Network Working Group
Request for Comments: 2250
Obsoletes: 2038
Category: Standards Track

D. Hoffman
G. Fernando
Sun Microsystems, Inc.
V. Goyal
Precept Software, Inc.
M. Civanlar
AT&T Labs - Research

January 1998

RTP Payload Format for MPEG1/MPEG2 Video

Status of this Memo

Copyright Notice

Abstract

This memo describes a packetization scheme for MPEG video and audio streams. The scheme proposed can be used to transport such a video or audio flow over the transport protocols supported by RTP. Two approaches are described. The first is designed to support maximum interoperability with MPEG System environments. The second is designed to provide maximum compatibility with other RTP-encapsulated media streams and future conference control work of the IETF.

This memo is a revision of RFC 2038, an Internet standards track protocol. In this revision, the packet loss resilience mechanisms in Section 3.4 were extended to include additional picture header information required for MPEG2. A new section on security considerations for this payload type is added.

1. Introduction

ISO/IEC JTC1/SC29 WG11 (also referred to as the MPEG committee) has defined the MPEG1 standard (ISO/IEC 11172)[1] and the MPEG2 standard (ISO/IEC 13818)[2]. This memo describes a packetization scheme to transport MPEG video and audio streams using the Real-time Transport Protocol (RTP), version 2 [3, 4].

The MPEG1 specification is defined in three parts: System, Video and Audio. It is designed primarily for CD-ROM-based applications, and is optimized for approximately 1.5 Mbits/sec combined data rates. The video and audio portions of the specification describe the basic format of the video or audio stream. These formats define the Elementary Streams (ES). The MPEG1 System specification defines an encapsulation of the ES that contains Presentation Time Stamps (PTS), Decoding Time Stamps and System Clock references, and performs multiplexing of MPEG1 compressed video and audio ES's with user data.

The MPEG2 specification is structured in a similar way. However, it hasn't been restricted only to CD-ROM applications. The MPEG2 System specification defines two system stream formats: the MPEG2 Transport Stream (MTS) and the MPEG2 Program Stream (MPS). The MTS is tailored for communicating or storing one or more programs of MPEG2 compressed data and also other data in relatively error-prone environments. The MPS is tailored for relatively error-free environments.

We seek to achieve interoperability among 4 types of end-systems in the following specification. The 4 types are:

1. Transmitting Interworking Unit (TIU)

   Receives MPEG information from a native MTS system for distribution over packet networks using a native RTP-based system layer (such as an IP-based internetwork). Examples: real-time encoder, MTS satellite link to Internet, video server with MTS-encoded source material.

2. Receiving Interworking Unit (RIU)

   Receives MPEG information in real time from an RTP-based network for forwarding to a native MTS environment. Examples: Internet-based video server to MTS-based cable distribution plant.

3. Transmitting Internet End-System (TAES)

   Transmits MPEG information generated or stored within the internet end-system itself, or received from internet-based computer networks. Example: video server.

4. Receiving Internet End-System (RAES)

   Receives MPEG information over an RTP-based internet for consumption at the internet end-system or forwarding to traditional computer network. Example: desktop PC or workstation viewing training video.

Each of the 2 types of transmitters must work with each of the 2 types of receivers. Because it is probable that the TAES, and certain that the RAES, will be based on existing and planned internet-connected computers, it is highly desirable for the interoperable protocol to be based on RTP.

Because of the range of applications that might employ MPEG streams, we propose to define two payload formats.

Much interest in the MPEG community is in the use of one of the MPEG System encodings, and hence, in Section 2 we propose encapsulations of MPEG1 System streams and MPEG2 Transport and Program Streams with RTP. This profile supports the full semantics of MPEG System and offers basic interoperability among all four end-system types.

When operating only among internet-based end-systems (i.e., TAES and RAES) a payload format that provides greater compatibility with the Internet architecture is desired, deferring some of the system issues to other protocols being defined in the Internet community (such as the MMUSIC WG). In Section 3 we propose an encapsulation of compressed video and audio data (referred to in MPEG documentation as "Elementary Streams" (ES)) complying with either MPEG1 or MPEG2. Here, neither of the System standards of MPEG1 or MPEG2 are utilized. The ES's are directly encapsulated with RTP.

