avtcore   S. Lugan
Internet-Draft   intoPIX
Intended status: Standards Track   A. Descampe
Expires: January 29, 2022   UCL
                                                        C. Damman
                                                           intoPIX
                                                        T. Richter
                                                               IIS
                                                      T. Bruylants
                                                           intoPIX
                                                    July 28, 2021

RTP Payload Format for ISO/IEC 21122 (JPEG XS)
draft-ietf-payload-rtp-jpegxs-18

Abstract

   This document specifies a Real-Time Transport Protocol (RTP) payload
   format to be used for transporting JPEG XS (ISO/IEC 21122) encoded
   video. JPEG XS is a low-latency, lightweight image coding system.
   Compared to an uncompressed video use case, it allows higher
   resolutions and video frame rates, while offering visually lossless
   quality, reduced power consumption, and encoding-decoding latency
   confined to a fraction of a video frame.

Status of This Memo

   This Internet-Draft is submitted in full conformance with the
   provisions of BCP 78 and BCP 79.

   Internet-Drafts are working documents of the Internet Engineering
   Task Force (IETF). Note that other groups may also distribute
   working documents as Internet-Drafts. The list of current Internet-
   Drafts is at https://datatracker.ietf.org/drafts/current/.

   Internet-Drafts are draft documents valid for a maximum of six months
   and may be updated, replaced, or obsoleted by other documents at any
   time. It is inappropriate to use Internet-Drafts as reference
   material or to cite them other than as "work in progress."

   This Internet-Draft will expire on January 29, 2022.

Copyright Notice

Table of Contents

1. Introduction

   This document specifies a payload format for packetization of JPEG XS
   [ISO21122-1] encoded video signals into the Real-time Transport
   Protocol (RTP) [RFC3550].

   The JPEG XS coding system offers compression and recompression of
   image sequences with very moderate computational resources while

remaining robust under multiple compression and decompression cycles
and mixing of content sources, e.g. embedding of subtitles, overlays
or logos. Typical target compression ratios ensuring visually
lossless quality are in the range of 2:1 to 10:1, depending on the
nature of the source material. The latency that is introduced by the
encoding-decoding process can be confined to a fraction of a video
frame, typically between a small number of lines down to below a
single line.

2. Conventions, Definitions, and Abbreviations

The key words "MUST", "MUST NOT", "REQUIRED", "SHALL", "SHALL NOT",
"SHOULD", "SHOULD NOT", "RECOMMENDED", "NOT RECOMMENDED", "MAY", and
"OPTIONAL" in this document are to be interpreted as described in BCP
14 [RFC2119] [RFC8174] when, and only when, they appear in all
capitals, as shown here.

Application Data Unit (ADU)
   The unit of source data provided as payload to the transport
   layer, and corresponding, in this RTP payload definition, to a
   single JPEG XS video frame.

Color specification box (CS box)
   An ISO color specification box defined in JPEG XS Part 3
   [ISO21122-3] that includes color related metadata required to
   correctly display JPEG XS video frames, such as color primaries,
   transfer characteristics and matrix coefficients.

EOC marker
   A marker that consists of the two bytes 0xff11 indicating the end
   of a JPEG XS codestream.

JPEG XS codestream
   A sequence of bytes representing a compressed image formatted
   according to JPEG XS Part 1 [ISO21122-1].

JPEG XS codestream header
   A sequence of bytes, starting with a SOC marker, at the beginning
   of each JPEG XS codestream encoded in multiple markers and marker
   segments that does not carry entropy coded data, but metadata such
   as the video frame dimension and component precision.

JPEG XS frame
   In the case of progressive video, a single JPEG XS picture
   segment. In the case of interlaced video, the concatenation of
   two JPEG XS picture segments.

JPEG XS header segment

   The concatenation of a video support box [ISO21122-3], a color
   specification box [ISO21122-3], and a JPEG XS codestream header.

JPEG XS picture segment
   The concatenation of a video support box [ISO21122-3], a color
   specification box [ISO21122-3], and a JPEG XS codestream.

JPEG XS stream
   A sequence of JPEG XS frames.

Marker
   A two-byte functional sequence that is part of a JPEG XS
   codestream starting with a 0xff byte and a subsequent byte
   defining its function.

Marker segment
   A marker along with a 16-bit marker size and payload data
   following the size.

Packetization unit
   A portion of an Application Data Unit whose boundaries coincide
   with boundaries of RTP packet payloads (excluding payload header),
   i.e. the first (resp. last) byte of a packetization unit is the
   first (resp. last) byte of an RTP packet payload (excluding its
   payload header).

SLH marker
   A marker that represents a slice header, as defined in
   [ISO21122-1].

Slice
   The smallest independently decodable unit of a JPEG XS codestream,
   bearing in mind that it decodes to wavelet coefficients which
   still require inverse wavelet filtering to give an image.

SOC marker
   A marker that consists of the two bytes 0xff10 indicating the
   start of a JPEG XS codestream. The SOC marker is considered an
   integral part of the JPEG XS codestream header.

Video support box (VS box)
   An ISO video support box, as defined in [ISO21122-3], that
   includes metadata required to play back a JPEG XS stream, such as
   its maximum bitrate, its subsampling structure, its buffer model
   and its frame rate.

3. Media Format Description

   This section explains the terminology and concepts used in this memo
   that are specific to JPEG XS as specified in [ISO21122-1],
   [ISO21122-2], and [ISO21122-3].

3.1. Image Data Structures

   JPEG XS is a low-latency lightweight image coding system for coding
   continuous-tone grayscale or continuous-tone color digital images.

   This coding system provides an efficient representation of image
   signals through the mathematical tool of wavelet analysis. The
   wavelet filter process separates each component into multiple bands,
   where each band consists of multiple coefficients describing the
   image signal of a given component within a frequency domain specific
   to the wavelet filter type, i.e. the particular filter corresponding
   to the band.

   Wavelet coefficients are grouped into precincts, where each precinct
   includes all coefficients over all bands that contribute to a spatial
   region of the image.

   One or multiple precincts are furthermore combined into slices
   consisting of an integer number of precincts. Precincts do not cross
   slice boundaries, and wavelet coefficients in precincts that are part
   of different slices can be decoded independently of each other.
   Note, however, that the wavelet transformation runs across slice
   boundaries. A slice always extends over the full width of the image,
   but may only cover parts of its height.

3.2. Codestream

   A JPEG XS codestream is formed by (in the given order):

   o a JPEG XS codestream header, which starts with an SOC marker,

   o one or more slices,

   o an EOC marker to signal the end of the codestream.

   The JPEG XS codestream format, including the definition of all
   markers, is further defined in [ISO21122-1]. It represents sample
   values of a single image, without any interpretation relative to a
   color space.

3.3. Video support box and color specification box

While the information defined in the codestream is sufficient to
reconstruct the sample values of one image, the interpretation of the
samples remains undefined by the codestream itself. This
interpretation is given by the video support box and the color
specification box which contain significant information to correctly
play the JPEG XS stream. The layout and syntax of these boxes,
together with their content, are defined in [ISO21122-3].

The video support box provides information on the maximum bitrate,
the frame rate, the interlaced mode (progressive or interlaced), the
subsampling image format, the informative timecode of the current
JPEG XS frame, the profile, level/sublevel used, and optionally on
the buffer model and the mastering display metadata.

Note that the profile and level/sublevel, specified by respectively
the Ppih and Plev fields [ISO21122-2], specify limits on the
capabilities needed to decode the codestream and handle the output.
Profiles represent a limit on the required algorithmic features and
parameter ranges used in the codestream. The combination of level
and sublevel defines a lower bound on the required throughput for a
decoder in respectively the image (or decoded) domain and the
codestream (or coded) domain. The actual defined profiles and
levels/sublevels, along with the associated values for the Ppih and
Plev fields, are defined in [ISO21122-2].

