyte	
length	2+n	integer	
ZOI	variable	hexbyte	
LPid	2+n	integer	
Pid	variable	hexbyte	

Table B.44 – Content element names for IDtool

IDtool (Non-normative Tool)	length	type	loop
IDraid	1+n	integer	
IDransl	1+n	integer	
IDrans	0, 4	hexbyte	

Table B.45 – Content element names for ZOI

ZOI	length	type	loop
NZzoi	1+n	integer	
Zone	variable	hexbyte	❑

Table B.46 – Content element names for Zone

Zone	length	type	loop
DCzoi	variable	integer	
Pzoi	variable	hexbyte	❑

Table B.47 – Content element names for Pzoi

Pzoi	length	type	loop
Mzoi	variable	hexbyte	
Nzoi	0, 1+n	integer	
Izoi	variable	hexbyte	❑

Table B.48 – Content element names for Pid

Pid (Normative Tool)	length	type	loop
NTid	1+n	integer	
NTdomain	1+n	integer	
NTgranularity	3	integer	
NTvalue	0, 4	hexbyte	

Table B.49 – Content element names for NTdomain

NTdomain	length	type	loop
PD	variable	hexbyte	
Fpd	variable	hexbyte	

Table B.50 – Content element names for NTgranularity

NTgranularity	length	type	loop
PD	2	hexbyte	
GL	1	hexbyte	

Table B.51 – Content element names for NTvalue

NTvalue	length	type	loop
Nv	2+n	integer	
Sv	1+n	integer	
data	variable	hexbyte	

Table B.52 – Content element names for Tid

Tid (Decryption Template)	length	type	loop
MEdecry	1+n	hexbyte	
CTdecry	2	hexbyte	
CPdecry	variable	hexbyte	

Tid (Authentication Temp.)	length	type	loop
MEauth	1+n	hexbyte	
Pauth	variable	hexbyte	

Tid (Hash Template)	length	type	loop
Hhash	1	hexbyte	
SIZhash	1	integer	

Table B.53 – Content element names for CPdecry

CPdecry (Block cipher)	length	type	loop
Mbc	3/4	hexbyte	
Pbc	1/4	hexbyte	
SIZbc	1	integer	
KTntid	variable	hexbyte	

CPdecry (Stream cipher)	length	type	loop
KTntid	variable	hexbyte	

CPdecry (Asymmetric cipher)	length	type	loop
KTntid	variable	hexbyte	

Table B.54 – Content element names for Pauth

Pauth (Hash-based Auth.)	length	type	loop
Mhmac	1	hexbyte	
Hhmac	1	hexbyte	
KTntid	variable	hexbyte	
SIZhmac	2	integer	

Pauth (Cipher-based Auth.)	length	type	loop
CAcmac	1	hexbyte	
Ccmac	1	hexbyte	
KTntid	variable	hexbyte	
SIZcmac	2	integer	

Pauth (Digital Signature)	length	type	loop
Mds	1	hexbyte	
Hds	1	hexbyte	
KTntid	variable	hexbyte	
SIZds	2	integer	

Table B.55 – Content element names for KTntid

KTntid	length	type	loop
LKkt	2	hexbyte	
KIDkt	1	hexbyte	
granularity	3	hexbyte	
data	variable	hexbyte	

B.2.4 JPEG 2000 Part 10 marker elements

The extended and additional JPEG 2000 Part 10 marker elements shall be used as the element names described in Table B.56. These marker content element names are defined in Tables B.57 to B.65.

Table B.56 – JPEG 2000 Part 10 extended and additional markers

Marker name	Element name	Code	Size	Main header	Tile-part header
Coding style default (part 2 extended)	COD	0xFF52	19 to 85	required	optional
Coding style component (part 2 extended)	COC	0xFF53	20 to 104	optional	optional
Region-of-interest (part 2 extended)	RGN	0xFF5E	7 to 32	optional	optional
Component registration (part 1 extended)	CRG	0xFF63	10 to 65,536	optional	not allowed
Additional dimension image and tile size	NSI	0xFF54	22 to 16,405	required	not allowed

Table B.57 – Content element names for NSI

content	length	type	loop
Ndim	1	integer	
Zsiz	4	integer	
ZOsiz	4	integer	
ZTsiz	5	integer	
ZTOsiz	4	integer	
ZRsiz	1	integer	❑

Table B.58 – Content element names for COD

content	length	type	loop
Scod	1	hexbyte	
SGcod	4	hexbyte	
SPcod	10-76	hexbyte	❑

Table B.59 – Content element names for COC

content	length	type	loop
Ccoc	1, 2	integer	
Scoc	1	hexbyte	
SPcoc	10-94	hexbyte	❑

Table B.60 – Content element names for SPcod/SPcoc

SPcod/SPcoc	length	type	loop
num_xlevels	1	integer	
num_ylevels	1	integer	
num_zlevels	1	integer	
xcb	1	integer	
ycb	1	integer	
zcb	1	integer	
style	1	integer	
xkernel	1	integer	
ykernel	1	integer	
zkernel	1	integer	
sso	2	integer	
reserved	1/2	–	
ppz	1/2	integer	
ppy	1/2	integer	
ppx	1/2	integer	

Table B.61 – Content element names for QCD

content	length	type	loop
Sqcd	1	integer	
SPqcd	1, 2	integer	

Table B.62 – Content element names for QCC

content	length	type	loop
Cqcc	1	integer	
Sqcc	1	integer	
SPqcc	1, 2	integer	

Table B.63 – Content element names for RGN

content	length	type	loop
Crgn	2	integer	
Srgn	1	integer	
SPrgn	25	integer	

Table B.64 – Content element names for SPrgn

SPrgn	length	type	loop
shift	1	integer	
XArgn	4	integer	
YArgn	4	integer	
ZArgn	4	integer	
XBrgn	4	integer	
YBrgn	4	integer	
ZBrgn	4	integer	

Table B.65 – Content element names for CRG

content	length	type	loop
Xcrg	2	integer	
Ycrg	2	integer	
Zcrg	2	integer	

B.2.5 JPEG 2000 Part 11 marker elements

The extended and additional JPEG 2000 Part 11 marker elements shall be used as the element names described in Table B.66. These marker content element names are defined in Tables B.67 to B.70.

Table B.66 – JPEG 2000 Part 11 additional markers

Marker name	Element name	Code	Size	Main header	Tile-part header
Error Protection Block	EPB	0xFF66	13 to 65,537	optional	optional
Error Sensitivity Descriptor	ESD	0xFF67	6 to 65,537	optional	optional
Error Protection Capability	EPC	0xFF68	11 to 65,537	required	optional
Residual Errors Descriptor	RED	0xFF69	6 to 65,537	optional	optional

Pid	Lid	hexbyte	

Table B.67 – Content element names for EPB

content	length	type	loop
Depb	1	integer	
LDPepb	4	integer	
Pepb	4	integer	
data	0-65,524	hexbyte	

Table B.68 – Content element names for EPC

content	length	type	loop
Pcrc	2	integer	
DL	4	integer	
Pepc	1	integer	
id	2	hexbyte	
Lid	2	hexbyte	

Table B.69 – Content element names for ESD

content	length	type	loop
Cesd	1, 2	integer	
Pesd	1	integer	
data	0-65,531	hexbyte	

Table B.70 – Content element names for RED

content	length	type	loop
Pred	1	integer	
data	0-65,531	hexbyte	

B.3 Examples of XML schemas

The following examples are XML schemas for the marker elements of the JPEG 2000 family codestream.

B.3.1 Example of an XML schema for a common header

The following example is of common XML schemas for all marker types of JPEG 2000 family codestreams.

```
<?xml version="1.0" ?>
<xs:schema  xmlns:xs="http://www.iso.org/jpeg/2001/XMLSchema"
            targetNamespace="http://www.iso.org/jpxml"
            xmlns="http://www.iso.org/jpxml">
  <xs:attributeGroup name="attrs.marker">
    <xs:attribute name="length"  type="xs:integer"   use="required" />
    <xs:attribute name="offset"  type="xs:integer"   use="optional" />
    <xs:attribute name="type"    use="required">
      <xs:simpleType>
        <xs:restriction base="xs:string">
          <xs:enumeration value="marker" />
        </xs:restriction>
      </xs:simpleType>
    </xs:attribute>
  </xs:attributeGroup>
  <xs:attributeGroup name="attrs.hex">
    <xs:attribute name="length"  type="xs:integer"   use="required" />
    <xs:attribute name="type"    use="required">
      <xs:simpleType>
        <xs:restriction base="xs:string">
          <xs:enumeration value="hexbyte" />
        </xs:restriction>
      </xs:simpleType>
    </xs:attribute>
  </xs:attributeGroup>
  <xs:attributeGroup name="attrs.int">
    <xs:attribute name="length"  type="xs:integer"   use="required" />
    <xs:attribute name="type"    use="required">
      <xs:simpleType>
        <xs:restriction base="xs:string">
          <xs:enumeration value="integer" />
        </xs:restriction>
      </xs:simpleType>
    </xs:attribute>
  </xs:attributeGroup>
  <xs:attributeGroup name="attrs.str">
    <xs:attribute name="length"  type="xs:integer"   use="required" />
    <xs:attribute name="type"    use="required">
      <xs:simpleType>
        <xs:restriction base="xs:string">
          <xs:enumeration value="string" />
        </xs:restriction>
      </xs:simpleType>
    </xs:attribute>
  </xs:attributeGroup>
  <xs:attributeGroup name="attrs.url">
    <xs:attribute name="length"  type="xs:integer"   use="required" />
    <xs:attribute name="type"    use="required">
      <xs:simpleType>
        <xs:restriction base="xs:string">
          <xs:enumeration value="location" />
        </xs:restriction>
      </xs:simpleType>
    </xs:attribute>
  </xs:attributeGroup>
  <xs:attributeGroup name="attrs.root">
    <xs:attribute name="length"  type="xs:integer"   use="required" />
    <xs:attribute name="name"    type="xs:anyURL"    use="optional" />
  </xs:attributeGroup>

  <!-- add following XML Schemas for the JPXML document. -->
   ...
</xs:schema>
```

