INTERNET-DRAFT
draft-ietf-avt-rtp-jpeg2000-beam-00.txt

Andrew Leung
Satoshi Futemma
Eisaburo Itakura
Sony Corporation

February 14, 2005
Expires: August 25, 2005

Enhanced Processing For Priority and Header in:
RTP Payload Format for JPEG 2000 Video Streams

Status of this Memo

Abstract

This memo describes extended uses for payload header in RFCXXXY, An RTP Payload Format for JPEG 2000 Video, for better support of JPEG 2000 features such as scalability and includes a main header recovery method.

This memo MUST be accompanied with an implementation of RFCXXXY for a complete implementation. RFCXXXY itself is a complete description of the payload header and signaling, this document only describes additional processing of values for the payload header. There is also another MIME and SDP marker signaling for implementations of this document.

1. Introduction

This document is an extension of RFCXXXY, An RTP Payload Format for

JPEG 2000 Video. There are additional mechanisms to be used with certain parts of the header in RFCXXXY to support JPEG 2000 features such as scalability and a main header recovery method. These mechanisms are described in detail in this document.

History

In the development of RFCXXXY, Sony Corporation filed a patent application on certain mechanisms with the main header recovery, priority table usage, etc. As these are not "essential" to the core RTP format and only describes a mechanism, it was decided that splitting these mechanisms from the core RTP format (essentially IPR free) into another document (IPR). This is the document describing the mechanisms.

Description of the mechanisms

o Main Header Compensation

JPEG 2000's scalable coding scheme allows for decompressing truncated or partial data streams but only when the main header is present. If the header is lost, the data is useless. Also, with JPEG 2000 video coding, coding parameters between frames will rarely change and previous headers may be used in newly received data without headers.

A recovery of the main header that has been lost is very simple with this procedure. In the case of JPEG 2000 video, it is common that encode parameters will not vary greatly between each successive frame. Even if the RTP packet including the main header of a frame has been dropped, decoding may be performed by using the main header of a previous frame.

o Priority Table

JPEG 2000 codestream has rich functionality built into it so decoders can easily handle scalable delivery or progressive transmission. Progressive transmission allows images to be reconstructed with increasing pixel accuracy or spatial resolution. This feature allows the reconstruction of images with different resolutions and pixel accuracy, for different target devices. A single image source can provide a codestream that is easily processed for smaller image display devices.

JPEG 2000 packets contain all compressed image data from a specific: layer, component, resolution level, and/or precinct. The order in which these JPEG 2000 packets are found in the codestream is called: progression order. The ordering of the JPEG 2000 packets can progress along four axes: layer, component, resolution and precinct.

Providing a priority field to indicate the importance of data contained in a given RTP packet can aid in usage of JPEG 2000 progressive and scalable functions.

1.1 Conventions Used in this Document

The key words "MUST", "MUST NOT", "REQUIRED", "SHALL", "SHALL NOT", "SHOULD", "SHOULD NOT", "RECOMMENDED", "MAY", and "OPTIONAL" in this document are to be interpreted as described in RFC2119 [1].

2. Payload Format Enhanced Processing

2.1 Enhanced Processing Markers

This section of the document describes changes in the mh_id and priority value which differ from RFCXXXY. Implementions of this document should follow protocol in RFCXXXY first then add in additional header processing for this document. Implementations following this document are expected to seamlessly work with implementations of just RFCXXXY.

The RTP payload header format for JPEG 2000 video stream is as follows:

```
 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
+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
|tp |MHF|mh_id|T|     priority  |           tile number         |
+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
|reserved       |             fragment offset                   |
+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
```

Fig. 3: RTP payload header format for JPEG 2000

mh_id (Main Header Identification) : 3 bits

Main header identification value. This is used for JPEG 2000 main header recovery.

The same mh_id value is used as long as the coding parameters described in the main header remains unchanged.

The initial value of mh_id is random, and may take any value between 1-7, but MUST NOT be 0.

When mh_id is 0, it has special usage for the receiver. This special usage is described in Section xxx of this document.