The color specification box indicates the color primaries, transfer
characteristics, matrix coefficients and video full range flag needed
to specify the color space of the video stream.

3.4. JPEG XS Frame

The concatenation of a video support box, a color specification box,
and a JPEG XS codestream forms a JPEG XS picture segment.

In the case of a progressive video stream, each JPEG XS frame
consists of one single JPEG XS picture segment.

In the case of an interlaced video stream, each JPEG XS frame is made
of two concatenated JPEG XS picture segments. The codestream of each
picture segment corresponds exclusively to one of the two fields of
the interlaced frame. Both picture segments SHALL contain identical
boxes (i.e. concatenation of the video support box and the color
specification box is byte exact the same for both picture segments of
the frame).

Note that the interlaced mode, as signaled by the frat field
[ISO21122-3] in the video support box, indicates either progressive,
interlaced top-field first, or interlaced bottom-field first mode.
Thus, in the case of interlaced content, its value SHALL also be
identical in both picture segments.

4. RTP Payload Format

This section specifies the payload format for JPEG XS streams over
the Real-time Transport Protocol (RTP) [RFC3550].

In order to be transported over RTP, each JPEG XS stream is
transported in a distinct RTP stream, identified by a distinct
Synchronization source (SSRC) [RFC3550].

A JPEG XS stream is divided into Application Data Units (ADUs), each
ADU corresponding to a single JPEG XS frame.

4.1. RTP packetization

An ADU is made of several packetization units. If a packetization
unit is bigger than the maximum size of an RTP packet payload, the
unit is split into multiple RTP packet payloads, as illustrated in
Figure 1. As seen there, each packet SHALL contain (part of) one and
only one packetization unit. A packetization unit may extend over
multiple packets. The payload of every packet SHALL have the same
size (based e.g. on the Maximum Transfer Unit of the network), except
(possibly) the last packet of a packetization unit. The boundaries
of a packetization unit SHALL coincide with the boundaries of the
payload of a packet (excluding the payload header), i.e. the first
(resp. last) byte of the packetization unit SHALL be the first (resp.
last) byte of the payload (excluding its header).

```
RTP          +-----+------------------------+
Packet #1  | Hdr | Packetization unit #1  |
             +-----+------------------------+
RTP          +-----+-------------------------------------+
Packet #2  | Hdr | Packetization unit #2               |
             +-----+-------------------------------------+
RTP          +-----+--------------------------------------------------+
Packet #3  | Hdr | Packetization unit #3  (part 1/3)                |
             +-----+--------------------------------------------------+
RTP          +-----+--------------------------------------------------+
Packet #4  | Hdr | Packetization unit #3  (part 2/3)                |
             +-----+--------------------------------------------------+
RTP          +-----+-----------------------------------------------+
Packet #5  | Hdr | Packetization unit #3  (part 3/3)             |
             +-----+-----------------------------------------------+
             ...
RTP          +-----+-----------------------------------------+
Packet #P  | Hdr | Packetization unit #N  (part q/q)       |
             +-----+-----------------------------------------+
```
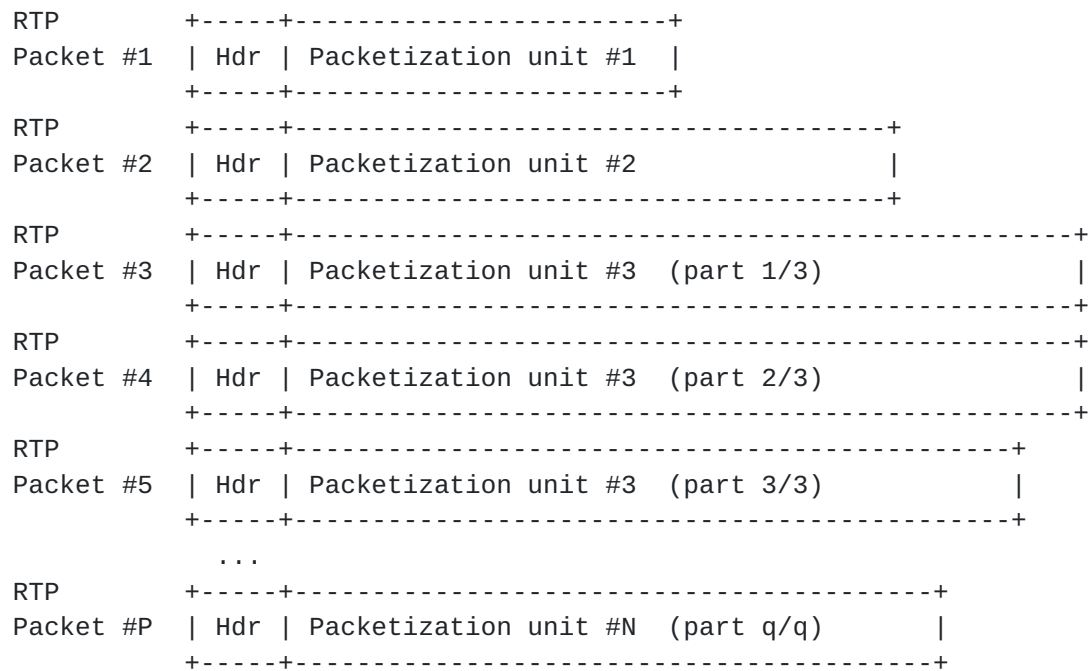
Figure 1: Example of ADU packetization

There are two different packetization modes defined for this RTP
payload format.

1. Codestream packetization mode: in this mode, the packetization
    unit SHALL be the entire JPEG XS picture segment (i.e. codestream
    preceded by boxes). This means that a progressive frame will
    have a single packetization unit, while an interlaced frame will
    have two. The progressive case is illustrated in Figure 2.