B.3.2 Example of an XML schema for a JPEG 2000 Part 1 codestream (single image)

```
<!-- part 1 marker element -->
<xs:element name="SOT">
  <xs:complexType>
    <xs:attributeGroup ref="attrs.marker"
                       use="required" />
    <xs:sequence>
      <xs:element ref="Isot" />
      <xs:element ref="Psot" />
      <xs:element ref="TPsot" />
      <xs:element ref="TNsot" />
    </xs:sequence>
  </xs:complexType>
</xs:element>

<xs:element name="SIZ">
  <xs:complexType>
    <xs:attributeGroup ref="attrs.marker"
                       use="required" />
    <xs:sequence>
      <xs:element ref="Rsiz" />
      <xs:element ref="Xsiz" />
      <xs:element ref="Ysiz" />
      <xs:element ref="OXsiz" />
      <xs:element ref="OYsiz" />
      <xs:element ref="XTsiz" />
      <xs:element ref="YTsiz" />
      <xs:element ref="XTOsiz" />
      <xs:element ref="YTOsiz" />
      <xs:sequence maxOccurs="unbounded">
        <xs:element ref="Csiz" />
        <xs:element ref="XRsiz" />
        <xs:element ref="YRsiz" />
      </xs:sequence>
    </xs:sequence>
  </xs:complexType>
</xs:element>

<xs:element name="COD">
  <xs:complexType>
    <xs:attributeGroup ref="attrs.marker"
                       use="required" />
    <xs:sequence>
      <xs:element ref="Scod" />
      <xs:element ref="SGcod" />
      <xs:sequence maxOccurs="unbounded">
        <xs:element ref="SPcod" />
      </xs:sequence>
    </xs:sequence>
  </xs:complexType>
</xs:element>

<xs:element name="COC">
  <xs:complexType>
    <xs:attributeGroup ref="attrs.marker"
                       use="required" />
    <xs:sequence>
      <xs:element ref="Ccoc" />
      <xs:element ref="Scoc" />
      <xs:sequence maxOccurs="unbounded">
        <xs:element ref="SPcoc" />
      </xs:sequence>
    </xs:sequence>
  </xs:complexType>
</xs:element>

<xs:element name="SGcod">
  <xs:complexType>
    <xs:attributeGroup ref="attrs.hex"
                       use="required" />
    <xs:sequence>
      <xs:element ref="progression" />
      <xs:element ref="num layers" />
      <xs:element ref="colour conv" />
    </xs:sequence>
  </xs:complexType>
</xs:element>

<xs:element name="SPcod">
  <xs:complexType>
    <xs:attributeGroup ref="attrs.hex"
                       use="required" />
    <xs:sequence>
      <xs:element ref="num_levels" />
      <xs:element ref="xcb" />
      <xs:element ref="ycb" />
      <xs:element ref="style" />
      <xs:element ref="wavelet" />
      <xs:sequence maxOccurs="unbounded">
        <xs:element ref="ppy" />
        <xs:element ref="ppx" />
      </xs:sequence>
    </xs:sequence>
  </xs:complexType>
</xs:element>

<xs:element name="SPcoc">
  <xs:complexType>
    <xs:attributeGroup ref="attrs.hex"
                       use="required" />
    <xs:sequence>
      <xs:element ref="num_levels" />
      <xs:element ref="xcb" />
      <xs:element ref="ycb" />
      <xs:element ref="style" />
      <xs:element ref="wavelet" />
      <xs:sequence maxOccurs="unbounded">
        <xs:element ref="ppy" />
        <xs:element ref="ppx" />
      </xs:sequence>
    </xs:sequence>
  </xs:complexType>
</xs:element>

<xs:element name="RGN">
  <xs:complexType>
    <xs:attributeGroup ref="attrs.marker"
                       use="required" />
    <xs:sequence>
      <xs:element ref="Crgn" />
      <xs:element ref="Srgn" />
      <xs:element ref="SPrgn" />
    </xs:sequence>
  </xs:complexType>
</xs:element>

<xs:element name="QCD">
  <xs:complexType>
    <xs:attributeGroup ref="attrs.marker"
                       use="required" />
    <xs:sequence>
      <xs:element ref="Sqcd" />
      <xs:sequence maxOccurs="unbounded">
        <xs:element ref="SPqcd" />
      </xs:sequence>
    </xs:sequence>
  </xs:complexType>
</xs:element>

<xs:element name="QCC">
  <xs:complexType>
    <xs:attributeGroup ref="attrs.marker"
                       use="required" />
    <xs:sequence>
      <xs:element ref="Cqcc" />
      <xs:element ref="Sqcc" />
      <xs:sequence maxOccurs="unbounded">
        <xs:element ref="SPqcc" />
      </xs:sequence>
    </xs:sequence>
  </xs:complexType>
</xs:element>

<xs:element name="POC">
  <xs:complexType>
    <xs:attributeGroup ref="attrs.marker"
                       use="required" />
    <xs:sequence maxOccurs="unbounded">
      <xs:element ref="RSpoc" />
      <xs:element ref="CSpoc" />
      <xs:element ref="LYEoc" />
      <xs:element ref="REpoc" />
      <xs:element ref="CEpoc" />
      <xs:element ref="Ppoc" />
    </xs:sequence>
  </xs:complexType>
</xs:element>
```

```
<xs:element name="TLM">
  <xs:complexType>
    <xs:attributeGroup ref="attrs.marker"
                       use="required" />
    <xs:sequence>
      <xs:element ref="Ztlm" />
      <xs:element ref="Stlm" />
      <xs:sequence maxOccurs="unbounded">
        <xs:element ref="Ttlm" />
        <xs:element ref="Ptlm" />
      </xs:sequence>
    </xs:sequence>
  </xs:complexType>
</xs:element>

<xs:element name="PLM">
  <xs:complexType>
    <xs:attributeGroup ref="attrs.marker"
                       use="required" />
    <xs:sequence maxOccurs="unbounded">
      <xs:element ref="Zplm" />
      <xs:sequence maxOccurs="unbounded">
        <xs:element ref="Nplm" />
        <xs:element ref="Iplm" />
      </xs:sequence>
    </xs:sequence>
  </xs:complexType>
</xs:element>

<xs:element name="PLT">
  <xs:complexType>
    <xs:attributeGroup ref="attrs.marker"
                       use="required" />
    <xs:sequence>
      <xs:element ref="Zplm" />
      <xs:sequence maxOccurs="unbounded">
        <xs:element ref="Iplm" />
      </xs:sequence>
    </xs:sequence>
  </xs:complexType>
</xs:element>

<xs:element name="PPM">
  <xs:complexType>
    <xs:attributeGroup ref="attrs.marker"
                       use="required" />
    <xs:sequence>
      <xs:element ref="Zppm" />
      <xs:sequence maxOccurs="unbounded">
        <xs:element ref="Nppm" />
        <xs:sequence maxOccurs="unbounded">
          <xs:element ref="Ippm" />
        </xs:sequence>
      </xs:sequence>
    </xs:sequence>
  </xs:complexType>
</xs:element>

<xs:element name="PPT">
  <xs:complexType>
    <xs:attributeGroup ref="attrs.marker"
                       use="required" />
    <xs:sequence>
      <xs:element ref="Zppt" />
      <xs:sequence maxOccurs="unbounded">
        <xs:element ref="Ippt" />
      </xs:sequence>
    </xs:sequence>
  </xs:complexType>
</xs:element>

<xs:element name="SOP">
  <xs:complexType>
    <xs:attributeGroup ref="attrs.marker"
                       use="required" />
    <xs:sequence>
      <xs:element ref="Nsop" />
    </xs:sequence>
  </xs:complexType>
</xs:element>

<xs:element name="CRG">
  <xs:complexType>
    <xs:attributeGroup ref="attrs.marker"
                       use="required" />
    <xs:sequence maxOccurs="unbounded">
      <xs:element ref="Xcrg" />
      <xs:element ref="Ycrg" />
    </xs:sequence>
  </xs:complexType>
</xs:element>

<xs:element name="COM">
  <xs:complexType>
    <xs:attributeGroup ref="attrs.marker"
                       use="required" />
    <xs:sequence>
      <xs:element ref="Rcom" />
      <xs:element ref="Ccom" />
    </xs:sequence>
  </xs:complexType>
</xs:element>

<!-- part 1 marker element -->
<xs:element name="Isot"  type="xs:integer">
  <xs:attributeGroup ref="attrs.int"
                     use="required" />
</xs:element>

<xs:element name="Psot"  type="xs:integer">
  <xs:attributeGroup ref="attrs.int"
                     use="required" />
</xs:element>

<xs:element name="TPsot"  type="xs:integer">
  <xs:attributeGroup ref="attrs.int"
                     use="required" />
</xs:element>

<xs:element name="TNsot"  type="xs:integer">
  <xs:attributeGroup ref="attrs.int"
                     use="required" />
</xs:element>

<xs:element name="Rsiz"  type="xs:integer">
  <xs:attributeGroup ref="attrs.int"
                     use="required" />
</xs:element>

<xs:element name="Xsiz"  type="xs:integer">
  <xs:attributeGroup ref="attrs.int"
                     use="required" />
</xs:element>

<xs:element name="Ysiz"  type="xs:integer">
  <xs:attributeGroup ref="attrs.int"
                     use="required" />
</xs:element>

<xs:element name="OXsiz"  type="xs:integer">
  <xs:attributeGroup ref="attrs.int"
                     use="required" />
</xs:element>

<xs:element name="OYsiz"  type="xs:integer">
  <xs:attributeGroup ref="attrs.int"
                     use="required" />
</xs:element>

<xs:element name="XTsiz"  type="xs:integer">
  <xs:attributeGroup ref="attrs.int"
                     use="required" />