Throughout this specification, we make extensive use of MPEG terminology. The reader should consult the primary MPEG references for definitive descriptions of this terminology.

2. Encapsulation of MPEG System and Transport Streams

Each RTP packet will contain a timestamp derived from the sender's 90KHz clock reference. This clock is synchronized to the system stream Program Clock Reference (PCR) or System Clock Reference (SCR) and represents the target transmission time of the first byte of the

packet payload. The RTP timestamp will not be passed to the MPEG decoder. This use of the timestamp is somewhat different than normally is the case in RTP, in that it is not considered to be the media display or presentation timestamp. The primary purposes of the RTP timestamp will be to estimate and reduce any network-induced jitter and to synchronize relative time drift between the transmitter and receiver.

For MPEG2 Transport Streams the RTP payload will contain an integral number of MPEG transport packets. To avoid end system inefficiencies, data from multiple small MTS packets (normally fixed in size at 188 bytes) are aggregated into a single RTP packet. The number of transport packets contained is computed by dividing RTP payload length by the length of an MTS packet (188).

For MPEG2 Program streams and MPEG1 system streams there are no packetization restrictions; these streams are treated as a packetized stream of bytes.

2.1 RTP header usage

The RTP header fields are used as follows:

Payload Type: Distinct payload types should be assigned for MPEG1 System Streams, MPEG2 Program Streams and MPEG2 Transport Streams. See [4] for payload type assignments.

M bit: Set to 1 whenever the timestamp is discontinuous (such as might happen when a sender switches from one data source to another). This allows the receiver and any intervening RTP mixers or translators that are synchronizing to the flow to ignore the difference between this timestamp and any previous timestamp in their clock phase detectors.

timestamp: 32 bit 90K Hz timestamp representing the target transmission time for the first byte of the packet.

3. Encapsulation of MPEG Elementary Streams

The following ES types may be encapsulated directly in RTP:

(a) MPEG1 Video (ISO/IEC 11172-2) (b) MPEG2 Video (ISO/IEC 13818-2) (c) MPEG1 Audio (ISO/IEC 11172-3) (d) MPEG2 Audio (ISO/IEC 13818-3)

A distinct RTP payload type is assigned to MPEG1/MPEG2 Video and MPEG1/MPEG2 Audio, respectively. Further indication as to whether the data is MPEG1 or MPEG2 need not be provided in the RTP or MPEG-specific headers of this encapsulation, as this information is available in the ES headers.

Presentation Time Stamps (PTS) of 32 bits with an accuracy of 90 kHz shall be carried in the fixed RTP header. All packets that make up a audio or video frame shall have the same time stamp.

3.1 MPEG Video elementary streams

MPEG1 Video can be distinguished from MPEG2 Video at the video sequence header, i.e. for MPEG2 Video a sequence_header() is followed by sequence_extension(). The particular profile and level of MPEG2 Video (MAIN_Profile@MAIN_Level, HIGH_Profile@HIGH_Level, etc) are determined by the profile_and_level_indicator field of the sequence_extension header of MPEG2 Video.

The MPEG bit-stream semantics were designed for relatively error-free environments, and there is significant amount of dependency (both temporal and spatial) within the stream such that loss of some data make other uncorrupted data useless. The format as defined in this encapsulation uses application layer framing information plus additional information in the RTP stream-specific header to allow for certain recovery mechanisms. Appendix 1 suggests several recovery strategies based on the properties of this encapsulation.

Since MPEG pictures can be large, they will normally be fragmented into packets of size less than a typical LAN/WAN MTU. The following fragmentation rules apply:

1. The MPEG Video_Sequence_Header, when present, will always be at the beginning of an RTP payload.
2. An MPEG GOP_header, when present, will always be at the beginning of the RTP payload, or will follow a Video_Sequence_Header.
3. An MPEG Picture_Header, when present, will always be at the beginning of a RTP payload, or will follow a GOP_header.