2. Slice packetization mode: in this mode, the packetization unit
    SHALL be the slice, i.e. there SHALL be data from no more than
    one slice per RTP packet. The first packetization unit SHALL be
    made of the JPEG XS header segment (i.e. the concatenation of the
    VS box, the CS box and the JPEG XS codestream header). This
    first unit is then followed by successive units, each containing
    one and only one slice. The packetization unit containing the
    last slice of a JPEG XS codestream SHALL also contain the EOC
    marker immediately following this last slice. This is
    illustrated in Figure 3. In the case of an interlaced frame, the
    JPEG XS header segment of the second field SHALL be in its own
    packetization unit.

```
RTP         +-----+-------------------------------------------------+
Packet #1  | Hdr | VS box + CS box + JPEG XS codestream (part 1/q)  |
            +-----+-------------------------------------------------+
RTP         +-----+-------------------------------------------------+
Packet #2  | Hdr | JPEG XS codestream (part 2/q)                    |
            +-----+-------------------------------------------------+
              ...
RTP         +-----+---------------------------------------+
Packet #P  | Hdr | JPEG XS codestream (part q/q)         |
            +-----+---------------------------------------+
```
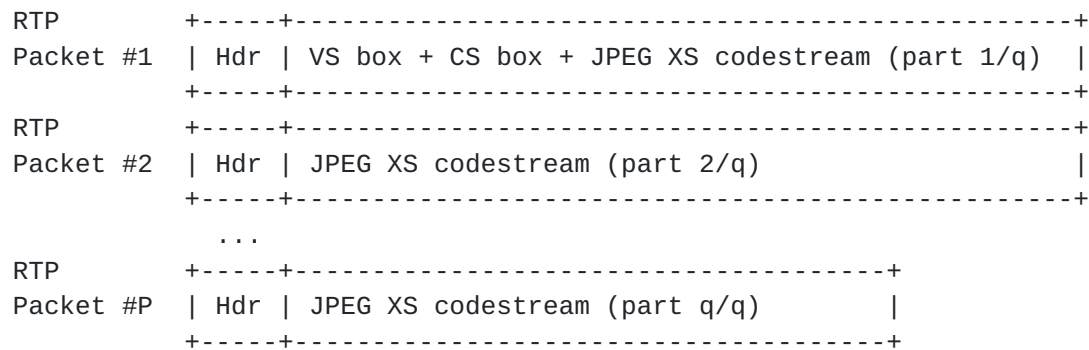
Figure 2: Example of codestream packetization mode

```
RTP         +-----+----------------------------+
Packet #1  | Hdr | JPEG XS header segment      |
            +-----+----------------------------+
RTP         +-----+-------------------------------------------------+
Packet #2  | Hdr | Slice #1  (part 1/2)                             |
            +-----+-------------------------------------------------+
RTP         +-----+---------------------------------------------+
Packet #3  | Hdr | Slice #1  (part 2/2)                         |
            +-----+---------------------------------------------+
RTP         +-----+-------------------------------------------------+
Packet #4  | Hdr | Slice #2  (part 1/3)                             |
            +-----+-------------------------------------------------+
              ...
RTP         +-----+---------------------------------------+
Packet #P  | Hdr | Slice #N  (part q/q) + EOC marker     |
            +-----+---------------------------------------+
```
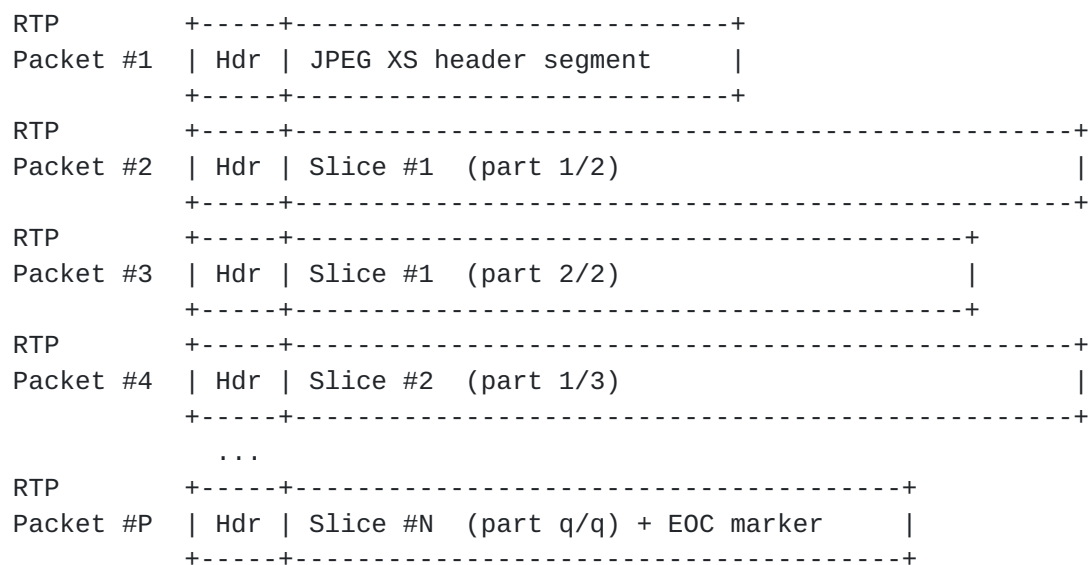
Figure 3: Example of slice packetization mode

In a constant bit-rate (CBR) scenario of JPEG XS, the codestream
packetization mode guarantees that a JPEG XS RTP stream will produce
a constant number of bytes per video frame, and a constant number of
RTP packets per video frame. However, to provide similar guarantees
with JPEG XS in a variable bit-rate (VBR) mode or when using the
slice packetization mode (for either CBR or VBR), additional
mechanisms are needed. This can involve a constraint at the rate
allocation stage in the JPEG XS encoder to impose a constant bit-rate
at the slice level, the usage of padding data, or the insertion of
empty RTP packets (i.e. an RTP packet whose payload data is empty).
But, management of the amount of produced packets per video frame is
application dependent and not a strict requirement of this RTP
payload specification.

4.2. **RTP Header Usage**

The format of the RTP header is specified in [RFC3550] and reprinted
in Figure 4 for convenience. This RTP payload format uses the fields
of the header in a manner consistent with that specification.

The RTP payload (and the settings for some RTP header bits) for
packetization units are specified in Section 4.3.

```
    0                   1                   2                   3
    0 1 2 3 4 5 6 7 8 9 0 1 2 3 4 5 6 7 8 9 0 1 2 3 4 5 6 7 8 9 0 1
   +-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
   | V |P|X|  CC   |M|     PT      |       sequence number         |
   +-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
   |                           timestamp                           |
   +-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
   |           synchronization source (SSRC) identifier           |
   +=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+
   |             contributing source (CSRC) identifiers           |
   |                             ....                              |
   +-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
```
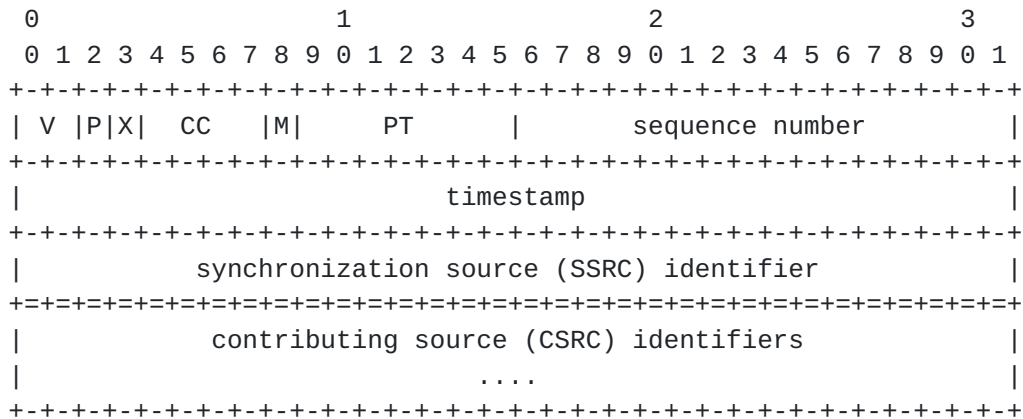
Figure 4: RTP header according to RFC 3550

The version (V), padding (P), extension (X), CSRC count (CC),
sequence number, synchronization source (SSRC) and contributing
source (CSRC) fields follow their respective definitions in
[RFC3550].

The remaining RTP header information to be set according to this RTP
payload format is set as follows:

Marker (M) [1 bit]:

   If progressive scan video is being transmitted, the marker bit
   denotes the end of a video frame. If interlaced video is being
   transmitted, it denotes the end of the field. The marker bit
   SHALL be set to 1 for the last packet of the video frame/field.
   It SHALL be set to 0 for all other packets.

Payload Type (PT) [7 bits]:

   A dynamically allocated payload type field that designates the
   payload as JPEG XS video.

Timestamp [32 bits]:

The RTP timestamp is set to the sampling timestamp of the content.
A 90 kHz clock rate SHALL be used.

As specified in [RFC3550] and [RFC4175], the RTP timestamp
designates the sampling instant of the first octet of the video
frame to which the RTP packet belongs. Packets SHALL NOT include
data from multiple video frames, and all packets belonging to the
same video frame SHALL have the same timestamp. Several
successive RTP packets will consequently have equal timestamps if
they belong to the same video frame (that is until the marker bit
is set to 1, marking the last packet of the video frame), and the
timestamp is only increased when a new video frame begins.

If the sampling instant does not correspond to an integer value of
the clock, the value SHALL be truncated to the next lowest
integer, with no ambiguity.