</xs:element>

<xs:element name="YTsiz"  type="xs:integer">
  <xs:attributeGroup ref="attrs.int"
                     use="required" />
</xs:element>

<xs:element name="XTOsiz"  type="xs:integer">
  <xs:attributeGroup ref="attrs.int"
                     use="required" />
</xs:element>

<xs:element name="YTOsiz"  type="xs:integer">
  <xs:attributeGroup ref="attrs.int"
                     use="required" />
</xs:element>

<xs:element name="Csiz"  type="xs:integer">
  <xs:attributeGroup ref="attrs.int"
                     use="required" />
</xs:element>
```

```
<xs:element name="XRsiz"  type="xs:integer">
  <xs:attributeGroup ref="attrs.int"
                     use="required" />
</xs:element>

<xs:element name="YRsiz"  type="xs:integer">
  <xs:attributeGroup ref="attrs.int"
                     use="required" />
</xs:element>

<xs:element name="Scod" type="hexbyte">
  <xs:attributeGroup ref="attrs.hex"
                     use="required" />
</xs:element>

<xs:element name="SGcod" type="hexbyte">
  <xs:attributeGroup ref="attrs.hex"
                     use="required" />
</xs:element>

<xs:element name="SPcod" type="hexbyte">
  <xs:attributeGroup ref="attrs.hex"
                     use="required" />
</xs:element>

<xs:element name="Ccoc" type="hexbyte">
  <xs:attributeGroup ref="attrs.hex"
                     use="required" />
</xs:element>

<xs:element name="Scoc" type="hexbyte">
  <xs:attributeGroup ref="attrs.hex"
                     use="required" />
</xs:element>

<xs:element name="SPcoc" type="hexbyte">
  <xs:attributeGroup ref="attrs.hex"
                     use="required" />
</xs:element>

<xs:element name="Crgn" type="hexbyte">
  <xs:attributeGroup ref="attrs.hex"
                     use="required" />
</xs:element>

<xs:element name="Srgn" type="hexbyte">
  <xs:attributeGroup ref="attrs.hex"
                     use="required" />
</xs:element>

<xs:element name="SPrgn" type="hexbyte">
  <xs:attributeGroup ref="attrs.hex"
                     use="required" />
</xs:element>

<xs:element name="Sqcd" type="hexbyte">
  <xs:attributeGroup ref="attrs.hex"
                     use="required" />
</xs:element>

<xs:element name="SPqcd" type="hexbyte">
  <xs:attributeGroup ref="attrs.hex"
                     use="required" />
</xs:element>

<xs:element name="Cqcc" type="hexbyte">
  <xs:attributeGroup ref="attrs.hex"
                     use="required" />
</xs:element>

<xs:element name="Sqcc" type="hexbyte">
  <xs:attributeGroup ref="attrs.hex"
                     use="required" />
</xs:element>

<xs:element name="SPqcc" type="hexbyte">
  <xs:attributeGroup ref="attrs.hex"
                     use="required" />
</xs:element>

<xs:element name="progression"
type="xs:integer">
  <xs:attributeGroup ref="attrs.int"
                     use="required" />
</xs:element>
```

```
<xs:element name="num layers"
type="xs:integer">
  <xs:attributeGroup ref="attrs.int"
                     use="required" />
</xs:element>

<xs:element name="colour_conv"
type="xs:integer">
  <xs:attributeGroup ref="attrs.int"
                     use="required" />
</xs:element>

<xs:element name="num levels"
type="xs:integer">
  <xs:attributeGroup ref="attrs.int"
                     use="required" />
</xs:element>

<xs:element name="xcb"  type="xs:integer">
  <xs:attributeGroup ref="attrs.int"
                     use="required" />
</xs:element>

<xs:element name="ycb"  type="xs:integer">
  <xs:attributeGroup ref="attrs.int"
                     use="required" />
</xs:element>

<xs:element name="style"  type="xs:integer">
  <xs:attributeGroup ref="attrs.int"
                     use="required" />
</xs:element>

<xs:element name="wavelet"  type="xs:integer">
  <xs:attributeGroup ref="attrs.int"
                     use="required" />
</xs:element>

<xs:element name="ppy"  type="xs:integer">
  <xs:attributeGroup ref="attrs.int"
                     use="required" />
</xs:element>

<xs:element name="ppx"  type="xs:integer">
  <xs:attributeGroup ref="attrs.int"
                     use="required" />
</xs:element>

<xs:element name="Crgn"  type="xs:integer">
  <xs:attributeGroup ref="attrs.int"
                     use="required" />
</xs:element>

<xs:element name="Srgn"  type="xs:integer">
  <xs:attributeGroup ref="attrs.int"
                     use="required" />
</xs:element>

<xs:element name="SPrgn"  type="xs:integer">
  <xs:attributeGroup ref="attrs.int"
                     use="required" />
</xs:element>

<xs:element name="Sqcd"  type="xs:integer">
  <xs:attributeGroup ref="attrs.int"
                     use="required" />
</xs:element>

<xs:element name="SPqcd"  type="xs:integer">
  <xs:attributeGroup ref="attrs.int"
                     use="required" />
</xs:element>

<xs:element name="Cqcc"  type="xs:integer">
  <xs:attributeGroup ref="attrs.int"
                     use="required" />
</xs:element>

<xs:element name="Sqcc"  type="xs:integer">
  <xs:attributeGroup ref="attrs.int"
                     use="required" />
</xs:element>

<xs:element name="SPqcc"  type="xs:integer">
  <xs:attributeGroup ref="attrs.int"
                     use="required" />
</xs:element>
```

```
<xs:element name="RSpoc"  type="xs:integer">
  <xs:attributeGroup ref="attrs.int"
                     use="required" />
</xs:element>

<xs:element name="CSpoc"  type="xs:integer">
  <xs:attributeGroup ref="attrs.int"
                     use="required" />
</xs:element>

<xs:element name="LYEoc"  type="xs:integer">
  <xs:attributeGroup ref="attrs.int"
                     use="required" />
</xs:element>

<xs:element name="REpoc"  type="xs:integer">
  <xs:attributeGroup ref="attrs.int"
                     use="required" />
</xs:element>

<xs:element name="CEpoc"  type="xs:integer">
  <xs:attributeGroup ref="attrs.int"
                     use="required" />
</xs:element>

<xs:element name="Ppoc"  type="xs:integer">
  <xs:attributeGroup ref="attrs.int"
                     use="required" />
</xs:element>

<xs:element name="Ztlm"  type="xs:integer">
  <xs:attributeGroup ref="attrs.int"
                     use="required" />
</xs:element>

<xs:element name="Stlm"  type="xs:integer">
  <xs:attributeGroup ref="attrs.int"
                     use="required" />
</xs:element>

<xs:element name="Ttlm"  type="xs:integer">
  <xs:attributeGroup ref="attrs.int"
                     use="required" />
</xs:element>

<xs:element name="Ptlm"  type="xs:integer">
  <xs:attributeGroup ref="attrs.int"
                     use="required" />
</xs:element>

<xs:element name="Zplm"  type="xs:integer">
  <xs:attributeGroup ref="attrs.int"
                     use="required" />
</xs:element>

<xs:element name="Nplm"  type="xs:integer">
  <xs:attributeGroup ref="attrs.int"
                     use="required" />
</xs:element>

<xs:element name="Iplm"  type="xs:integer">
  <xs:attributeGroup ref="attrs.int"
                     use="required" />
</xs:element>

<xs:element name="Zplm"  type="xs:integer">
  <xs:attributeGroup ref="attrs.int"
                     use="required" />
</xs:element>

<xs:element name="Iplm"  type="xs:integer">
  <xs:attributeGroup ref="attrs.int"
                     use="required" />
</xs:element>

<xs:element name="Zppm"  type="xs:integer">
  <xs:attributeGroup ref="attrs.int"
                     use="required" />
</xs:element>

<xs:element name="Nppm"  type="xs:integer">
  <xs:attributeGroup ref="attrs.int"
                     use="required" />
</xs:element>

<xs:element name="Ippm"  type="xs:integer">
  <xs:attributeGroup ref="attrs.int"
                     use="required" />
</xs:element>

<xs:element name="Zppt"  type="xs:integer">
  <xs:attributeGroup ref="attrs.int"
                     use="required" />
</xs:element>

<xs:element name="Ippt"  type="xs:integer">
  <xs:attributeGroup ref="attrs.int"
                     use="required" />
</xs:element>

<xs:element name="Nsop"  type="xs:integer">
  <xs:attributeGroup ref="attrs.int"
                     use="required" />
</xs:element>

<xs:element name="Xcrg"  type="xs:integer">
  <xs:attributeGroup ref="attrs.int"
                     use="required" />
</xs:element>

<xs:element name="Ycrg"  type="xs:integer">
  <xs:attributeGroup ref="attrs.int"
                     use="required" />
</xs:element>

<xs:element name="Rcom"  type="xs:integer">
  <xs:attributeGroup ref="attrs.int"
                     use="required" />
</xs:element>

<xs:element name="Ccom"  type="xs:string">
  <xs:attributeGroup ref="attrs.str"
                     use="required" />
</xs:element>
```