The mh_id value MUST increment by 1 every time a new main header is transmitted. Once the mh_id value is greater than 7, it rolls over to 1.

priority : 8 bits

The priority field indicates the importance of the JPEG 2000 packet included in the payload. Typically, a higher priority is set in the packets containing JPEG 2000 packets containing the lower sub-bands.

This header is described in detail in documentXXX. Systems not using the method described in documentXXX at the sender this value SHOULD be set to 0 and receivers SHOULD ignore this value.

Special values of priority:

0 : This is reserved for payload which contain a header (main or tile part header.) This is considered the highest importance.

1 to 255 : These values decrease in importance as the values increase. (i.e. 1 is more important than 2, etc.) Hence applying priority values should correlate directly to JPEG 2000 codestream in importance in basic usage.

The lower the priority value is the higher the priority. Simply, the priority value 0 is the highest priority and 255 is the lowest priority. We define the priority value 0 as a special priority value for the headers (the main header or tile-part header). When any headers (the main header or tile-part header) are packed into the RTP packet, the sender MUST set the priority value to 0.

3. Priority Mapping Table

For the progression order, the priority value for each JPEG 2000 packet is given by the priority mapping table.

3.1 Pre-Defined Priority Mapping

This document specify several commonly-used several priority mapping table, pre-defined priority mapping tables: packet number based (default), progression-based, layer-based, resolution-based, component-based.

Packet number priority mapping is REQUIRED to be supported by clients implementing this specification. Other priority mapping tables (progression, layer, resolution, and component based) are OPTIONAL to implementations of this specification.

Rules that all implementations of this specification MUST follow in all priority modes:

When there is a header in the packet with a JPEG 2000 packet, the sender MUST set the payload packet priority value to 0.

When there are multiple JPEG 2000 packets in the same RTP payload packet, the sender MUST set the payload packet priority value to the lowest priority value of the lowest JPEG 2000 packet. (i.e. if JPEG 2000 packets with priority: 5,6,7 are packed into a single payload, the priority value MUST be 5.)

3.1.1 Packet number based Ordering

This is the default mode for payload packet priority value and all implementation of this specification MUST support.

The sender will have a one-to-one association between payload packet priority value and the payload packet value (i.e. the JPEG 2000 codestream.) The RTP packet value is equal to the JPEG 2000 packet value.

If the packet value of JPEG 2000 codestream is greater than 255, the sender MUST set the payload priority value to 255.

3.1.2 Progression-based Ordering

The sender will assign the payload packet priority value only based on layer, resolution, and component ordering of the codestream.

This is similar to the JPEG 2000 packet number based format but will not take into account the precinct number or position in the JPEG 2000 codestream.

For example:
If the codestream is ordered in LRCP (Layer, Resolution, Component, Position)

```
All the packets in layer       0
                   resolution  0
                   component   0 : packet priority value : 1

All the packets in layer       0
                   resolution  0
                   component   1 : packet priority value : 2

All the packets in layer       0
                   resolution  0
                   component   2 : packet priority value : 3
```

3.1.3 Layer-based Ordering

Layer-based priority mapping table simplifies the default mapping to just matching JPEG 2000 packets together from the same layer.

For example:
All the packets in layer 0 : packet priority value : 1
All the packets in layer 1 : packet priority value : 2
All the packets in layer 2 : packet priority value : 3
All the packets in layer 3 : packet priority value : 4

3.1.4 Resolution-based Ordering

Resolution-based priority mapping table is similar to the layer based order but for JPEG 2000 packets of the same resolution

For example:
All the packets in resolution 0 : packet priority value : 1
All the packets in resolution 1 : packet priority value : 2
All the packets in resolution 2 : packet priority value : 3
All the packets in resolution 3 : packet priority value : 4

3.1.5 Component-based Ordering

Component-based priority mapping table is mapping together JPEG 2000 components of the same component

For example:
All the packets in component 0 : packet priority value : 1
All the packets in component 1 : packet priority value : 2
All the packets in component 2 : packet priority value : 3
All the packets in component 3 : packet priority value : 4

3.2 Application Specific Priority Table

The application specific priority table specification is intended for experimental use as new applications and new priority mapping tables are developed.

A case sensitive 8 character ASCII code describing the application specific priority mapping name. The description of these application specific priority tables are outside the scope of this document.