4.3. Payload Header Usage

The first four bytes of the payload of an RTP packet in this RTP
payload format are referred to as the payload header. Figure 5
illustrates the structure of this payload header.

```
 0                   1                   2                   3
 0 1 2 3 4 5 6 7 8 9 0 1 2 3 4 5 6 7 8 9 0 1 2 3 4 5 6 7 8 9 0 1
+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
|T|K|L| I |F counter|     SEP counter      |     P counter      |
+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
```
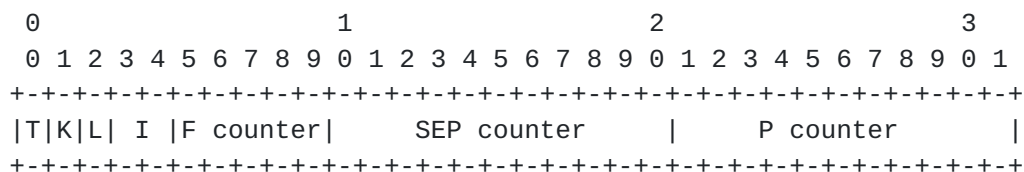
Figure 5: Payload header

The payload header consists of the following fields:

Transmission mode (T) [1 bit]:

The T bit is set to indicate that packets are sent sequentially by
the transmitter. This information allows a receiver to dimension
its input buffer(s) accordingly. If T=0, nothing can be assumed
about the transmission order and packets may be sent out-of-order
by the transmitter. If T=1, packets SHALL be sent sequentially by
the transmitter. The T bit value SHALL be identical for all
packets of the RTP stream.

pacKetization mode (K) [1 bit]:

The K bit is set to indicate which packetization mode is used.
K=0 indicates codestream packetization mode, while K=1 indicates
slice packetization mode. In the case that the Transmission mode

(T) is set to 0 (out-of-order), the slice packetization mode SHALL
be used and K SHALL be set to 1. This is required, because only
the slice packetization mode supports out-of-order packet
transmission. The K bit value SHALL be identical for all packets
of the RTP stream.

Last (L) [1 bit]:

   The L bit is set to indicate the last packet of a packetization
   unit. As the end of the video frame also ends the packet
   containing the last unit of the video frame, the L bit is set
   whenever the M bit is set. In the codestream packetization mode
   the L bit and M bit get an equivalent meaning, so they SHALL have
   identical values in each packet.

Interlaced information (I) [2 bit]:

   These two I bits are used to indicate how the JPEG XS frame is
   scanned (progressive or interlaced). In case of an interlaced
   frame, they also indicate which JPEG XS picture segment the
   payload is part of (first or second).

   00: The payload is progressively scanned.

   01: Reserved for future use.

   10: The payload is part of the first JPEG XS picture segment of
        an interlaced video frame. The height specified in the
        included JPEG XS codestream header is half of the height of the
        entire displayed image.

   11: The payload is part of the second JPEG XS picture segment of
        an interlaced video frame. The height specified in the
        included JPEG XS codestream header is half of the height of the
        entire displayed image.

 F counter [5 bits]:

   The frame (F) counter identifies the video frame number modulo 32
   to which a packet belongs. Frame numbers are incremented by 1 for
   each video frame transmitted. The frame number, in addition to
   the timestamp, may help the decoder manage its input buffer and
   bring packets back into their natural order.

 SEP counter [11 bits]:

   The Slice and Extended Packet (SEP) counter is used differently
   depending on the packetization mode.

* In the case of codestream packetization mode (K=0), this
  counter resets whenever the Packet counter resets (see
  Section 4.4), and increments by 1 whenever the Packet counter
  overruns.

* In the case of slice packetization mode (K=1), this counter
  identifies the slice modulo 2047 to which the packet
  contributes. If the data belongs to the JPEG XS header
  segment, this field SHALL have its maximal value, namely 2047 =
  0x07ff. Otherwise, it is the slice index modulo 2047. Slice
  indices are counted from 0 (corresponding to the top of the
  video frame).

P counter [11 bits]:

   The packet (P) counter identifies the packet number modulo 2048
   within the current packetization unit. It is set to 0 at the
   start of the packetization unit and incremented by 1 for every
   subsequent packet (if any) belonging to the same unit.
   Practically, if codestream packetization mode is enabled, this
   field counts the packets within a JPEG XS picture segment and is
   extended by the SEP counter when it overruns. If slice
   packetization mode is enabled, this field counts the packets
   within a slice or within the JPEG XS header segment.

4.4. Payload Data

   The payload data of a JPEG XS RTP stream consists of a concatenation
   of multiple JPEG XS frames. Within the RTP stream, all of the video
   support boxes and all of the color specification boxes SHALL retain
   their respective layouts for each JPEG XS frame. Thus, each video
   support box in the RTP stream SHALL define the same sub boxes. The
   effective values in the boxes are allowed to change under the
   condition that their relative byte offsets SHALL NOT change.

   Each JPEG XS frame is the concatenation of one or more packetization
   unit(s), as explained in Section 4.1. Figure 6 depicts this layout
   for a progressive video frame in the codestream packetization mode,
   Figure 7 depicts this layout for an interlaced video frame in the
   codestream packetization mode, Figure 8 depicts this layout for a
   progressive video frame in the slice packetization mode and Figure 9
   depicts this layout for an interlaced video frame in the slice
   packetization mode. The Frame counter value is not indicated because
   the value is constant for all packetization units of a given video
   frame.

```
+=====[ Packetization unit (PU) #1 ]====+
|             Video support box          |  SEP counter=0
|   +--------------------------------+   |  P counter=0
|   :      Sub boxes of the VS box   :   |
|   +--------------------------------+   |
+- - - - - - - - - - - - - - - - - - -+
|          Color specification box       |
|   +--------------------------------+   |
|   :        Fields of the CS box    :   |
|   +--------------------------------+   |
+- - - - - - - - - - - - - - - - - - -+
|              JPEG XS codestream        |
:                 (part 1/q)             :  M=0, K=0, L=0, I=00
+---------------------------------------+
|              JPEG XS codestream        |  SEP counter=0
|                 (part 2/q)             |  P counter=1
:                                        :  M=0, K=0, L=0, I=00
+---------------------------------------+
|              JPEG XS codestream        |  SEP counter=0
|                 (part 3/q)             |  P counter=2
:                                        :  M=0, K=0, L=0, I=00
+---------------------------------------+
:                                        :
+---------------------------------------+
|              JPEG XS codestream        |  SEP counter=1
|                (part 2049/q)           |  P counter=0
:                                        :  M=0, K=0, L=0, I=00
+---------------------------------------+
:                                        :
+---------------------------------------+
|              JPEG XS codestream        |  SEP counter=(q-1) div 2048
|                 (part q/q)             |  P counter=(q-1) mod 2048
:                                        :  M=1, K=0, L=1, I=00
+=======================================+
```