B.3.3 Example of an XML schema for a JPEG 2000 Part 2 codestream (single/layered image)

```
<!-- part 1 marker element -->
…
<!-- part 2 marker element -->
<xs:element name="SPcod">
  <xs:complexType>
    <xs:attributeGroup ref="attrs.hex"
                       use="required" />
    <xs:sequence>
      <xs:element ref="num levels" />
      <xs:element ref="xcb" />
      <xs:element ref="ycb" />
      <xs:element ref="style" />
      <xs:element ref="wavelet" />
      <xs:element ref="sso" />
      <xs:sequence maxOccurs="unbounded">
        <xs:element ref="ppy" />
        <xs:element ref="ppx" />
      </xs:sequence>
    </xs:sequence>
  </xs:complexType>
</xs:element>

<xs:element name="SPcoc">
  <xs:complexType>
    <xs:attributeGroup ref="attrs.hex"
                       use="required" />
    <xs:sequence>
      <xs:element ref="num levels" />
      <xs:element ref="xcb" />
      <xs:element ref="ycb" />
      <xs:element ref="style" />
      <xs:element ref="wavelet" />
      <xs:element ref="sso" />
      <xs:sequence maxOccurs="unbounded">
        <xs:element ref="ppy" />
        <xs:element ref="ppx" />
      </xs:sequence>
    </xs:sequence>
  </xs:complexType>
```

```
</xs:element>

<xs:element name="SPrgn">
  <xs:complexType>
    <xs:attributeGroup ref="attrs.hex"
                       use="required" />
    <xs:sequence>
      <xs:element ref="shift" />
      <xs:element ref="XArgn" />
      <xs:element ref="YArgn" />
      <xs:element ref="XBrgn" />
      <xs:element ref="YBrgn" />
    </xs:sequence>
  </xs:complexType>
</xs:element>

<xs:element name="DCO">
  <xs:complexType>
    <xs:attributeGroup ref="attrs.marker"
                       use="required" />
    <xs:sequence>
      <xs:element ref="Sdco" />
      <xs:sequence maxOccurs="unbounded">
        <xs:element ref="SPdco" />
      </xs:sequence>
    </xs:sequence>
  </xs:complexType>
</xs:element>

<xs:element name="VMS">
  <xs:complexType>
    <xs:attributeGroup ref="attrs.marker"
                       use="required" />
    <xs:sequence>
      <xs:element ref="Cvms" />
      <xs:element ref="Svms" />
      <xs:element ref="Wvms" />
      <xs:element ref="Rvms" />
      <xs:element ref="Avms" />
      <xs:element ref="Bvms" />
    </xs:sequence>
  </xs:complexType>
</xs:element>

<xs:element name="VMS">
  <xs:complexType>
    <xs:attributeGroup ref="attrs.marker"
                       use="required" />
    <xs:sequence>
      <xs:element ref="Cvms" />
      <xs:element ref="Svms" />
      <xs:element ref="Wvms" />
      <xs:element ref="Rvms" />
      <xs:element ref="Avms" />
      <xs:element ref="Bvms" />
    </xs:sequence>
  </xs:complexType>
</xs:element>

<xs:element name="DFS">
  <xs:complexType>
    <xs:attributeGroup ref="attrs.marker"
                       use="required" />
    <xs:sequence>
      <xs:element ref="Sdfs" />
      <xs:element ref="Idfs" />
      <xs:element ref="Ddfs" />
    </xs:sequence>
  </xs:complexType>
</xs:element>

<xs:element name="ADS">
  <xs:complexType>
    <xs:attributeGroup ref="attrs.marker"
                       use="required" />
    <xs:sequence>
      <xs:element ref="Sads" />
      <xs:element ref="IOads" />
      <xs:element ref="DOads" />
      <xs:element ref="ISads" />
      <xs:element ref="DSads" />
    </xs:sequence>
  </xs:complexType>
</xs:element>

<xs:element name="ATK">
  <xs:complexType>
    <xs:attributeGroup ref="attrs.marker"
                       use="required" />
    <xs:sequence>
      <xs:element ref="Satk" />
      <xs:element ref="Katk" />
      <xs:element ref="Natk" />
      <xs:sequence maxOccurs="unbounded">
        <xs:element ref="Oatk" />
        <xs:element ref="Eatk" />
        <xs:element ref="Batk" />
        <xs:element ref="LCatk" />
        <xs:sequence maxOccurs="unbounded">
          <xs:element ref="Aatk" />
        </xs:sequence>
      </xs:sequence>
    </xs:sequence>
  </xs:complexType>
</xs:element>

<xs:element name="CBD">
  <xs:complexType>
    <xs:attributeGroup ref="attrs.marker"
                       use="required" />
    <xs:sequence>
      <xs:element ref="Ncbd" />
      <xs:sequence maxOccurs="unbounded">
        <xs:element ref="BDcbd" />
      </xs:sequence>
    </xs:sequence>
  </xs:complexType>
</xs:element>

<xs:element name="MCT">
  <xs:complexType>
    <xs:attributeGroup ref="attrs.marker"
                       use="required" />
    <xs:sequence>
      <xs:element ref="Zmct" />
      <xs:element ref="Imct" />
      <xs:element ref="Ymct" />
      <xs:sequence maxOccurs="unbounded">
        <xs:element ref="SPmct" />
      </xs:sequence>
    </xs:sequence>
  </xs:complexType>
</xs:element>

<xs:element name="MCC">
  <xs:complexType>
    <xs:attributeGroup ref="attrs.marker"
                       use="required" />
    <xs:sequence>
      <xs:element ref="Zmcc" />
      <xs:element ref="Imcc" />
      <xs:element ref="Ymcc" />
      <xs:element ref="Qmcc" />
      <xs:sequence maxOccurs="unbounded">
        <xs:element ref="Xmcc" />
        <xs:element ref="Nmcc" />
        <xs:sequence maxOccurs="unbounded">
          <xs:element ref="Cmcc" />
        </xs:sequence>
        <xs:sequence maxOccurs="unbounded">
          <xs:element ref="Mmcc" />
        </xs:sequence>
        <xs:sequence maxOccurs="unbounded">
          <xs:element ref="Wmcc" />
        </xs:sequence>
        <xs:element ref="Tmcc" />
        <xs:element ref="Omcc" />
      </xs:sequence>
    </xs:sequence>
  </xs:complexType>
</xs:element>

<xs:element name="MCO">
  <xs:complexType>
    <xs:attributeGroup ref="attrs.marker"
                       use="required" />
    <xs:sequence>
      <xs:element ref="Nmco" />
      <xs:sequence maxOccurs="unbounded">
        <xs:element ref="Imco" />
      </xs:sequence>
    </xs:sequence>
  </xs:complexType>
</xs:element>
```

```
<xs:element name="NLT">
  <xs:complexType>
    <xs:attributeGroup ref="attrs.marker"
                       use="required" />
    <xs:sequence>
      <xs:element ref="Cnlt" />
      <xs:element ref="BDnlt" />
      <xs:element ref="Tnlt" />
      <xs:sequence maxOccurs="unbounded">
        <xs:element ref="STnlt" />
      </xs:sequence>
    </xs:sequence>
  </xs:complexType>
</xs:element>

<xs:element name="QPD">
  <xs:complexType>
    <xs:attributeGroup ref="attrs.marker"
                       use="required" />
    <xs:sequence>
      <xs:element ref="PLqpd" />
      <xs:element ref="PPqpd" />
      <xs:element ref="Sqpd" />
      <xs:sequence maxOccurs="unbounded">
        <xs:element ref="SPqpd" />
      </xs:sequence>
    </xs:sequence>
  </xs:complexType>
</xs:element>

<xs:element name="QPC">
  <xs:complexType>
    <xs:attributeGroup ref="attrs.marker"
                       use="required" />
    <xs:sequence>
      <xs:element ref="Cqpc" />
      <xs:element ref="PLqpc" />
      <xs:element ref="PPqpc" />
      <xs:element ref="Sqpc" />
      <xs:sequence maxOccurs="unbounded">
        <xs:element ref="SPqpc" />
      </xs:sequence>
    </xs:sequence>
  </xs:complexType>
</xs:element>

<!-- part 1 content element -->
…
<!-- part 2 content element -->
<xs:element name="sso" type="xs:integer">
  <xs:attributeGroup ref="attrs.int"
                     use="required" />
</xs:element>

<xs:element name="shift" type="xs:integer">
  <xs:attributeGroup ref="attrs.int"
                     use="required" />
</xs:element>

<xs:element name="XArgn" type="xs:integer">
  <xs:attributeGroup ref="attrs.int"
                     use="required" />
</xs:element>

<xs:element name="YArgn" type="xs:integer">
  <xs:attributeGroup ref="attrs.int"
                     use="required" />
</xs:element>

<xs:element name="XBrgn" type="xs:integer">
  <xs:attributeGroup ref="attrs.int"
                     use="required" />
</xs:element>

<xs:element name="YBrgn" type="xs:integer">
  <xs:attributeGroup ref="attrs.int"
                     use="required" />
</xs:element>

<xs:element name="Sdco" type="xs:integer">
  <xs:attributeGroup ref="attrs.int"
                     use="required" />
</xs:element>

<xs:element name="SPdco" type="xs:integer">
  <xs:attributeGroup ref="attrs.int"
                     use="required" />
</xs:element>

<xs:element name="Cvms" type="xs:integer">
  <xs:attributeGroup ref="attrs.int"
                     use="required" />
</xs:element>

<xs:element name="Svms" type="xs:integer">
  <xs:attributeGroup ref="attrs.int"
                     use="required" />
</xs:element>

<xs:element name="Wvms" type="xs:integer">
  <xs:attributeGroup ref="attrs.int"
                     use="required" />
</xs:element>

<xs:element name="Rvms" type="xs:integer">
  <xs:attributeGroup ref="attrs.int"
                     use="required" />
</xs:element>

<xs:element name="Avms" type="xs:integer">
  <xs:attributeGroup ref="attrs.int"
                     use="required" />
</xs:element>

<xs:element name="Bvms" type="xs:integer">
  <xs:attributeGroup ref="attrs.int"
                     use="required" />
</xs:element>

<xs:element name="Sads" type="xs:integer">
  <xs:attributeGroup ref="attrs.int"
                     use="required" />
</xs:element>

<xs:element name="IOads" type="xs:integer">
  <xs:attributeGroup ref="attrs.int"
                     use="required" />
</xs:element>

<xs:element name="DOads" type="xs:integer">
  <xs:attributeGroup ref="attrs.int"
                     use="required" />
</xs:element>

<xs:element name="ISads" type="xs:integer">
  <xs:attributeGroup ref="attrs.int"
                     use="required" />
</xs:element>

<xs:element name="DSads" type="xs:integer">
  <xs:attributeGroup ref="attrs.int"
                     use="required" />
</xs:element>

<xs:element name="Satk" type="xs:integer">
  <xs:attributeGroup ref="attrs.int"
                     use="required" />
</xs:element>

<xs:element name="Katk" type="xs:integer">
  <xs:attributeGroup ref="attrs.int"
                     use="required" />
</xs:element>

<xs:element name="Natk" type="xs:integer">
  <xs:attributeGroup ref="attrs.int"
                     use="required" />
</xs:element>

<xs:element name="Oatk" type="xs:integer">
  <xs:attributeGroup ref="attrs.int"
                     use="required" />
</xs:element>

<xs:element name="Eatk" type="xs:integer">
  <xs:attributeGroup ref="attrs.int"
                     use="required" />
</xs:element>

<xs:element name="Batk" type="xs:integer">
  <xs:attributeGroup ref="attrs.int"
                     use="required" />
</xs:element>
```

```
<xs:element name="LCatk" type="xs:integer">
  <xs:attributeGroup ref="attrs.int"
                     use="required" />
</xs:element>

<xs:element name="Aatk" type="xs:integer">
  <xs:attributeGroup ref="attrs.int"
                     use="required" />
</xs:element>

<xs:element name="Ncbd" type="xs:integer">
  <xs:attributeGroup ref="attrs.int"
                     use="required" />
</xs:element>

<xs:element name="BDcbd" type="xs:integer">
  <xs:attributeGroup ref="attrs.int"
                     use="required" />
</xs:element>

<xs:element name="Zmct" type="xs:integer">
  <xs:attributeGroup ref="attrs.int"
                     use="required" />
</xs:element>

<xs:element name="Imct" type="xs:integer">
  <xs:attributeGroup ref="attrs.int"
                     use="required" />
</xs:element>

<xs:element name="Ymct" type="xs:integer">
  <xs:attributeGroup ref="attrs.int"
                     use="required" />
</xs:element>

<xs:element name="SPmct" type="xs:integer">
  <xs:attributeGroup ref="attrs.int"
                     use="required" />
</xs:element>

<xs:element name="Zmcc" type="xs:integer">
  <xs:attributeGroup ref="attrs.int"
                     use="required" />
</xs:element>

<xs:element name="Imcc" type="xs:integer">
  <xs:attributeGroup ref="attrs.int"
                     use="required" />
</xs:element>

<xs:element name="Ymcc" type="xs:integer">
  <xs:attributeGroup ref="attrs.int"
                     use="required" />
</xs:element>

<xs:element name="Qmcc" type="xs:integer">
  <xs:attributeGroup ref="attrs.int"
                     use="required" />
</xs:element>

<xs:element name="Xmcc" type="xs:integer">
  <xs:attributeGroup ref="attrs.int"
                     use="required" />
</xs:element>

<xs:element name="Nmcc" type="xs:integer">
  <xs:attributeGroup ref="attrs.int"
                     use="required" />
</xs:element>

<xs:element name="Cmcc" type="xs:integer">
  <xs:attributeGroup ref="attrs.int"
                     use="required" />
</xs:element>

<xs:element name="Mmcc" type="xs:integer">
  <xs:attributeGroup ref="attrs.int"
                     use="required" />
</xs:element>

<xs:element name="Wmcc" type="xs:integer">
  <xs:attributeGroup ref="attrs.int"
                     use="required" />
</xs:element>

<xs:element name="Tmcc" type="xs:integer">
  <xs:attributeGroup ref="attrs.int"
                     use="required" />
</xs:element>

<xs:element name="Omcc" type="xs:integer">
  <xs:attributeGroup ref="attrs.int"
                     use="required" />
</xs:element>

<xs:element name="MCO">type="xs:integer">
  <xs:attributeGroup ref="attrs.int"
                     use="required" />
</xs:element>

<xs:element name="Nmco" type="xs:integer">
  <xs:attributeGroup ref="attrs.int"
                     use="required" />
</xs:element>

<xs:element name="Imco" type="xs:integer">
  <xs:attributeGroup ref="attrs.int"
                     use="required" />
</xs:element>

<xs:element name="Cnlt" type="xs:integer">
  <xs:attributeGroup ref="attrs.int"
                     use="required" />
</xs:element>

<xs:element name="BDnlt" type="xs:integer">
  <xs:attributeGroup ref="attrs.int"
                     use="required" />
</xs:element>

<xs:element name="Tnlt" type="xs:integer">
  <xs:attributeGroup ref="attrs.int"
                     use="required" />
</xs:element>

<xs:element name="STnlt" type="xs:integer">
  <xs:attributeGroup ref="attrs.int"
                     use="required" />
</xs:element>

<xs:element name="PLqpd" type="xs:integer">
  <xs:attributeGroup ref="attrs.int"
                     use="required" />
</xs:element>

<xs:element name="PPqpd" type="xs:integer">
  <xs:attributeGroup ref="attrs.int"
                     use="required" />
</xs:element>

<xs:element name="Sqpd" type="xs:integer">
  <xs:attributeGroup ref="attrs.int"
                     use="required" />
</xs:element>

<xs:element name="SPqpd" type="xs:integer">
  <xs:attributeGroup ref="attrs.int"
                     use="required" />
</xs:element>

<xs:element name="Cqpc" type="xs:integer">
  <xs:attributeGroup ref="attrs.int"
                     use="required" />
</xs:element>

<xs:element name="PLqpc" type="xs:integer">
  <xs:attributeGroup ref="attrs.int"
                     use="required" />
</xs:element>

<xs:element name="PPqpc" type="xs:integer">
  <xs:attributeGroup ref="attrs.int"
                     use="required" />
</xs:element>

<xs:element name="Sqpc" type="xs:integer">
  <xs:attributeGroup ref="attrs.int"
                     use="required" />
</xs:element>

<xs:element name="SPqpc" type="xs:integer">
  <xs:attributeGroup ref="attrs.int"
                     use="required" />
```

```
</xs:element>
```