This extension may be used when codestream is divided into many layers and many resolutions.

4. JPEG 2000 Main Header Compensation Scheme

The mh_id field of the payload header is used to recognize whether the encoding parameters of the main header are the same as the encoding parameters of the previous frame. The same value is set in mh_id of the RTP packet in the same frame. The mh_id and encode parameters are not associated with each other as 1:1 but they are used to recognize whether the encode parameters of the previous frame are the same or not in the event of lost headers.

The mh_id field value SHOULD be saved from previous frames to be used to recover the current frame's main header. If the mh_id of the current frame has the same value as the mh_id value of the previous frame, the previous frame's main header SHOULD be used to decode the current frame, in case of a lost header.

The sender MUST increment mh_id when parameters in the header change and send a new main header accordingly.

The receiver MAY use the mh_id and MAY retain the header for such compensation.

4.1 Sender Processing

The sender MUST transmit RTP packets with the same mh_id value unless the encoder parameters are different from the previous frame. The encoding parameters are the fixed information marker segment (SIZ marker) and functional marker segments (COD, COC, RGN, QCD, QCC, and POC) specified in JPEG 2000 Part 1 Annex A [2].
An initial value of mh_id MUST be selected randomly between 1 and 7.
If the encode parameters changes, the sender transmitting RTP packets MUST increment the mh_id value by one, but when mh_id value becomes greater than 7, a sender MUST set mh_id value to 1.

4.2 Receiver Processing

When the receiver receives the main header completely, the RTP sequence number, the mh_id and main header should be saved. Only the last main header that was received completely SHOULD be saved. When the mh_id value is 0, the receiver SHOULD NOT save the header.

When the main header is not received, the receiver may compare the current payload header's mh_id value with the saved mh_id value. When the values are the same, decoding may be performed by using the saved main header.

If the mh_id field is set to 0, the receiver MUST not save the main header and MUST NOT compensate for lost headers.

5. Security Consideration

RTP packets using the payload format defined in this specification are subject to the security considerations discussed in the RTP specifications[3] and any applicable profile. This implies that confidentiality of the media streams is achieved by encryption. Data compression used with this payload format is applied end-to-end, encryption may be performed on the compressed data so there is no conflict between the two operations.

A potential denial-of-service threat exists for data encodings using compression techniques that have non-uniform receiver-end computational load. The attacker can inject pathological datagrams into the stream which are complex to decode and cause the receiver to be overloaded. The usage of authentication of at least the RTP packet is RECOMMENDED, for example with SRTP [4].

If QoS enhanced service is used, RTP 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.

If best-effort service is being used, users of this payload format MUST 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, 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.

As with any IP-based protocol, in some circumstances a receiver may be overloaded simply by receiving too many packets, either desired or undesired. Network-layer authentication may be used to discard packets from undesired sources, but the processing cost of the authentication itself may be too high. In a multicast environment, pruning of specific sources may be implemented in future versions of IGMP [7] and in multicast routing protocols to allow a receiver to select which sources are allowed to reach it.

6. IANA Consideration

6.1 MIME Registration

This document defines a new RTP payload name and associated MIME type, jpeg2000.

The receiver MUST ignore any unspecified parameter.

The MIME registration form for JPEG 2000 video stream is enclosed below:

MIME media type name: video

MIME subtype name: jpeg2000

REQUIRED parameters: BEAM

BEAM: Implmentations of this standard must include this option in the parameter list when establishing a session. If this option is supported, by the receiver/sender, both should reply with this option in the mime parameter list. If the answer omits this in the mime parameter list, the answerer does not support this option

Encoding considerations:
JPEG 2000 video stream may be transmitted with RTP as specified in this document.

Security considerations: see section 9 of RFC XXXX.

Interoperability considerations:
JPEG 2000 video stream is a sequence of JPEG 2000 still images. An implementation in compliant with [1] can decode and attempt to display the encoded JPEG 2000 video stream.