Figure 6: Example of JPEG XS Payload Data (codestream packetization
                mode, progressive video frame)

```
+=====[ Packetization unit (PU) #1 ]====+
|            Video support box          |  SEP counter=0
+- - - - - - - - - - - - - - - - - - - -+  P counter=0
|         Color specification box       |
+- - - - - - - - - - - - - - - - - - - -+
|      JPEG XS codestream (1st field)   |
:              (part 1/q)               :  M=0, K=0, L=0, I=10
+---------------------------------------+
|      JPEG XS codestream (1st field)   |  SEP counter=0
|              (part 2/q)               |  P counter=1
:                                       :  M=0, K=0, L=0, I=10
+---------------------------------------+
:                                       :
+---------------------------------------+
|      JPEG XS codestream (1st field)   |  SEP counter=1
|             (part 2049/q)             |  P counter=0
:                                       :  M=0, K=0, L=0, I=10
+---------------------------------------+
:                                       :
+---------------------------------------+
|      JPEG XS codestream (1st field)   |  SEP counter=(q-1) div 2048
|              (part q/q)               |  P counter=(q-1) mod 2048
:                                       :  M=1, K=0, L=1, I=10
+==============[ PU #2 ]================+
|            Video support box          |  SEP counter=0
+- - - - - - - - - - - - - - - - - - - -+  P counter=0
|         Color specification box       |
+- - - - - - - - - - - - - - - - - - - -+
|      JPEG XS codestream (2nd field)   |
|              (part 1/q)               |
:                                       :  M=0, K=0, L=0, I=11
+---------------------------------------+
|      JPEG XS codestream (2nd field)   |  SEP counter=0
|              (part 2/q)               |  P counter=1
:                                       :  M=0, K=0, L=0, I=11
+---------------------------------------+
:                                       :
+---------------------------------------+
|      JPEG XS codestream (2nd field)   |  SEP counter=(q-1) div 2048
|              (part q/q)               |  P counter=(q-1) mod 2048
:                                       :  M=1, K=0, L=1, I=11
+=======================================+
```

Figure 7: Example of JPEG XS Payload Data (codestream packetization
                 mode, interlaced video frame)

```
+===[ PU #1: JPEG XS Header segment ]===+
|            Video support box          | SEP counter=0x07FF
+- - - - - - - - - - - - - - - - - - - -+ P counter=0
|          Color specification box      |
+- - - - - - - - - - - - - - - - - - - -+
|         JPEG XS codestream header     |
|   +-------------------------------+   |
|   :  Markers and marker segments  :   |
|   +-------------------------------+   | M=0, T=0, K=1, L=1, I=00
+=========[ PU #2: Slice #1 ]=========+
|   +-------------------------------+   | SEP counter=0
|   |          SLH Marker           |   | P counter=0
|   +-------------------------------+   |
|   :        Entropy Coded Data     :   |
|   +-------------------------------+   | M=0, T=0, K=1, L=1, I=00
+=========[ PU #3: Slice #2 ]=========+
|               Slice #2                | SEP counter=1
|               (part 1/q)              | P counter=0
:                                       : M=0, T=0, K=1, L=0, I=00
+---------------------------------------+
|               Slice #2                | SEP counter=1
|               (part 2/q)              | P counter=1
:                                       : M=0, T=0, K=1, L=0, I=00
+---------------------------------------+
:                                       :
+---------------------------------------+
|               Slice #2                | SEP counter=1
|               (part q/q)              | P counter=q-1
:                                       : M=0, T=0, K=1, L=1, I=00
+=======================================+
:                                       :
+=======[ PU #N: Slice #(N-1) ]=======+
|             Slice #(N-1)              | SEP counter=N-2
|              (part 1/r)               | P counter=0
:                                       : M=0, T=0, K=1, L=0, I=00
+---------------------------------------+
:                                       :
+---------------------------------------+
|             Slice #(N-1)              | SEP counter=N-2
|              (part r/r)               | P counter=r-1
:              + EOC marker             : M=1, T=0, K=1, L=1, I=00
+=======================================+
```

        Figure 8: Example of JPEG XS Payload Data (slice packetization mode,
                        progressive video frame)

```
+====[ PU #1: JPEG XS Hdr segment 1 ]===+
|           Video support box           |  SEP counter=0x07FF
+- - - - - - - - - - - - - - - - - - - -+  P counter=0
|         Color specification box       |
+- - - - - - - - - - - - - - - - - - - -+
|       JPEG XS codestream header 1     |
|  +-------------------------------+  |
|  :   Markers and marker segments   :  |
|  +-------------------------------+  |  M=0, T=0, K=1, L=1, I=10
+====[ PU #2: Slice #1 (1st field) ]====+
|  +-------------------------------+  |  SEP counter=0
|  |            SLH Marker          |  |  P counter=0
|  +-------------------------------+  |
|  :        Entropy Coded Data       :  |
|  +-------------------------------+  |  M=0, T=0, K=1, L=1, I=10
+====[ PU #3: Slice #2 (1st field) ]===+
|               Slice #2                |  SEP counter=1
|               (part 1/q)              |  P counter=0
:                                       :  M=0, T=0, K=1, L=0, I=10
+---------------------------------------+
|               Slice #2                |  SEP counter=1
|               (part 2/q)              |  P counter=1
:                                       :  M=0, T=0, K=1, L=0, I=10
+---------------------------------------+
:                                       :
+---------------------------------------+
|               Slice #2                |  SEP counter=1
|               (part q/q)              |  P counter=q-1
:                                       :  M=0, T=0, K=1, L=1, I=10
+=======================================+
:                                       :
+==[ PU #N: Slice #(N-1) (1st field) ]==+
|            Slice #(N-1)                |  SEP counter=N-2
|            (part 1/r)                  |  P counter=0
:                                       :  M=0, T=0, K=1, L=0, I=10
+---------------------------------------+
:                                       :
+---------------------------------------+
|            Slice #(N-1)                |  SEP counter=N-2
|            (part r/r)                  |  P counter=r-1
:            + EOC marker                :  M=1, T=0, K=1, L=1, I=10
+=======================================+
+===[ PU #N+1: JPEG XS Hdr segment 2 ]==+
|           Video support box           |  SEP counter=0x07FF
+- - - - - - - - - - - - - - - - - - - -+  P counter=0
|         Color specification box       |
+- - - - - - - - - - - - - - - - - - - -+
|       JPEG XS codestream header 2     |
```

```
     |   +-------------------------------+   |
     |   :   Markers and marker segments    :   |
     |   +-------------------------------+   |  M=0, T=0, K=1, L=1, I=11
     +===[ PU #N+2: Slice #1 (2nd field) ]===+
     |   +-------------------------------+   |  SEP counter=0
     |   |          SLH Marker          |   |  P counter=0
     |   +-------------------------------+   |
     |   :       Entropy Coded Data       :   |
     |   +-------------------------------+   |  M=0, T=0, K=1, L=1, I=11
     +===[ PU #N+3: Slice #2 (2nd field) ]===+
     |              Slice #2             |  SEP counter=1
     |              (part 1/s)           |  P counter=0
     :                                   :  M=0, T=0, K=1, L=0, I=11
     +-----------------------------------+
     |              Slice #2             |  SEP counter=1
     |              (part 2/s)           |  P counter=1
     :                                   :  M=0, T=0, K=1, L=0, I=11
     +-----------------------------------+
     :                                   :
     +-----------------------------------+
     |              Slice #2             |  SEP counter=1
     |              (part s/s)           |  P counter=s-1
     :                                   :  M=0, T=0, K=1, L=1, I=11
     +===================================+
     :                                   :
     +==[ PU #2N: Slice #(N-1) (2nd field) ]=+
     |            Slice #(N-1)           |  SEP counter=N-2
     |              (part 1/t)           |  P counter=0
     :                                   :  M=0, T=0, K=1, L=0, I=11
     +-----------------------------------+
     :                                   :
     +-----------------------------------+
     |            Slice #(N-1)           |  SEP counter=N-2
     |              (part t/t)           |  P counter=t-1
     :              + EOC marker         :  M=1, T=0, K=1, L=1, I=11
     +===================================+
```

              Figure 9: Example of JPEG XS Payload Data (slice packetization mode,
                            interlaced video frame)

5. Traffic Shaping and Delivery Timing

   In order to facilitate proper synchronization between senders and
   receivers it is RECOMMENDED to implement traffic shaping and delivery
   timing in accordance with the Network Compatibility Model compliance
   definitions specified in [SMPTE-ST2110-21]. In such case, the
   session description SHALL signal the compliance with the media type
   parameter TP. The actual applied traffic shaping and timing delivery

mechanism is outside the scope of this memo and does not influence
the payload packetization.