B.3.4 Example of an XML schema for a JPEG 2000 Part 8 codestream (security)

```
<!-- part 1 marker element -->
…
<!-- part 8 marker element -->
<xs:element name="INSEC">
  <xs:complexType>
    <xs:attributeGroup ref="attrs.marker"
                       use="required" />
    <xs:sequence>
      <xs:element ref="index" />
      <xs:element ref="Rinsec" />
      <xs:element ref="APinsec" />
    </xs:sequence>
  </xs:complexType>
</xs:element>

<xs:element name="SEC">
  <xs:complexType>
    <xs:attributeGroup ref="attrs.marker"
                       use="required" />
    <xs:sequence>
      <xs:element ref="Zsec" />
      <xs:element ref="Psec" />
      <xs:sequence maxOccurs="unbounded">
        <xs:element ref="Tool" />
      </xs:sequence>
    </xs:sequence>
  </xs:complexType>
</xs:element>

<xs:element name="Psec">
  <xs:complexType>
    <xs:attributeGroup ref="attr.hex"
                       use="required" />
    <xs:sequence>
      <xs:element ref="Fpsec" />
      <xs:element ref="Ntool" />
      <xs:element ref="Imax" />
      <xs:element ref="Ptrlcp" />
    </xs:sequence>
  </xs:complexType>
</xs:element>

<xs:element name="Tool">
  <xs:complexType>
    <xs:attributeGroup ref="attr.hex"
                       use="required" />
    <xs:sequence>
      <xs:element ref="type" />
      <xs:element ref="index" />
      <xs:element ref="IDtool" />
      <xs:element ref="length" />
      <xs:element ref="ZOI" />
      <xs:element ref="LPid" />
      <xs:element ref="Pid" />
    </xs:sequence>
  </xs:complexType>
</xs:element>

<!-- IDtool for non-normative tools -->
<xs:element name="IDtool">
  <xs:complexType>
    <xs:attributeGroup ref="attr.hex"
                       use="required" />
    <xs:sequence>
      <xs:element ref="IDaid" />
      <xs:element ref="IDransl" />
      <xs:element ref="IDrans" />
    </xs:sequence>
  </xs:complexType>
</xs:element>

<xs:element name="ZOI">
  <xs:complexType>
    <xs:attributeGroup ref="attr.hex"
                       use="required" />
    <xs:sequence>
      <xs:element ref="NZzoi" />
      <xs:element ref="Zone"
                  maxOccurs="unbounded" />
    </xs:sequence>
  </xs:complexType>
</xs:element>

<xs:element name="Zone">
  <xs:complexType>
    <xs:attributeGroup ref="attr.hex"
                       use="required" />
    <xs:sequence>
      <xs:element ref="DCzoi" />
      <xs:element ref="Pzoi"
                  maxOccurs="unbounded" />
    </xs:sequence>
  </xs:complexType>
</xs:element>

<xs:element name="Pzoi">
  <xs:complexType>
    <xs:attributeGroup ref="attr.hex"
                       use="required" />
    <xs:sequence>
      <xs:element ref="Mzoi" />
      <xs:element ref="Nzoi" />
      <xs:element ref="Izoi"
                  maxOccurs="unbounded" />
    </xs:sequence>
  </xs:complexType>
</xs:element>

<!-- Pid for normative tools -->
<xs:element name="Pid">
  <xs:complexType>
    <xs:attributeGroup ref="attr.hex"
                       use="required" />
    <xs:sequence>
      <xs:element ref="NTid" />
      <xs:element ref="NTdomain" />
      <xs:element ref="NTgranularity" />
      <xs:element ref="NTvalue" />
    </xs:sequence>
  </xs:complexType>
</xs:element>

<xs:element name="NTdomain">
  <xs:complexType>
    <xs:attributeGroup ref="attr.hex"
                       use="required" />
    <xs:sequence>
      <xs:element ref="PD" />
      <xs:element ref="Fpd" />
    </xs:sequence>
  </xs:complexType>
</xs:element>

<xs:element name="NTgranularity">
  <xs:complexType>
    <xs:attributeGroup ref="attr.hex"
                       use="required" />
    <xs:sequence>
      <xs:element ref="PD" />
      <xs:element ref="GL" />
    </xs:sequence>
  </xs:complexType>
</xs:element>

<xs:element name="NTvalue">
  <xs:complexType>
    <xs:attributeGroup ref="attr.hex"
                       use="required" />
    <xs:sequence>
      <xs:element ref="Nv" />
      <xs:element ref="Sv" />
      <xs:element ref="data" />
    </xs:sequence>
  </xs:complexType>
</xs:element>

<xs:element name="NTid">
  <xs:complexType>
    <xs:attributeGroup ref="attr.hex"
                       use="required" />
    <xs:choice>
      <xs:sequence>
```

```
        <xs:element ref="MEdecry" />
        <xs:element ref="CTdecry" />
        <xs:element ref="CPdecry" />
      </xs:sequence>
      <xs:sequence>
        <xs:element ref="MEauth" />
        <xs:element ref="Pauth" />
      </xs:sequence>
      <xs:sequence>
        <xs:element ref="Hhash" />
        <xs:element ref="SIZhash" />
      </xs:sequence>
    </xs:choice>
  </xs:complexType>
</xs:element>

<xs:element name="CPdecry">
  <xs:complexType>
    <xs:attributeGroup ref="attr.hex"
                       use="required" />
    <xs:choice>
      <xs:sequence>
        <xs:element ref="Mbc" />
        <xs:element ref="Pbc" />
        <xs:element ref="SIZbc" />
        <xs:element ref="KTntid" />
      </xs:sequence>
      <xs:sequence>
        <xs:element ref="KTntid" />
      </xs:sequence>
    </xs:choice>
  </xs:complexType>
</xs:element>

<xs:element name="Pauth">
  <xs:complexType>
    <xs:attributeGroup ref="attr.hex"
                       use="required" />
    <xs:choice>
      <xs:sequence>
        <xs:element ref="Mhash" />
        <xs:element ref="Hhash" />
        <xs:element ref="KTntid" />
        <xs:element ref="SIZhash" />
      </xs:sequence>
      <xs:sequence>
        <xs:element ref="CAcmac" />
        <xs:element ref="Ccmac" />
        <xs:element ref="KTntid" />
        <xs:element ref="SIZcmac" />
      </xs:sequence>
      <xs:sequence>
        <xs:element ref="Mds" />
        <xs:element ref="Hds" />
        <xs:element ref="KTntid" />
        <xs:element ref="SIZds" />
      </xs:sequence>
    </xs:choice>
  </xs:complexType>
</xs:element>

<xs:element name="KTntid">
  <xs:complexType>
    <xs:attributeGroup ref="attr.hex"
                       use="required" />
    <xs:sequence>
      <xs:element ref="LKkt" />
      <xs:element ref="KIDkt" />
      <xs:element ref="granularity" />
      <xs:element ref="data" />
    </xs:sequence>
  </xs:complexType>
</xs:element>

<!-- part 1 content element -->
…
<!-- part 8 content element -->
<xs:element name="index" type="xs:integer">
  <xs:attributeGroup ref="attrs.int"
                     use="required" />
</xs:element>

<xs:element name="Rinsec" type="hexbyte">
  <xs:attributeGroup ref="attrs.hex"
                     use="required" />
</xs:element>
```

```
<xs:element name="APinsec" type="hexbyte">
  <xs:attributeGroup ref="attrs.hex"
                     use="required" />
</xs:element>

<xs:element name="Zsec" type="xs:integer">
  <xs:attributeGroup ref="attrs.int"
                     use="required" />
</xs:element>

<xs:element name="Psec" type="hexbyte">
  <xs:attributeGroup ref="attrs.hex"
                     use="required" />
</xs:element>

<xs:element name="Tool" type="hexbyte">
  <xs:attributeGroup ref="attrs.hex"
                     use="required" />
</xs:element>

<xs:element name="Fpsec" type="hexbyte">
  <xs:attributeGroup ref="attrs.hex"
                     use="required" />
</xs:element>

<xs:element name="Ntool" type="xs:integer">
  <xs:attributeGroup ref="attrs.int"
                     use="required" />
</xs:element>

<xs:element name="Imax" type="xs:integer">
  <xs:attributeGroup ref="attrs.int"
                     use="required" />
</xs:element>

<xs:element name="Ptrlcp" type="hexbyte">
  <xs:attributeGroup ref="attrs.hex"
                     use="required" />
</xs:element>

<xs:element name="type" type="xs:integer">
  <xs:attributeGroup ref="attrs.int"
                     use="required" />
</xs:element>

<xs:element name="index" type="xs:integer">
  <xs:attributeGroup ref="attrs.int"
                     use="required" />
</xs:element>

<xs:element name="IDtool" type="hexbyte">
  <xs:attributeGroup ref="attrs.hex"
                     use="required" />
</xs:element>

<xs:element name="length" type="xs:integer">
  <xs:attributeGroup ref="attrs.int"
                     use="required" />
</xs:element>

<xs:element name="ZOI" type="hexbyte">
  <xs:attributeGroup ref="attrs.hex"
                     use="required" />
</xs:element>

<xs:element name="LPid" type="xs:integer">
  <xs:attributeGroup ref="attrs.int"
                     use="required" />
</xs:element>

<xs:element name="Pid" type="hexbyte">
  <xs:attributeGroup ref="attrs.hex"
                     use="required" />
</xs:element>

<xs:element name="IDaid" type="xs:integer">
  <xs:attributeGroup ref="attrs.int"
                     use="required" />
</xs:element>

<xs:element name="IDansi" type="xs:integer">
  <xs:attributeGroup ref="attrs.int"
                     use="required" />
</xs:element>

<xs:element name="IDrans" type="hexbyte">
  <xs:attributeGroup ref="attrs.hex"
```

```
                    use="required" />
</xs:element>

<xs:element name="NZzoi" type="xs:integer">
  <xs:attributeGroup ref="attrs.int"
                    use="required" />
</xs:element>

<xs:element name="Zone" type="hexbyte">
  <xs:attributeGroup ref="attrs.hex"
                    use="required" />
</xs:element>

<xs:element name="DCzoi" type="xs:integer">
  <xs:attributeGroup ref="attrs.int"
                    use="required" />
</xs:element>

<xs:element name="Pzoi" type="hexbyte">
  <xs:attributeGroup ref="attrs.hex"
                    use="required" />
</xs:element>

<xs:element name="Mzoi" type="hexbyte">
  <xs:attributeGroup ref="attrs.hex"
                    use="required" />
</xs:element>

<xs:element name="Nzoi" type="xs:integer">
  <xs:attributeGroup ref="attrs.int"
                    use="required" />
</xs:element>

<xs:element name="Izoi"type="hexbyte">
  <xs:attributeGroup ref="attrs.hex"
                    use="required" />
</xs:element>

<xs:element name="NTid" type="hexbyte">
  <xs:attributeGroup ref="attrs.hex"
                    use="required" />
</xs:element>

<xs:element name="NTdomain" type="hexbyte">
  <xs:attributeGroup ref="attrs.hex"
                    use="required" />
</xs:element>

<xs:element name="NTgranularity"
type="hexbyte">
  <xs:attributeGroup ref="attrs.hex"
                    use="required" />
</xs:element>

<xs:element name="NTvalue" type="hexbyte">
  <xs:attributeGroup ref="attrs.hex"
                    use="required" />
</xs:element>

<xs:element name="PD" type="hexbyte">
  <xs:attributeGroup ref="attrs.hex"
                    use="required" />
</xs:element>

<xs:element name="Fpd" type="hexbyte">
  <xs:attributeGroup ref="attrs.hex"
                    use="required" />
</xs:element>

<xs:element name="PD" type="hexbyte">
  <xs:attributeGroup ref="attrs.hex"
                    use="required" />
</xs:element>

<xs:element name="GL" type="hexbyte">
  <xs:attributeGroup ref="attrs.hex"
                    use="required" />
</xs:element>

<xs:element name="Nv" type="xs:integer">
  <xs:attributeGroup ref="attrs.int"
                    use="required" />
</xs:element>

<xs:element name="Sv" type="xs:integer">
  <xs:attributeGroup ref="attrs.int"
                    use="required" />
</xs:element>

<xs:element name="MEdecry" type="hexbyte">
  <xs:attributeGroup ref="attrs.hex"
                    use="required" />
</xs:element>

<xs:element name="CTdecry" type="hexbyte">
  <xs:attributeGroup ref="attrs.hex"
                    use="required" />
</xs:element>

<xs:element name="CPdecry" type="hexbyte">
  <xs:attributeGroup ref="attrs.hex"
                    use="required" />
</xs:element>

<xs:element name="MEauth" type="hexbyte">
  <xs:attributeGroup ref="attrs.hex"
                    use="required" />
</xs:element>

<xs:element name="Pauth" type="hexbyte">
  <xs:attributeGroup ref="attrs.hex"
                    use="required" />
</xs:element>

<xs:element name="Hhash" type="hexbyte">
  <xs:attributeGroup ref="attrs.hex"
                    use="required" />
</xs:element>

<xs:element name="SIZhash" type="xs:integer">
  <xs:attributeGroup ref="attrs.int"
                    use="required" />
</xs:element>

<xs:element name="Mbc" type="hexbyte">
  <xs:attributeGroup ref="attrs.hex"
                    use="required" />
</xs:element>

<xs:element name="Pbc" type="hexbyte">
  <xs:attributeGroup ref="attrs.hex"
                    use="required" />
</xs:element>

<xs:element name="SIZbc" type="xs:integer">
<xs:attributeGroup ref="attrs.int"
                    use="required" />
</xs:element>

<xs:element name="KTntid" type="hexbyte">
  <xs:attributeGroup ref="attrs.hex"
                    use="required" />
</xs:element>

<xs:element name="Mhash" type="hexbyte">
  <xs:attributeGroup ref="attrs.hex"
                    use="required" />
</xs:element>

<xs:element name="Hhash" type="hexbyte">
  <xs:attributeGroup ref="attrs.hex"
                    use="required" />
</xs:element>

<xs:element name="SIZhash" type="hexbyte">
  <xs:attributeGroup ref="attrs.hex"
                    use="required" />
</xs:element>

<xs:element name="CAcmac" type="hexbyte">
  <xs:attributeGroup ref="attrs.hex"
                    use="required" />
</xs:element>

<xs:element name="Ccmac" type="hexbyte">
  <xs:attributeGroup ref="attrs.hex"
                    use="required" />
</xs:element>

<xs:element name="SIZcmac" type="xs:integer">
  <xs:attributeGroup ref="attrs.int"
                    use="required" />
</xs:element>
```

```
<xs:element name="Mds" type="hexbyte">
  <xs:attributeGroup ref="attrs.hex"
                     use="required" />
</xs:element>

<xs:element name="Hds" type="hexbyte">
  <xs:attributeGroup ref="attrs.hex"
                     use="required" />
</xs:element>

<xs:element name="SIZds" type="xs:integer">
  <xs:attributeGroup ref="attrs.int"
                     use="required" />
</xs:element>

<xs:element name="LKkt" type="hexbyte">
  <xs:attributeGroup ref="attrs.hex"
                     use="required" />
</xs:element>

<xs:element name="KIDkt" type="hexbyte">
  <xs:attributeGroup ref="attrs.hex"
                     use="required" />
</xs:element>

<xs:element name="granularity" type="hexbyte">
  <xs:attributeGroup ref="attrs.hex"
                     use="required" />
</xs:element>

<xs:element name="data" type="hexbyte">
  <xs:attributeGroup ref="attrs.hex"
                     use="required" />
</xs:element>
```