Published specification: ISO/IEC 15444-1 | ITU-T Rec. T.800

Applications which use this media type:
video streaming and communication

Additional information: none

Magic number(s): none

File extension(s): none

Macintosh File Type Code(s): none

Person & email address to contact for further information:
Eisaburo Itakura, Satoshi Futemma
Email: {itakura|satosi-f}@sm.sony.co.jp

Intended usage: COMMON

Author/Change controller:
Eisaburo Itakura, Satoshi Futemma
Email: {itakura|satosi-f}@sm.sony.co.jp

6.2 SDP Parameters

In addition to section 7.2 in RFCXXXY:

The MIME media type video/jpeg2000 string is mapped to fields in the Session Description Protocol (SDP) [5] as follows:

- o The media name in the "m=" line of SDP MUST be video.

- o The encoding name in the "a=rtpmap" line of SDP MUST be jpeg2000 (the MIME subtype).

- o The clock rate in the "a=rtpmap" line MUST be 90000.

- o The REQUIRED parameters "BEAM" MUST be included in the "a=fmtp" line of SDP.

- o The OPTIONAL parameters "priority-table-default", "priority-table-definition", "priority-table-protocol", MUST be included in the "a=fmtp" line of SDP.

These parameters are expressed as a MIME media type string, in the form of a semicolon separated list of parameter=value pairs.

Therefore, an example of media representation in SDP is as follows:

```
m=video 49170/2 RTP/AVP 98
a=rtpmap:98 jpeg2000/90000
a=fmtp:98 BEAM;sampling=YCbCr-4:2:0;width=128;height=128
```

7. Usage with the SDP Offer/Answer Model

In addition to section 8 in RFCXXXY:

When offering JPEG 2000 over RTP using SDP in an Offer/Answer model [6], the following rules and limitations apply:

- o All parameters MUST have an acceptable value for that parameter.

- o All parameters MUST correspond to the parameters of the payload.

o The parameters "BEAM" MUST appear in the offer
If the parameter "BEAM" is not in the answer, receivers should not process the header according to this document. Senders SHOULD continue to send data with payload headers according to mechanisms outlined in this document. This is highly recommended for multicast streams where not all receivers are of the same type.

7.1 Examples

An example offer/answer exchanges are provided.

7.1.2 Example 1

Alice offers BEAM functionality, YCbCr 422 color space, interlace image with 720-pixel width and 480-pixel height and several priority-table options (jp2-packet, progression, layer, resolution, component) as below:

```
v=0
o=alice 2890844526 2890844526 IN IP4 host.anywhere.com
s=
c=IN IP4 host.anywhere.com
t=0 0
m=video 49170 RTP/AVP 98
a=rtpmap:98 jpeg2000/90000
a=fmtp:98 BEAM;sampling=YCbCr-4:2:2;interlace
a=fmtp:98 priority-table-definition=jp2-packet,progression,layer,
  resolution,component; width=720; height=480
```

Bob accepts BEAM functionality, YCbCr-4:2:2 color space,interlace image and jp2-packet based priority mapping (default mapping table) and replies:

```
v=0
o=bob 2890844730 2890844731 IN IP4 host.example.com
s=
c=IN IP4 host.example.com
t=0 0
m=video 49920 RTP/AVP 98
a=rtpmap:98 jpeg2000/90000
a=fmtp:98 BEAM;sampling=YCbCr-4:2:2;interlace
```

Note that "priority-table-definition" parameter in Bob's answer is omitted, so default priority mapping table (jp2-packet number based priority mapping) is used.

7.1.2 Example 2

Alice offers YCbCr 420 color space, progressive image with 320-pixel width and 240-pixel height and layer priority-table options as below:

```
v=0
o=alice 2890844526 2890844526 IN IP4 host.anywhere.com
s=
c=IN IP4 host.anywhere.com
t=0 0
m=video 49170 RTP/AVP 98
a=rtpmap:98 jpeg2000/90000
a=fmtp:98 BEAM;sampling=YCbCr-4:2:0
a=fmtp:98 priority-table-definition=layer; width=320; height=240
```

Bob does not accept BEAM functionality but accepts YCbCr-4:2:0 color space,interlace image and layer based priority mapping and replies:

```
v=0
o=bob 2890844730 2890844731 IN IP4 host.example.com
s=
c=IN IP4 host.example.com
t=0 0
m=video 49920 RTP/AVP 98
a=rtpmap:98 jpeg2000/90000
a=fmtp:98 sampling=YCbCr-4:2:2
```

Note that "BEAM" parameter was not in Bob's answer so Alice must not use settings described in this document for sending or receiving.