Each ES header must be completely contained within the packet. Consequently, a minimum RTP payload size of 261 bytes must be supported to contain the largest single header defined in the ES (that is, the extension_data() header containing the quant_matrix_extension()). Otherwise, there are no restrictions on where headers may appear within packet payloads.

In MPEG, each picture is made up of one or more "slices," and a slice is intended to be the unit of recovery from data loss or corruption. An MPEG-compliant decoder will normally advance to the beginning of next slice whenever an error is encountered in the stream. MPEG slice begin and end bits are provided in the encapsulation header to facilitate this.

The beginning of a slice must either be the first data in a packet (after any MPEG ES headers) or must follow after some integral number of slices in a packet. This requirement insures that the beginning of the next slice after one with a missing packet can be found without requiring that the receiver scan the packet contents. Slices may be fragmented across packets as long as all the above rules are met.

An implementation based on this encapsulation assumes that the Video_Sequence_Header is repeated periodically in the MPEG bit-stream. In practice (though not required by MPEG standard) this is used to allow channel switching and to receive and start decoding a continuously relayed MPEG bit-stream at arbitrary points in the media stream. It is suggested that when playing back from an MPEG stream from a file format (where the Video_Sequence_Header may only be represented at the beginning of the stream) that the first Video_Sequence_Header (preceded by an end-of-stream indicator) be saved by the packetizer for periodic injection in to the network stream.

3.2 MPEG Audio elementary streams

MPEG1 Audio can be distinguished from MPEG2 Audio from the MPEG ancillary_data() header. For either MPEG1 or MPEG2 Audio, distinct Presentation Time Stamps may be present for frames which correspond to either 384 samples for Layer-I, or 1152 samples for Layer-II or Layer-III. The actual number of bytes required to represent this number of samples will vary depending on the encoder parameters.

Multiple audio frames may be encapsulated within one RTP packet. In this case, an integral number of audio frames must be contained within the packet and the fragmentation header defined in Section 3.5 shall be set to 0.

Also, if relatively short packets are to be used, one frame may be so large that it may straddle multiple RTP packets. For example, for Layer-II MPEG audio sampled at a rate of 44.1 KHz each frame would represent a time slot of 26.1 msec. At this sampling rate if the compressed bit-rate is 384 kbits/sec (i.e. 48 kBytes/sec) then the average audio frame size would be 1.25 KBytes. If packets were to be 500 Bytes long, then each audio frame would straddle 3 RTP packets.

The audio fragmentation indicator header (See Section 3.5) shall be present for an MPEG1/2 Audio payload type to provide for this fragmentation.

3.3 RTP Fixed Header for MPEG ES encapsulation

The RTP header fields are used as follows:

Payload Type: Distinct payload types should be assigned for video elementary streams and audio elementary streams. See [4] for payload type assignments.

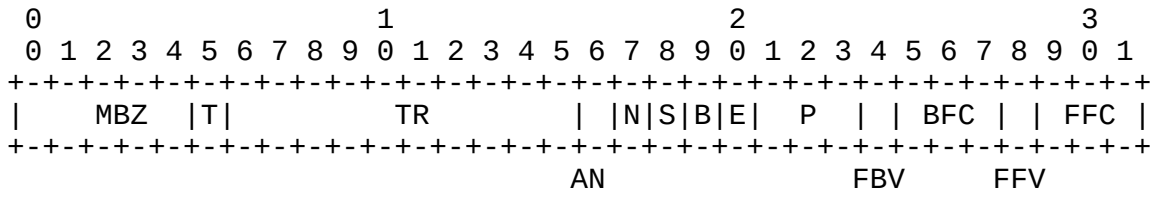
M bit: For video, set to 1 on packet containing MPEG frame end code, 0 otherwise. For audio, set to 1 on first packet of a "talk-spurt," 0 otherwise.

PT: MPEG video or audio stream ID.

timestamp: 32-bit 90K Hz timestamp representing presentation time of MPEG picture or audio frame. Same for all packets that make up a picture or audio frame. May not be monotonically increasing in video stream if B pictures present in stream. For packets that contain only a video sequence and/or GOP header, the timestamp is that of the subsequent picture.