B.3.5 Example of an XML schema for a JPEG 2000 Part 10 codestream (3-D image)

```
<!-- part 1 marker element -->
…
<!-- part 10 marker element -->
<xs:element name="NSI">
  <xs:complexType>
    <xs:attributeGroup ref="attrs.marker"
                       use="required" />
    <xs:sequence>
      <xs:element ref="Ndim" />
      <xs:element ref="Zsiz" />
      <xs:element ref="ZOsiz" />
      <xs:element ref="ZTsiz" />
      <xs:element ref="ZTOsiz" />
      <xs:sequence maxOccurs="unbounded">
        <xs:element ref="ZRsiz" />
      </xs:sequence>
    </xs:sequence>
  </xs:complexType>
</xs:element>

<xs:element name="SPcod">
  <xs:complexType>
    <xs:attributeGroup ref="attrs.hex"
                       use="required" />
    <xs:sequence>
      <xs:element ref="num xlevels" />
      <xs:element ref="num_ylevels" />
      <xs:element ref="num zlevels" />
      <xs:element ref="xcb" />
      <xs:element ref="ycb" />
      <xs:element ref="zcb" />
      <xs:element ref="style" />
      <xs:element ref="xkernel" />
      <xs:element ref="ykernel" />
      <xs:element ref="zkernel" />
      <xs:element ref="sso" />
      <xs:element ref="reserved" />
      <xs:sequence maxOccurs="unbounded">
        <xs:element ref="ppz" />
        <xs:element ref="ppy" />
        <xs:element ref="ppx" />
      </xs:sequence>
    </xs:sequence>
  </xs:complexType>
</xs:element>

<xs:element name="SPcoc">
  <xs:complexType>
    <xs:attributeGroup ref="attrs.hex"
                       use="required" />
    <xs:sequence>
      <xs:element ref="num_xlevels" />
      <xs:element ref="num ylevels" />
      <xs:element ref="num zlevels" />
      <xs:element ref="xcb" />
      <xs:element ref="ycb" />
      <xs:element ref="zcb" />
      <xs:element ref="style" />
      <xs:element ref="xkernel" />
      <xs:element ref="ykernel" />
      <xs:element ref="zkernel" />
      <xs:element ref="sso" />
      <xs:element ref="reserved" />
      <xs:sequence maxOccurs="unbounded">
        <xs:element ref="ppz" />
        <xs:element ref="ppy" />
        <xs:element ref="ppx" />
      </xs:sequence>
    </xs:sequence>
  </xs:complexType>
</xs:element>

<xs:element name="SPrgn">
  <xs:complexType>
    <xs:attributeGroup ref="attrs.hex"
                       use="required" />
    <xs:sequence>
      <xs:element ref="shift" />
      <xs:element ref="XArgn" />
      <xs:element ref="YArgn" />
      <xs:element ref="ZArgn" />
      <xs:element ref="XBrgn" />
      <xs:element ref="YBrgn" />
      <xs:element ref="ZBrgn" />
    </xs:sequence>
  </xs:complexType>
</xs:element>

<xs:element name="CRG">
  <xs:complexType>
    <xs:attributeGroup ref="attrs.marker"
                       use="required" />
    <xs:sequence maxOccurs="unbounded">
      <xs:element ref="Xcrg" />
      <xs:element ref="Ycrg" />
      <xs:element ref="Zcrg" />
    </xs:sequence>
  </xs:complexType>
</xs:element>

<!-- part 1 content element -->
…
<!-- part 10 content element -->
<xs:element name="Ndim" type="xs:integer">
  <xs:attributeGroup ref="attrs.int"
                     use="required" />
</xs:element>

<xs:element name="Zsiz" type="xs:integer">
  <xs:attributeGroup ref="attrs.int"
                     use="required" />
</xs:element>

<xs:element name="ZOsiz" type="xs:integer">
  <xs:attributeGroup ref="attrs.int"
                     use="required" />
</xs:element>

<xs:element name="ZTsiz" type="xs:integer">
  <xs:attributeGroup ref="attrs.int"
                     use="required" />
</xs:element>

<xs:element name="ZTOsiz" type="xs:integer">
  <xs:attributeGroup ref="attrs.int"
                     use="required" />
</xs:element>

<xs:element name="ZRsiz" type="xs:integer">
```

```
  <xs:attributeGroup ref="attrs.int"
                     use="required" />
</xs:element>

<xs:element name="num_xlevels"
type="xs:integer">
  <xs:attributeGroup ref="attrs.int"
                     use="required" />
</xs:element>

<xs:element name="num ylevels"
type="xs:integer">
  <xs:attributeGroup ref="attrs.int"
                     use="required" />
</xs:element>

<xs:element name="num_zlevels"
type="xs:integer">
  <xs:attributeGroup ref="attrs.int"
                     use="required" />
</xs:element>

<xs:element name="ycb" type="xs:integer">
  <xs:attributeGroup ref="attrs.int"
                     use="required" />
</xs:element>

<xs:element name="zcb" type="xs:integer">
  <xs:attributeGroup ref="attrs.int"
                     use="required" />
</xs:element>
```

```
<xs:element name="zkernel" type="xs:integer">
  <xs:attributeGroup ref="attrs.int"
                     use="required" />
</xs:element>

<xs:element name="reserved" type="xs:integer">
  <xs:attributeGroup ref="attrs.int"
                     use="required" />
</xs:element>

<xs:element name="ppz" type="xs:integer">
  <xs:attributeGroup ref="attrs.int"
                     use="required" />
</xs:element>

<xs:element name="ZArgn" type="xs:integer">
  <xs:attributeGroup ref="attrs.int"
                     use="required" />
</xs:element>

<xs:element name="ZBrgn" type="xs:integer">
  <xs:attributeGroup ref="attrs.int"
                     use="required" />
</xs:element>

<xs:element name="Zcrg" type="xs:integer">
  <xs:attributeGroup ref="attrs.int"
                     use="required" />
</xs:element>
```