7.2 Sample Headers in Detail

This section has various sample headers in various configurations for reference.

For reference, the 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
+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
|tp |MHF|mh_id|T|     priority  |           tile number         |
+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
|reserved       |             fragment offset                   |
+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
```

For the first packet with the main header, this is what it will look like.

Note, for this example MTU will be taken as: 1500bytes (Ethernet)

Sample 1: Progressive image with single tile, 3500bytes
(i.e. thumbnail)

First Packet:
This packet will have the whole main header.
210bytes

```
 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
+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
|0 0|1 1|1 0 1|1|0 0 0 0 0 0 0 0|0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0|
+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
|0 0 0 0 0 0 0 0|0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0|
+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
|FF4FFF51002F000 ....                                           |
+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
```

Second Packet:
This packet will have a tile header and the first tile part LLband
1500bytes

```
 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
+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
|0 0|1 1|1 0 1|0|0 0 0 0 0 0 0 1|0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0|
+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
|0 0 0 0 0 0 0 0|0 0 0 0 0 0 0 0 0 0 0 0 0 0 1 1 0 1 0 0 1 0|
+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
|FF90 000A 0000 0000 2DB3  0001 FF93                            |
+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
```

Third Packet:
This packet will have the next part in the tile, no tile header
1500bytes

```
 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
+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
|0 0|0 0|1 0 1|0|0 0 0 0 0 0 1 0|0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0|
+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
|0 0 0 0 0 0 0 0|0 0 0 0 0 0 0 0 0 0 0 0 0 1 1 0 1 0 1 0 1 1 1 0|
+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
|E841 4526 4556 9850 C2EA  ....                                 |
+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
```

Fourth Packet:
Last packet for the image
290bytes

```
 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
+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
|0 0|0 0|1 0 1|0|0 0 0 0 0 0 1 1|0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0|
+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
|0 0 0 0 0 0 0 0|0 0 0 0 0 0 0 0 0 0 0 0 1 1 0 0 1 0 0 0 1 0 1 0|
+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
|A55D 8B73 3B25 25C7 B9EB        ....                    2FBEB153|
+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
```

Smaple 2: Image with 4 tiles

First Packet:
This packet will have the whole main header.
210bytes

```
 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
+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
|0 0|1 1|0 0 1|1|0 0 0 0 0 0 0 0|0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0|
+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
|0 0 0 0 0 0 0 0|0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0|
+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
|FF4FFF51002F000 ....                                            |
+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
```

Second Packet:
This packet will have a first tile part (tile 0)
1400bytes

```
 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
+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
|0 0|0 0|0 0 1|0|0 0 0 0 0 0 0 1|0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0|
+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
|0 0 0 0 0 0 0 0|0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 1 1 0 1 0 0 1 0|
+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
|FF90 000A 0000 0000 0578  0001 FF93  ....                       |
+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
```

```
    Third Packet:
    This packet will have a second tile part (tile 1)
    1423bytes
     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
    +-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
    |0 0|0 0|0 0 1|0|0 0 0 0 0 0 0 1|0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 1|
    +-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
    |0 0 0 0 0 0 0 0|0 0 0 0 0 0 0 0 0 0 0 0 0 0 1 1 0 0 1 0 0 1 0 1 0|
    +-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
    |FF90 000A 0001 0000 058F 0001 FF93    ....                     |
    +-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+

    Fourth Packet:
    This packet will have a third tile part (tile 2)
    1355bytes

     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
    +-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
    |0 0|0 0|0 0 1|0|0 0 0 0 0 0 0 1|0 0 0 0 0 0 0 0 0 0 0 0 0 0 1 0|
    +-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
    |0 0 0 0 0 0 0 0|0 0 0 0 0 0 0 0 0 0 0 0 1 0 1 1 1 1 0 1 1 0 0 1|
    +-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
    |FF90 000A 0002 0000 054B 0001 FF93    ....                     |
    +-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+