3.4 MPEG Video-specific header

This header shall be attached to each RTP packet after the RTP fixed header.



MBZ: Unused. Must be set to zero in current specification. This space is reserved for future use.

T: MPEG-2 (Two) specific header extension present (1 bit). Set to 1 when the MPEG-2 video-specific header extension (see Section 3.4.1) follows this header. This extension may be needed for improved error resilience; however, its inclusion in an RTP packet is optional. (See Appendix 1.)

TR: Temporal-Reference (10 bits). The temporal reference of the current picture within the current GOP. This value ranges from 0-1023 and is constant for all RTP packets of a given picture.

AN: Active N bit for error resilience (1 bit). Set to 1 when the following bit (N) is used to signal changes in the picture header information for MPEG-2 payloads. It must be set to 0 for MPEG-1 payloads or when N bit is not used.

N: New picture header (1 bit). Used for MPEG-2 payloads when the previous bit (AN) is set to 1. Otherwise, it must be set to zero. Set to 1 when the information contained in the previously transmitted Picture Headers can't be used to reconstruct a header for the current picture. This happens when the current picture is encoded using a different set of parameters than the previous pictures of the same type. The N bit must be constant for all RTP packets that belong to the same picture so that receipt of any packet from a picture allows detecting whether information necessary for reconstruction was contained in that picture (N = 1) or a previous one (N = 0).

S: Sequence-header-present (1 bit). Normally 0 and set to 1 at the occurrence of each MPEG sequence header. Used to detect presence of sequence header in RTP packet.

B: Beginning-of-slice (BS) (1 bit). Set when the start of the packet payload is a slice start code, or when a slice start code is preceded only by one or more of a Video_Sequence_Header, GOP_header and/or Picture_Header.

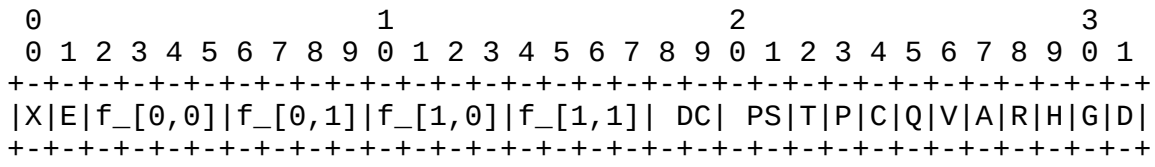
E: End-of-slice (ES) (1 bit). Set when the last byte of the payload is the end of an MPEG slice.

P: Picture-Type (3 bits). I (1), P (2), B (3) or D (4). This value is constant for each RTP packet of a given picture. Value 000B is forbidden and 101B - 111B are reserved to support future extensions to the MPEG ES specification.

FBV: full_pel_backward_vector
BFC: backward_f_code
FFV: full_pel_forward_vector
FFC: forward_f_code
Obtained from the most recent picture header, and are constant for each RTP packet of a given picture. For I frames none of these values are present in the picture header and

they must be set to zero in the RTP header. For P frames only the last two values are present and FBV and BFC must be set to zero in the RTP header. For B frames all the four values are present.

3.4.1 MPEG-2 Video-specific header extension



X: Unused (1 bit). Must be set to zero in current specification. This space is reserved for future use.

E: Extensions present (1 bit). If set to 1, this header extension, including the composite display extension when D = 1, will be followed by one or more of the following extensions: quant matrix extension, picture display extension, picture temporal scalable extension, picture spatial scalable extension and copyright extension.

The first byte of these extensions data gives the length of the extensions in 32 bit words including the length field itself. Zero padding bytes are used at the end if required to align the extensions to 32 bit boundary.

Since they may not be vital in decoding of a picture, the inclusion of any one of these extensions in an RTP packet is optional even when the MPEG-2 video-specific header extension is included in the packet (T = 1). (See Appendix 1.) If present, they should be copied from the corresponding extensions following the most recent MPEG-2 picture coding extension and they remain constant for each RTP packet of a given picture.