6. Congestion Control Considerations

Congestion control for RTP SHALL be used in accordance with
[RFC3550], and with any applicable RTP profile: e.g. RTP/AVP
[RFC3551] or RTP/AVPF [RFC4585].

While JPEG XS is mainly designed to be used in controlled network
environments, it can also be employed in best-effort network
environments, like the Internet. However, in this case the users of
this payload format SHALL monitor packet loss to ensure that the
packet loss rate is within acceptable parameters. This can be
achieved for example by means of the RTP Control Protocol (RTCP)
Feedback for Congestion Control [RFC8888].

In addition, Circuit Breakers [RFC8083] is an update to RTP [RFC3550]
that defines criteria for when one is required to stop sending RTP
Packet Streams and applications implementing this standard SHALL
comply with it.

[RFC8085] provides additional information on the best practices for
applying congestion control to UDP streams.

7. Payload Format Parameters

This section specifies the required and optional parameters of the
payload format and/or the RTP stream. All parameters are
declarative, meaning that the information signaled by the parameters
is also present in the payload data, namely in the payload header
(see Section 4.3) or in the JPEG XS header segment [ISO21122-1]
[ISO21122-3]. When provided, their respective values SHALL be
consistent with the payload.

7.1. Media Type Registration

This registration is done using the template defined in [RFC6838]
and following [RFC4855].

The receiver SHALL ignore any unrecognized parameter.

Type name: video

Subtype name: jxsv

Clock rate: 90000

Required parameters:

   rate: The RTP timestamp clock rate. Applications using this
      payload format SHALL use a value of 90000.

   packetmode: This parameter specifies the configured packetization
      mode as defined by the pacKetization mode (K) bit in the
      payload header of Section 4.3. This value SHALL be equal to
      the K bit value configured in the RTP stream (i.e. 0 for
      codestream or 1 for slice).

Optional parameters:

   transmode: This parameter specifies the configured transmission
      mode as defined by the Transmission mode (T) bit in the payload
      header of Section 4.3. If specified, this value SHALL be equal
      to the T bit value configured in the RTP stream (i.e. 0 for
      out-of-order-allowed or 1 for sequential-only). If not
      specified, a value 1 (sequential-only) SHALL be assumed and the
      T bit SHALL be set to 1.

   profile: The JPEG XS profile [ISO21122-2] in use. Any white
      space in the profile name SHALL be omitted. Examples of valid
      profile names are 'Main444.12' or 'High444.12'.

   level: The JPEG XS level [ISO21122-2] in use. Any white space in
      the level name SHALL be omitted. Examples of valid levels are
      '2k-1' or '4k-2'.

   sublevel: The JPEG XS sublevel [ISO21122-2] in use. Any white
      space in the sublevel name SHALL be omitted. Examples of valid
      sublevels are 'Sublev3bpp' or 'Sublev6bpp'.

   depth: Determines the number of bits per sample. This is an
      integer with typical values including 8, 10, 12, and 16.

   width: Determines the number of pixels per line. This is an
      integer between 1 and 32767 inclusive.

   height: Determines the number of lines per video frame. This is
      an integer between 1 and 32767 inclusive.

   exactframerate: Signals the video frame rate in frames per
      second. Integer frame rates SHALL be signaled as a single
      decimal number (e.g. "25") whilst non-integer frame rates SHALL
      be signaled as a ratio of two integer decimal numbers separated
      by a "forward-slash" character (e.g. "30000/1001"), utilizing
      the numerically smallest numerator value possible.

   interlace: If this parameter name is present, it indicates that
      the video is interlaced, or that the video is Progressive
      segmented Frame (PsF). If this parameter name is not present,
      the progressive video format SHALL be assumed.

   segmented: If this parameter name is present, and the interlace
      parameter name is also present, then the video is a Progressive
      segmented Frame (PsF). Signaling of this parameter without the
      interlace parameter is forbidden.

   sampling: Signals the color difference signal sub-sampling
      structure.

      Signals utilizing the non-constant luminance Y'C'B C'R signal
      format of Recommendation ITU-R BT.601-7, Recommendation ITU-R
      BT.709-6, Recommendation ITU-R BT.2020-2, or Recommendation
      ITU-R BT.2100 SHALL use the appropriate one of the following
      values for the Media Type Parameter "sampling":

         YCbCr-4:4:4   (4:4:4 sampling)
         YCbCr-4:2:2   (4:2:2 sampling)
         YCbCr-4:2:0   (4:2:0 sampling)

      Signals utilizing the Constant Luminance Y'C C'BC C'RC signal
      format of Recommendation ITU-R BT.2020-2 SHALL use the
      appropriate one of the following values for the Media Type
      Parameter "sampling":

         CLYCbCr-4:4:4 (4:4:4 sampling)
         CLYCbCr-4:2:2 (4:2:2 sampling)
         CLYCbCr-4:2:0 (4:2:0 sampling)

      Signals utilizing the constant intensity I CT CP signal format
      of Recommendation ITU-R BT.2100 SHALL use the appropriate one
      of the following values for the Media Type Parameter
      "sampling":

         ICtCp-4:4:4   (4:4:4 sampling)
         ICtCp-4:2:2   (4:2:2 sampling)
         ICtCp-4:2:0   (4:2:0 sampling)

      Signals utilizing the 4:4:4 R' G' B' or RGB signal format (such
      as that of Recommendation ITU-R BT.601, Recommendation ITU-R
      BT.709, Recommendation ITU-R BT.2020, Recommendation ITU-R
      BT.2100, SMPTE ST 2065-1 or ST 2065-3) SHALL use the following
      value for the Media Type Parameter sampling.

         RGB   (RGB or R' G' B' samples)

Signals utilizing the 4:4:4 X' Y' Z' signal format (such as
defined in SMPTE ST 428-1) SHALL use the following value for
the Media Type Parameter sampling.

   XYZ   (X' Y' Z' samples)

Key signals as defined in SMPTE RP 157 SHALL use the value key
for the Media Type Parameter sampling. The Key signal is
represented as a single component.

   KEY   (Samples of the key signal)

Signals utilizing a color sub-sampling other than what is
defined here SHALL use the following value for the Media Type
Parameter sampling.

   UNSPECIFIED   (Sampling signaled by the payload.)

colorimetry: Specifies the system colorimetry used by the image
samples. Valid values and their specification are:

   BT601-5   ITU-R Recommendation BT.601-5.
   BT709-2   ITU-R Recommendation BT.709-2.
   SMPTE240M   SMPTE ST 240M.
   BT601   ITU-R Recommendation BT.601-7.
   BT709   ITU-R Recommendation BT.709-6.
   BT2020   ITU-R Recommendation BT.2020-2.
   BT2100   ITU-R Recommendation BT.2100
             Table 2 titled "System colorimetry".
   ST2065-1   SMPTE ST 2065-1 Academy Color Encoding
             Specification (ACES).
   ST2065-3   SMPTE ST 2065-3 Academy Density Exchange
             Encoding (ADX).
   XYZ   ISO/IEC 11664-1, section titled
             "1931 Observer".
   UNSPECIFIED Colorimetry is signaled in the payload by
             the color specification box of [ISO21122-3],
             or it must be manually coordinated between
             sender and receiver.

Signals utilizing the Recommendation ITU-R BT.2100 colorimetry
SHOULD also signal the representational range using the
optional parameter RANGE defined below. Signals utilizing the
UNSPECIFIED colorimetry might require manual coordination
between the sender and the receiver.

TCS: Transfer Characteristic System. This parameter specifies
   the transfer characteristic system of the image samples. Valid
   values and their specification are:

     SDR   Standard Dynamic Range video streams that
   utilize the OETF of ITU-R Recommendation
   BT.709 or ITU-R Recommendation BT.2020. Such
   streams SHALL be assumed to target the EOTF
   specified in ITU-R Recommendation BT.1886.