    Fifth Packet:
    This packet will have a fourth tile part (tile 3)
    1290bytes

     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
    +-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
    |0 0|0 0|0 0 1|0|0 0 0 0 0 0 0 1|0 0 0 0 0 0 0 0 0 0 0 0 0 0 1 1|
    +-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
    |0 0 0 0 0 0 0 0|0 0 0 0 0 0 0 0 0 0 0 1 0 0 0 1 0 0 1 0 0 1 0 0|
    +-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
    |FF90 000A 0003 0000 050A 0001 FF93    ....                     |
    +-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
```

Sample 3: Packing multiple tiles in single payload, fragmented header
No header compensation, progressive image

First Packet:
This packet will have the first part of the main header.
110bytes

```
 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
+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
|0 0|0 1|0 0 0|1|0 0 0 0 0 0 0 0|0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0|
+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
|0 0 0 0 0 0 0 0|0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0|
+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
|FF4FFF51002F000 ....                                           |
+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
```

Second Packet:
This packet has the second part of the header.
1400bytes

```
 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
+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
|0 0|1 0|0 0 0|1|0 0 0 0 0 0 0 0|0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0|
+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
|0 0 0 0 0 0 0 0|0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 1 1 0 1 1 1 0|
+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
|FF6400FF ....                                                  |
+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
```

Third Packet:
This packet has two tiles, tile 0 and tile 1
1400bytes

```
 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
+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
|0 0|0 0|0 0 0|1|0 0 0 0 0 0 0 1|0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0|
+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
|0 0 0 0 0 0 0 0|0 0 0 0 0 0 0 0 0 0 0 0 1 0 1 1 1 1 0 0 1 1 0|
+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
|FF90 000A 0000 0000 02BC 0001 FF93  ...                        |
|FF90 000A 0001 0000 02BC 0001 FF93 ...                         |
+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
```

Fourth Packet:
This packet has one tile, tile 2
1395bytes

```
 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
+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
|0 0|0 0|0 0 0|0|0 0 0 0 0 0 0 1|0 0 0 0 0 0 0 0 0 0 0 0 0 0 1 0|
+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
|0 0 0 0 0 0 0 0|0 0 0 0 0 0 0 0 0 0 0 0 1 0 1 1 0 1 0 1 1 1 1 0|
+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
|FF90 000A 0002 0000 0573 0001 FF93    ....                     |
+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
```

Sample 4: Interlace image, single tile

First packet:
This packet will have the whole main header for the odd field
210bytes

```
 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
+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
|0 1|1 1|0 1 0|1|0 0 0 0 0 0 0 0|0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0|
+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
|0 0 0 0 0 0 0 0|0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0|
+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
|FF4FFF51002F000 ....                                           |
+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
```

Second packet:
This packet will have the first part of the odd field's tile
1400bytes

```
 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
+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
|0 1|0 0|0 1 0|1|0 0 0 0 0 0 0 0|0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0|
+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
|0 0 0 0 0 0 0 0|0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 1 1 0 1 0 0 1 0|
+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
|FF90 000A 0000 0000 0578  0001 FF93  ....                      |
+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
```

Third packet:
This packet will have the second part of the odd field's tile
1400bytes

```
 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
+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
|0 1|0 0|0 1 0|1|0 0 0 0 0 0 0 1|0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0|
+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
|0 0 0 0 0 0 0 0|0 0 0 0 0 0 0 0 0 0 0 0 1 1 0 0 1 0 0 1 0 1 0|
+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
|7F04 E708 27D9 D11D 22CB ...                                    |
+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
```

Fourth packet:
This packet will have the third part of the odd field's tile
1300bytes

```
 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
+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
|0 1|0 0|0 1 0|1|0 0 0 0 0 0 0 1|0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0|
+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
|0 0 0 0 0 0 0 0|0 0 0 0 0 0 0 0 0 0 0 0 1 0 1 1 1 1 0 0 0 0 1 0|
+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
|98BD EC9B 2826 DC62 D4AB ...                                    |
+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
```

Fifth packet:
This packet will have the whole main header for the even field