The extension start code (32 bits) and the extension start code ID (4 bits) are included. Therefore the extensions are self identifying.

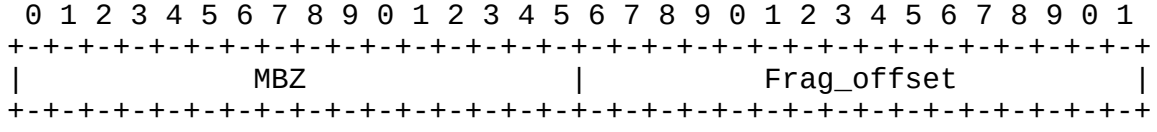
f_[0,0]: forward horizontal f_code (4 bits)
f_[0,1]: forward vertical f_code (4 bits)
f_[1,0]: backward horizontal f_code (4 bits)
f_[1,1]: backward vertical f_code (4 bits)
DC: intra_DC_precision (2 bits)
PS: picture_structure (2 bits)

T: top_field_first (1 bit)
P: frame_predicted_frame_dct (1 bit)
C: concealment_motion_vectors (1 bit)
Q: q_scale type (1 bit)
V: intra_vlc_format (1 bit)
A: alternate scan (1 bit)
R: repeat_first_field (1 bit)
H: chroma_420_type (1 bit)
G: progressive frame (1 bit)
D: composite_display_flag (1 bit). If set to 1, next 32 bits following this one contains 12 zeros followed by 20 bits of composite display information.

These values are copied from the most recent picture coding extension and are constant for each RTP packet of a given picture. Their meanings are as explained in the MPEG-2 standard.

3.5 MPEG Audio-specific header

This header shall be attached to each RTP packet at the start of the payload and after any RTP headers for an MPEG1/2 Audio payload type.



Frag_offset: Byte offset into the audio frame for the data in this packet.

4. Security Considerations

RTP packets using the payload format defined in this specification are subject to the security considerations discussed in the RTP specification [3], and any appropriate RTP profile (for example [4]). This implies that confidentiality of the media streams is achieved by encryption. Because the data compression used with this payload format is applied end-to-end, encryption may be performed after compression 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. However, this encoding does not exhibit any significant non-uniformity.

As with any IP-based protocol, in some circumstances a receiver may be overloaded simply by the receipt of 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 [5] and in multicast routing protocols to allow a receiver to select which sources are allowed to reach it.

A security review of this payload format found no additional considerations beyond those in the RTP specification.

Appendix 1. Error Recovery and Resynchronization Strategies.

The following error recovery and resynchronization strategies are intended to be guidelines only. A compliant receiver is free to employ alternative (or no) strategies.

When initially decoding an RTP-encapsulated MPEG Elementary Stream, the receiver may discard all packets until the Sequence-header-present bit is set to 1. At this point, sufficient state information is contained in the stream to allow processing by an MPEG decoder.

Loss of packets containing the GOP_header and/or Picture_Header are detected by an unexpected change in the Temporal-Reference and Picture-Type values. Consider the following example GOP sequence:

```
In display order: 0B 1B 2I 3B 4B 5P 6B 7B 8P GOP_HDR 0B ...
In stream order:  2I 0B 1B 5P 3B 4B 8P 6B 7B GOP_HDR 2I ...
```

Consider also two counters:

```
ref_pic_temp (Reference Picture (I,P) Temporal Reference)
dep_pic_temp (Dependent Picture (B) Temporal Reference)
```

At each GOP beginning, set these counters to the temporal reference value of the corresponding picture type. For our example GOP sequence, ref_pic_temp = 2 and dep_pic_temp = 0. Keep incrementing BOTH counters by unity with each following picture. Ref_pic_temp should match the temporal references of the I and P frames, and dep_pic_temp should match the temporal references of the B frames.

```
    dep_pic_temp: -  0  1  2  3  4  5  6  7        8  9
In stream order:  2I 0B 1B 5P 3B 4B 8P 6B 7B GOP_H 2I 0B 1B ...
    ref_pic_temp: 2  3  4  5  6  7  8  9  10  ^    11
                  -------------------------  |    ^
                             Match           Drop |
                                                  Mismatch
                                                   in ref_pic_temp
```

The loss of a GOP header can be detected by matching the appropriate counter (based on picture type) to the temporal reference value. A mismatch indicates a lost GOP header. If desired, a GOP header can be re-constructed using a "null" time_code, repeating the closed_gop flag from previous GOP headers, and setting the broken_link flag to 1.