     PQ   High dynamic range video streams that utilize
   the Perceptual Quantization system of ITU-R
   Recommendation BT.2100.

     HLG   High dynamic range video streams that utilize
   the Hybrid Log-Gamma system of ITU-R
   Recommendation BT.2100.

     UNSPECIFIED Video streams whose transfer characteristics
   are signaled by the payload as specified in
   [ISO21122-3], or must be manually
   coordinated between sender and receiver.

RANGE: This parameter SHOULD be used to signal the encoding range
   of the sample values within the stream. When paired with ITU
   Rec BT.2100 colorimetry, this parameter has two allowed values
   NARROW and FULL, corresponding to the ranges specified in table
   9 of ITU Rec BT.2100. In any other context, this parameter has
   three allowed values: NARROW, FULLPROTECT, and FULL, which
   correspond to the ranges specified in SMPTE RP 2077. In the
   absence of this parameter, and for all but the UNSPECIFIED
   colorimetry, NARROW SHALL be the assumed value. When paired
   with the UNSPECIFIED colorimetry, FULL SHALL be the default
   assumed value.

Encoding considerations:
   This media type is framed in RTP and contains binary data; see
   Section 4.8 in [RFC6838].

Security considerations:
   Please see the Security Considerations (Section 10) of RFC XXXX.

Interoperability considerations:
   None.

Published specification:
   See RFC XXXX and its References section.

Applications that use this media type:
   Any application that transmits video over RTP (like SMPTE ST
   2110).

   Fragment identifier considerations:
      N/A.

   Additional information:
      None.

   Person & email address to contact for further information:
      S. Lugan <rtp@intopix.com> and Th. Richter <jpeg-xs-
      techsupport@iis.fraunhofer.de>.

   Intended usage:
      COMMON

   Restrictions on usage:
      This media type depends on RTP framing, and hence is only defined
      for transfer via RTP [RFC3550].

   Author:
      See the Authors' Addresses section of RFC XXXX.

   Change controller:
      IETF Audio/Video Transport working group delegated from the IESG.

8. SDP Parameters

   A mapping of the parameters into the Session Description Protocol
   (SDP) [RFC8866] is provided for applications that use SDP.

8.1. Mapping of Payload Type Parameters to SDP

   The media type video/jxsv string is mapped to fields in the Session
   Description Protocol (SDP) [RFC8866] as follows:

      The media type ("video") goes in SDP "m=" as the media name.

      The media subtype ("jxsv") goes in SDP "a=rtpmap" as the encoding
      name, followed by a slash ("/") and the required parameter "rate"
      corresponding to the RTP timestamp clock rate (which for the
      payload format defined in this document SHALL be 90000).

      The required parameter "packetmode", and any of the additional
      optional parameters, as described in Section 7.1, go in the SDP
      media format description, being the "a=fmtp" attribute (Format
      Parameters), by copying them directly from the MIME media type
      string as a semicolon-separated list of parameter=value pairs.

   All parameters of the media format SHALL correspond to the parameters
   of the payload. In case of discrepancies between payload parameter

values and SDP fields, the values from the payload data SHALL
prevail.

The receiver SHALL ignore any parameter that is not defined in
Section 7.1.

An example SDP mapping for JPEG XS video is as follows:

```
m=video 30000 RTP/AVP 112
a=rtpmap:112 jxsv/90000
a=fmtp:112 packetmode=0;sampling=YCbCr-4:2:2;
           width=1920;height=1080;depth=10;
           colorimetry=BT709;TCS=SDR;RANGE=FULL;TP=2110TPNL
```

In this example, a JPEG XS RTP stream is to be sent to UDP
destination port 30000, with an RTP dynamic payload type of 112 and a
media clock rate of 90000 Hz. Note that the "a=fmtp:" line has been
wrapped to fit this page, and will be a single long line in the SDP
file. This example includes the TP parameter (as specified in
Section 5).

8.2. Usage with SDP Offer/Answer Model

When JPEG XS is offered over RTP using SDP in an offer/answer model
[RFC3264] for negotiation for unicast usage, the following
limitations and rules apply:

The "a=fmtp" attribute SHALL be present specifying the required
parameter "packetmode", and MAY specify any of the optional
parameters, as described in Section 7.1.

All parameters in the "a=fmtp" attribute indicate sending
capabilities (i.e. properties of the payload).

An answerer of the SDP is required to support all parameters and
values of the parameters provided by the offerer; otherwise, the
answerer SHALL reject the session. It falls on the offerer to use
values that are expected to be supported by the answerer. If the
answerer accepts the session, it SHALL reply with the exact same
parameters values in the "a=fmtp" attribute as it was offered.

The same RTP payload type number used in the offer SHOULD be used
in the answer, as specified in [RFC3264].

9. IANA Considerations

   The IANA is requested to register the media type registration "video/
   jxsv" as specified in Section 7.1. The media type is also requested
   to be added to the IANA registry for "RTP Payload Format MIME types"
   <https://www.iana.org/assignments/rtp-parameters>.

10. Security Considerations

   RTP packets using the payload format defined in this memo are subject
   to the security considerations discussed in [RFC3550] and in any
   applicable RTP profile such as RTP/AVP [RFC3551], RTP/AVPF [RFC4585],
   RTP/SAVP [RFC3711], or RTP/SAVPF [RFC5124]. This implies that
   confidentiality of the media streams is achieved by encryption.

   However, as "Securing the RTP Framework: Why RTP Does Not Mandate a
   Single Media Security Solution" [RFC7202] discusses, it is not an RTP
   payload format's responsibility to discuss or mandate what solutions
   are used to meet the basic security goals like confidentiality,
   integrity, and source authenticity for RTP in general. This
   responsibility lies on anyone using RTP in an application. They can
   find guidance on available security mechanisms and important
   considerations in "Options for Securing RTP Sessions" [RFC7201].
   Applications SHOULD use one or more appropriate strong security
   mechanisms.

   Implementations of this RTP payload format need to take appropriate
   security considerations into account. It is important for the
   decoder to be robust against malicious or malformed payloads and
   ensure that they do not cause the decoder to overrun its allocated
   memory or otherwise misbehave. An overrun in allocated memory could
   lead to arbitrary code execution by an attacker. The same applies to
   the encoder, even though problems in encoders are typically rarer.

   This payload format and the JPEG XS encoding do not exhibit any
   substantial non-uniformity, either in output or in complexity to
   perform the decoding operation and thus are unlikely to pose a
   denial-of-service threat due to the receipt of pathological
   datagrams.

   This payload format and the JPEG XS encoding do not contain code that
   is executable.

   It is important to note that HD or UHDTV JPEG XS-encoded video can
   have significant bandwidth requirements (typically more than 1 Gbps
   for ultra high-definition video, especially if using high framerate).
   This is sufficient to cause potential for denial-of-service if
   transmitted onto most currently available Internet paths.

Accordingly, if best-effort service is being used, users of this
payload format SHALL monitor packet loss to ensure that the packet
loss rate is within acceptable parameters. Packet loss is considered
acceptable if a TCP flow across the same network path, and
experiencing the same network conditions, would achieve an average
throughput, measured on a reasonable timescale, that is not less than
the RTP flow is achieving. This condition can be satisfied by
implementing congestion control mechanisms to adapt the transmission
rate (or the number of layers subscribed for a layered multicast
session), or by arranging for a receiver to leave the session if the
loss rate is unacceptably high.

This payload format may also be used in networks that provide
quality-of-service guarantees. If enhanced service is being used,
receivers SHOULD monitor packet loss to ensure that the service that
was requested is actually being delivered. If it is not, then they
SHOULD assume that they are receiving best-effort service and behave
accordingly.