B.4.6 Example of an XML schema for a JPEG 2000 Part 11 codestream (wireless)

```
<!-- part 1 marker element -->
…
<!-- part 11 marker element -->
<xs:element name="EPB">
  <xs:complexType>
    <xs:attributeGroup ref="attrs.marker"
                       use="required" />
    <xs:sequence>
      <xs:element ref="Depb" />
      <xs:element ref="LDPepb" />
      <xs:element ref="Pepb" />
      <xs:element ref="data" />
    </xs:sequence>
  </xs:complexType>
</xs:element>

<xs:element name="EPC">
  <xs:complexType>
    <xs:attributeGroup ref="attrs.marker"
                       use="required" />
    <xs:sequence>
      <xs:element ref="Pcrc" />
      <xs:element ref="DL" />
      <xs:element ref="Pepc" />
      <xs:sequence maxOccurs="unbounded">
        <xs:element ref="id" />
        <xs:element ref="Lid" />
        <xs:element ref="Pid" />
      </xs:sequence>
    </xs:sequence>
  </xs:complexType>
</xs:element>

<xs:element name="ESD">
  <xs:complexType>
    <xs:attributeGroup ref="attrs.marker"
                       use="required" />
    <xs:sequence>
      <xs:element ref="Cesd" />
      <xs:element ref="Pesd" />
      <xs:element ref="data" />
    </xs:sequence>
  </xs:complexType>
</xs:element>

<xs:element name="RED">
  <xs:complexType>
    <xs:attributeGroup ref="attrs.marker"
                       use="required" />
    <xs:sequence>
      <xs:element ref="Pred" />
```

```
      <xs:element ref="data" />
    </xs:sequence>
  </xs:complexType>
</xs:element>

<!-- part 1 content element -->
…
<!-- part 11 content element -->
<xs:element ref="Depb" type="xs:integer">
  <xs:attributeGroup ref="attrs.int"
                     use="required" />
</xs:element>

<xs:element ref="LDPepb" type="xs:integer">
  <xs:attributeGroup ref="attrs.int"
                     use="required" />
</xs:element>

<xs:element ref="Pepb" type="xs:integer">
  <xs:attributeGroup ref="attrs.int"
                     use="required" />
</xs:element>

<xs:element ref="Pcrc" type="xs:integer">
  <xs:attributeGroup ref="attrs.int"
                     use="required" />
</xs:element>

<xs:element ref="DL" type="xs:integer">
  <xs:attributeGroup ref="attrs.int"
                     use="required" />
</xs:element>

<xs:element ref="Pepc" type="xs:integer">
  <xs:attributeGroup ref="attrs.int"
                     use="required" />
</xs:element>

<xs:element ref="id" type="hexbyte">
  <xs:attributeGroup ref="attrs.hex"
                     use="required" />
</xs:element>

<xs:element ref="Lid" type="hexbyte">
  <xs:attributeGroup ref="attrs.hex"
                     use="required" />
</xs:element>

<xs:element ref="Pid" type="hexbyte">
  <xs:attributeGroup ref="attrs.hex"
                     use="required" />
```

```
</xs:element>

<xs:element ref="Cesd" type="xs:integer">
  <xs:attributeGroup ref="attrs.int"
                     use="required" />
</xs:element>

<xs:element ref="Pesd" type="xs:integer">
  <xs:attributeGroup ref="attrs.int"
                     use="required" />
</xs:element>
```

```
<xs:element ref="Pred" type="xs:integer">
  <xs:attributeGroup ref="attrs.int"
                     use="required" />
</xs:element>

<xs:element ref="data" type="hexbyte">
  <xs:attributeGroup ref="attrs.hex"
                     use="required" />
</xs:element>
```