The loss of a Picture_Header can also be detected by a mismatch in the Temporal Reference contained in the RTP packet from the appropriate dep_pic_temp or ref_pic_temp counters at the receiver.

For MPEG-1 payloads, after scanning to the next Beginning-of-slice the Picture_Header is reconstructed from the P, TR, FBV, BFC, FFV and FFC contained in that packet, and from stream-dependent default values.

For MPEG-2, additional information is needed for the reconstruction. This information is provided by the MPEG-2 video specific header extension contained in that packet if the T bit is set to 1, or the Picture Header for the current picture may be available from previous packets belonging to the same picture. The transmitter's strategy for inclusion of the MPEG-2 video specific header extension may depend upon a number of factors. This header may not be needed when:

1. the information has been transmitted a sufficient number of times in previous packets to assure reception with the desired probability, or

2. the information is transmitted over a separate reliable channel, or

3. expected loss rates are low enough that missed frames are not a concern, or

4. conserving bandwidth is more important than error resilience, etc.

If T=1 and E=0, there may be extensions present in the original video bitstream that are not included in the current packet. The transmitter may choose not to include extensions in a packet when they are not necessary for decoding or if one of the cases listed above for not including the MPEG-2 video specific header extension in a packet applies only to the extension data.

If N=0, then the Picture Header from a previous picture of the same type (I,P or B) may be used so long as at least one packet has been received for every intervening picture of the same type and that the N bit was 0 for each of those pictures. This may involve:

1. Saving the relevant picture header information that can be obtained from the MPEG-2 video specific header extension or directly from the video bitstream for each picture type,

2. Keeping validity indicators for this saved information based on the received N bits and lost packets, and,

3. Updating the data whenever a packet with N=1 is received.

If the necessary information is not available from any of these sources, data deletion until a new picture start code is advised.

Any time an RTP packet is lost (as indicated by a gap in the RTP sequence number), the receiver may discard all packets until the Beginning-of-slice bit is set. At this point, sufficient state information is contained in the stream to allow processing by an MPEG decoder starting at the next slice boundary (possibly after reconstruction of the GOP_header and/or Picture_Header as described above).

References

[1] ISO/IEC International Standard 11172; "Coding of moving pictures and associated audio for digital storage media up to about 1,5 Mbits/s", November 1993.

[2] ISO/IEC International Standard 13818; "Generic coding of moving pictures and associated audio information", November 1994.

[3] Schulzrinne, H., Casner, S., Frederick, R., and V. Jacobson, "RTP: A Transport Protocol for Real-Time Applications", RFC 1889, January 1996.

[4] Schulzrinne, H., "RTP Profile for Audio and Video Conferences with Minimal Control", RFC 1890, January 1996.

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

Authors' Addresses

Gerard Fernando
Sun Microsystems, Inc.
Mail-stop UMPK14-305
2550 Garcia Avenue
Mountain View, California 94043-1100
USA

Phone: +1 415-786-6373
EMail: gerard.fernando@eng.sun.com

Vivek Goyal
Precept Software, Inc.
1072 Arastradero Rd,
Palo Alto, CA 94304
USA

Phone: +1 415-845-5200
EMail: goyal@precept.com

Don Hoffman
Sun Microsystems, Inc.
Mail-stop UMPK14-305
2550 Garcia Avenue
Mountain View, California 94043-1100
USA

Phone: +1 503-297-1580
EMail: don.hoffman@eng.sun.com

M. Reha Civanlar
AT&T Labs - Research
100 Schutlz Drive, 3-213
Red Bank, NJ 07701-7033
USA

Phone: +1 732-345-3305
EMail: civanlar@research.att.com

Full Copyright Statement