11. Acknowledgments

The authors would like to thank the following people for their
valuable contributions to this memo: Arnaud Germain, Alexandre
Willeme, Gael Rouvroy, Siegfried Foessel, and Jean-Baptise Lorent.

12. RFC Editor Considerations

Note to RFC Editor: This section may be removed after carrying out
all the instructions of this section.

RFC XXXX is to be replaced by the RFC number this specification
receives when published.

13. References

13.1. Normative References

[ISO21122-1]
          International Organization for Standardization (ISO) -
          International Electrotechnical Commission (IEC),
          "Information technology - JPEG XS low-latency lightweight
          image coding system - Part 1: Core coding system", ISO/
          IEC IS 21122-1.

   [ISO21122-2]
              International Organization for Standardization (ISO) -
              International Electrotechnical Commission (IEC),
              "Information technology - JPEG XS low-latency lightweight
              image coding system - Part 2: Profiles and buffer models",
              ISO/IEC IS 21122-2.

   [ISO21122-3]
              International Organization for Standardization (ISO) -
              International Electrotechnical Commission (IEC),
              "Information technology - JPEG XS low-latency lightweight
              image coding system - Part 3: Transport and container
              formats", ISO/IEC IS 21122-3.

   [RFC2119] Bradner, S., "Key words for use in RFCs to Indicate
              Requirement Levels", BCP 14, RFC 2119,
              DOI 10.17487/RFC2119, March 1997,
              <https://www.rfc-editor.org/info/rfc2119>.

   [RFC3264] Rosenberg, J. and H. Schulzrinne, "An Offer/Answer Model
              with Session Description Protocol (SDP)", RFC 3264,
              DOI 10.17487/RFC3264, June 2002,
              <https://www.rfc-editor.org/info/rfc3264>.

   [RFC3550] Schulzrinne, H., Casner, S., Frederick, R., and V.
              Jacobson, "RTP: A Transport Protocol for Real-Time
              Applications", STD 64, RFC 3550, DOI 10.17487/RFC3550,
              July 2003, <https://www.rfc-editor.org/info/rfc3550>.

   [RFC3551] Schulzrinne, H. and S. Casner, "RTP Profile for Audio and
              Video Conferences with Minimal Control", STD 65, RFC 3551,
              DOI 10.17487/RFC3551, July 2003,
              <https://www.rfc-editor.org/info/rfc3551>.

   [RFC4855] Casner, S., "Media Type Registration of RTP Payload
              Formats", RFC 4855, DOI 10.17487/RFC4855, February 2007,
              <https://www.rfc-editor.org/info/rfc4855>.

   [RFC6838] Freed, N., Klensin, J., and T. Hansen, "Media Type
              Specifications and Registration Procedures", BCP 13,
              RFC 6838, DOI 10.17487/RFC6838, January 2013,
              <https://www.rfc-editor.org/info/rfc6838>.

   [RFC8083] Perkins, C. and V. Singh, "Multimedia Congestion Control:
              Circuit Breakers for Unicast RTP Sessions", RFC 8083,
              DOI 10.17487/RFC8083, March 2017,
              <https://www.rfc-editor.org/info/rfc8083>.

   [RFC8085] Eggert, L., Fairhurst, G., and G. Shepherd, "UDP Usage
              Guidelines", BCP 145, RFC 8085, DOI 10.17487/RFC8085,
              March 2017, <https://www.rfc-editor.org/info/rfc8085>.

   [RFC8174] Leiba, B., "Ambiguity of Uppercase vs Lowercase in RFC
              2119 Key Words", BCP 14, RFC 8174, DOI 10.17487/RFC8174,
              May 2017, <https://www.rfc-editor.org/info/rfc8174>.

   [RFC8866] Begen, A., Kyzivat, P., Perkins, C., and M. Handley, "SDP:
              Session Description Protocol", RFC 8866,
              DOI 10.17487/RFC8866, January 2021,
              <https://www.rfc-editor.org/info/rfc8866>.

13.2. Informative References

   [RFC3711] Baugher, M., McGrew, D., Naslund, M., Carrara, E., and K.
              Norrman, "The Secure Real-time Transport Protocol (SRTP)",
              RFC 3711, DOI 10.17487/RFC3711, March 2004,
              <https://www.rfc-editor.org/info/rfc3711>.

   [RFC4175] Gharai, L. and C. Perkins, "RTP Payload Format for
              Uncompressed Video", RFC 4175, DOI 10.17487/RFC4175,
              September 2005, <https://www.rfc-editor.org/info/rfc4175>.

   [RFC4585] Ott, J., Wenger, S., Sato, N., Burmeister, C., and J. Rey,
              "Extended RTP Profile for Real-time Transport Control
              Protocol (RTCP)-Based Feedback (RTP/AVPF)", RFC 4585,
              DOI 10.17487/RFC4585, July 2006,
              <https://www.rfc-editor.org/info/rfc4585>.

   [RFC5124] Ott, J. and E. Carrara, "Extended Secure RTP Profile for
              Real-time Transport Control Protocol (RTCP)-Based Feedback
              (RTP/SAVPF)", RFC 5124, DOI 10.17487/RFC5124, February
              2008, <https://www.rfc-editor.org/info/rfc5124>.

   [RFC7201] Westerlund, M. and C. Perkins, "Options for Securing RTP
              Sessions", RFC 7201, DOI 10.17487/RFC7201, April 2014,
              <https://www.rfc-editor.org/info/rfc7201>.

   [RFC7202] Perkins, C. and M. Westerlund, "Securing the RTP
              Framework: Why RTP Does Not Mandate a Single Media
              Security Solution", RFC 7202, DOI 10.17487/RFC7202, April
              2014, <https://www.rfc-editor.org/info/rfc7202>.

   [RFC8888] Sarker, Z., Perkins, C., Singh, V., and M. Ramalho, "RTP
              Control Protocol (RTCP) Feedback for Congestion Control",
              RFC 8888, DOI 10.17487/RFC8888, January 2021,
              <https://www.rfc-editor.org/info/rfc8888>.

   [SMPTE-ST2110-21]
              Society of Motion Picture and Television Engineers, "SMPTE
              Standard - Professional Media Over Managed IP Networks:
              Traffic Shaping and Delivery Timing for Video", SMPTE ST
              2110-21:2017, 2017,
              <https://doi.org/10.5594/SMPTE.ST2110-21.2017>.

Authors' Addresses

   Sebastien Lugan
   intoPIX S.A.
   Rue Emile Francqui, 9
   1435 Mont-Saint-Guibert
   Belgium

   Phone: +32 10 23 84 70
   Email: rtp@intopix.com
   URI: https://www.intopix.com/

   Antonin Descampe
   Universite catholique de Louvain
   Place du Levant, 3 - bte L5.03.02
   1348 Louvain-la-Neuve
   Belgium

   Phone: +32 10 47 25 97
   Email: antonin.descampe@uclouvain.be
   URI: https://uclouvain.be/en/research-institutes/icteam

   Corentin Damman
   intoPIX S.A.
   Rue Emile Francqui, 9
   1435 Mont-Saint-Guibert
   Belgium

   Phone: +32 10 23 84 70
   Email: c.damman@intopix.com
   URI: https://www.intopix.com/

Thomas Richter
Fraunhofer IIS
Am Wolfsmantel 33
91048 Erlangen
Germany

Phone: +49 9131 776 5126
Email: thomas.richter@iis.fraunhofer.de
URI: https://www.iis.fraunhofer.de/

Tim Bruylants
intoPIX S.A.
Rue Emile Francqui, 9
1435 Mont-Saint-Guibert
Belgium

Phone: +32 10 23 84 70
Email: t.bruylants@intopix.com
URI: https://www.intopix.com/