```
 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
+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
|1 0|1 1|0 1 1|1|0 0 0 0 0 0 0 0|0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0|
+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
|0 0 0 0 0 0 0 0|0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0|
+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
|FF4FFF51002F000 ....                                            |
+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
```

Sixth packet:
This packet will have the first part of the odd field's tile
1400bytes

```
 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
+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
|1 0|0 0|0 1 0|1|0 0 0 0 0 0 0 0|0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0|
+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
|0 0 0 0 0 0 0 0|0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 1 1 0 1 0 0 1 0|
+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
|FF90 000A 0000 0000 0578  0001 FF93  ....                      |
+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
```

Seventh packet:
This packet will have the second part of the odd field's tile
1400bytes

```
 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
+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
|1 0|0 0|0 1 0|1|0 0 0 0 0 0 0 1|0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0|
+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
|0 0 0 0 0 0 0 0|0 0 0 0 0 0 0 0 0 0 0 0 1 1 0 0 1 0 0 1 0 1 0|
+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
|626C 42F0 166B 6BD0 F8E1 ...                                   |
+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
```

Eighth packet:
This packet will have the third part of the odd field's tile
1300bytes

```
 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
+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
|1 0|0 0|0 1 0|1|0 0 0 0 0 0 0 1|0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0|
+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
|0 0 0 0 0 0 0 0|0 0 0 0 0 0 0 0 0 0 0 1 0 1 1 1 1 0 0 0 0 1 0|
+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
|8114 41D5 18AB 4A1B ...                                        |
+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
```

11. Intellectual Property Right Statement

The IETF takes no position regarding the validity or scope of any Intellectual Property Rights or other rights that might be claimed to pertain to the implementation or use of the technology described in this document or the extent to which any license under such rights might or might not be available; nor does it represent that it has made any independent effort to identify any such rights. Information on the procedures with respect to rights in RFC documents can be found in BCP 78 and BCP 79.

Copies of IPR disclosures made to the IETF Secretariat and any assurances of licenses to be made available, or the result of an attempt made to obtain a general license or permission for the use of such proprietary rights by implementers or users of this specification can be obtained from the IETF on-line IPR repository at http://www.ietf.org/ipr.

The IETF invites any interested party to bring to its attention any copyrights, patents or patent applications, or other proprietary rights that may cover technology that may be required to implement this standard. Please address the information to the IETF at ietf-ipr@ietf.org.

12. Informative Appendix

12.1 Recommended Practices

Receiver Processing

In general, the receiver should scan for headers in packets that have an MHF value > 0 to aid in main header recovery.

Receivers should be aware of both BEAM capable and incapbale senders. If the sender is incapable of BEAM functionality, receivers should not interpret headers as described in this document.

13. References

Normative References

[1] S. Bradner, "Key words for use in RFCs to Indicate Requirement Levels", BCP14, RFC2119, March 1997.

[2] ISO/IEC JTC1/SC29, ISO/IEC 15444-1 | ITU-T Rec. T.800 "Information technology - JPEG 2000 image coding system - Part 1: Core coding system", December 2000.

[3] H. Schulzrinne, S. Casner, R. Frederick, and V. Jacobson, "RTP: A Transport Protocol for Real Time Applications", STD 64, RFC 3550, July 2003.

[4] M. Baugher, D. McGrew, M. Naslund, E. Carrara and K. Norrman, "The Secure Real-time Transport Protocol (SRTP)", RFC 3711, March 2004.

[5] M. Handley and V. Jacobson, "SDP: Session Description Protocol", RFC 2327, April 1998.

[6] J. Rosenberg and H. Schulzrinne, "An Offer/Answer Model with Session Description Protocol (SDP)", RFC 3264, June 2002.

Informative References

[7] Deering, S., "Host Extensions for IP Multicasting", STD 5, RFC 1112, August 1989.

14. Authors' Addresses

Andrew Leung/Satoshi Futemma/Eisaburo Itakura
Sony Corporation
6-7-35 Kitashinagawa Shinagawa-ku
Tokyo 141-0001 JAPAN
Phone: +81 3 5448 2125
Fax: +81 3 5448 4560
Email: andrew.leung@jp.sony.com, {satosi-f|itakura}@sm.sony.co.jp

15. Full Copyright Statement