Annex C

Examples and guidelines

(This annex does not form an integral part of this Recommendation | International Standard.)

This annex includes a number of examples intended to indicate how the encoding process works, and how the resulting codestream needs to be output. This annex is entirely informative.

C.1 Software conventions for the box type

This provides some examples of implementation for box type conversion to a JPXML element name with python, which include no error checking. This alternative version may be more efficient when implemented in software as it has fewer operations along the fast path, and may have error checking mechanisms for a robust application.

C.1.1 Conventions to JPXML element name from a box type

```
def convert_4cc_to_xml(code):
  p = 0
  str = ""
  while p < len(code):
    ch = code[p]
    if ch == " ":
      ch = "_"
    elif ch == "\\" and code[p+1:p+4].isdigit():
      val = int(code[p+1:p+4], 8)
      if val == 32:
        ch = "_"
      else:
        ch = "." + ("0" + hex(val)[2:])[-2:]
      p += 3
    elif not ch.isalnum():
      ch = "." + ("0" + hex(ord(ch))[2:])[-2:]
    p += 1
    str += ch
  if str[0] == "_" or str[0] == "." or str[0].isdigit() or str[:3] == "xml":
    str = "_" + str
  return str
```

Figure C.1 – Python source code for converting a box type to a JPXML element name

C.1.2 Conventions from JPXML element name to a box type

```
def convert_xml_to_4cc(code):
  if code[0] == "_":
    code = code[1:]
  p = 0
  str = ""
  while p < len(code):
    ch = code[p]
    if ch == "_":
      ch = "\\040"
    elif ch==".":
      ch = "\\" + ("0" + oct(int(code[p+1:p+3], 16)))[-3:]
      p += 2
    p += 1
    str += ch
  return str
```

Figure C.2 – Python source code for converting a JPXML element name to a box type

C.2 Example of JPXML document conversion

This JPEG 2000 image example consists of six boxes, and the "jp2h" superbox contains "ihdr" and "colr" boxes. Each box has an LBox value, so the converter will use it as the length attribute value in each box element. The last box may have zero length representations that content data extends to the end of a file.

```
Box[0] Type: 'jP\040\040', offset=0x00000, length=12.
   <JPEG2000 Signature box : d0a870a>.

Box[1] Type: 'ftyp', offset=0x0000c, length=20.
   <File Type box>.
   Brand='jp2 ', Minor Version=0.
   Compatibility: 'jp2 '

Box[2] Type: 'jp2h', offset=0x00020, length=45
|     <JP2 Header box [JP2]> <superbox>
| Box[3] Type: 'ihdr', offset=0x00028, length=22
|     <Image Header box>
|     Image size=(400, 300), Depth=8, Number of components=3,
|     Compression type(7)=ITU-T T.800 (JPEG2000),
|     Colour type(1)=unknown
|
| Box[4] Type: 'colr', offset=0x0003e, length=15
|     <Colour Specification box>
|     Method(1)= Enumerated Colour space
|     Prec=0, Approx=0, Colour(16 )=sRGB
|
Box[5] Type: 'jp2c', offset=0x0004d, length=0
   <Contiguous Codestream box><stream[0]>
   Codestream offset=85 (0x00055), Length=196439 (0x2ff57)
```

Figure C.3 – Example of an input image for this JPXML representation process

C.2.1 JPXML structural document

The "jP\040\040" box type is converted to a "jP__" element name, and other 4CC box types are used for the element names. If there is an XML container box in an image, an element name needs to be "_xml_". The root element of the JPXML document, "jpBox", needs to have a file attribute for identifying the file name of this document. Each box needs to have a length attribute and a location reference to a box itself in an image, and the location reference needs to be represented with an XML structure.

For the purpose of representing image structure, a JPXML document without box contents may be used. This document may have base64 binary converted contents for some purposes. In many cases, this document may have a length attribute for calculating an absolute offset location from a structural relative location address with XML form.

```
<?xml version="1.0" ?>
<jx:jpxml  xl="http://www.w3.org/1999/xlink"
           jx="http://www.iso.org/jpeg/jpxml/1.0"
           jx:file="small-lena.jp2">
   <jx:jP__ length="12" type="box:box" />
   <jx:ftyp length="20" type="box:box" />
   <jx:jp2h length="45" type="box:box" >
      <jx:ihdr length="22" type="box:box" />
      <jx:colr length="15" type="box:box" />
   </jx:jp2h>
   <jx:jp2c length="0" xl:href="#xpointer(box:jp2c)"/>
</jx:jpxml>
```

Figure C.4 – An example of a JPXML document without box contents

The JPXML document with box content elements may be used for image conversion, image editing, image security, etc. In many cases, this document needs to have a length attribute for calculating an absolute offset location from a structural relative location address with XML form. The length attribute in the box needs to be changed correctly when box contents are added or changed.

```
<?xml version="1.0" ?>
<jx:jpxml xl="http://www.w3.org/1999/xlink" xs="http://www.w3.org/2001/XMLSchema"
          jx="http://www.jpeg.org/jpxml/1.0" jx:file="lena.jp2">
   <jx:jP__ length="12" type="jx:box">
      < jx:signature length="4" type="xs:hexBinary">0x0d0a870a</jx:signature>
   </jx:jP__>
   <jx:ftyp length="20" type="jx:box">
      <jx:brand length="4" type="xs:string">jp2 </jx:brand>
      <jx:version length="4" type="xs:int">0</jx:version>
      <jx:compatibility length="4" type="xs:string">jp2 </jx:compatibility>
   </jx:ftyp>
   <jx:jp2h length="45" type="jx:box">
      <jx:ihdr length="22" type="jx:box">
         <jx:width length="4" type="xs:int">512</jx:width>
         <jx:height length="4" type="xs:int">512</jx:height>
         <jx:depth length="2" type="xs:int">8</jx:depth>
         <jx:num_components length="1" type="xs:int">3</jx:num_components>
         <jx:coding length="1" type="xs:int">7</jx:coding><!--ITU-T T.800-->
         <jx:colour length="1" type="xs:int">1</jx:colour><!--unknown-->
         <jx:ipr length="1" type="xs:boolean">false</jx:ipr>
      </jx:ihdr>
      <jx:colr length="15" type="jx:box">
         <jx:method length="1" type="xs:int">1</jx:method><!--Enumerated colour--
     >
         <jx:precision length="1" type="xs:int">0</jx:precision>
         <jx:approx length="1" type="xs:int">0</jx:approx>
         <jx:colour length="4" type="xs:int">16</jx:colour><!--sRGB-->
      </jx:colr >
   </jx:jp2h>
   <jx:jp2c length="0" xl:href="#xpointer(jx:jp2c)" />
</jx:jpxml>
```

Figure C.5 – An example of a JPXML document with box content child elements

C.2.2 Reference to internal data in an image

The relative location generator converts an absolute offset location to a structural location path which is expressed by XML location tools; XPath, XLink and XPointer. The absolute location generator conducts reverse order of the relative location conversion process. Figure 7 illustrates converting processes between absolute and relative locations. These generators need the schemas to calculate each location when an input from these generators has no box length information, because the generator uses box length values for identifying a box element location in the XML domain. The process of generating a relative location consists of two steps: 1) repeat reading a box element's `length` attribute and subtracting its length value from an offset value whenever the offset is greater than the reading of a current box length, and 2) describe an XML location of the extracted element by using XML location tools, e.g., the offset "54" in Figure 3 is the location to the element of '`<ihdr />`'; Image Header box ('`ihdr`'), so the location of offset "54" is converted to '`xpointer(//ihdr)`' or '`xpointer(/jpBox/ihdr)`'.

SERIES OF ITU-T RECOMMENDATIONS

Series A	Organization of the work of ITU-T
Series D	General tariff principles
Series E	Overall network operation, telephone service, service operation and human factors
Series F	Non-telephone telecommunication services
Series G	Transmission systems and media, digital systems and networks
Series H	Audiovisual and multimedia systems
Series I	Integrated services digital network
Series J	Cable networks and transmission of television, sound programme and other multimedia signals
Series K	Protection against interference
Series L	Construction, installation and protection of cables and other elements of outside plant
Series M	Telecommunication management, including TMN and network maintenance
Series N	Maintenance: international sound programme and television transmission circuits
Series O	Specifications of measuring equipment
Series P	Terminals and subjective and objective assessment methods
Series Q	Switching and signalling
Series R	Telegraph transmission
Series S	Telegraph services terminal equipment
Series T	**Terminals for telematic services**
Series U	Telegraph switching
Series V	Data communication over the telephone network
Series X	Data networks, open system communications and security
Series Y	Global information infrastructure, Internet protocol aspects and next-generation networks
Series Z	Languages and general software aspects for telecommunication systems

Printed in Switzerland
Geneva, 2013