                   RTP Payload Format for Bundled MPEG

Status of this Memo

Copyright Notice

Abstract

   This document describes a payload type for bundled, MPEG-2 encoded
   video and audio data that may be used with RTP, version 2. Bundling
   has some advantages for this payload type particularly when it is
   used for video-on-demand applications. This payload type may be used
   when its advantages are important enough to sacrifice the modularity
   of having separate audio and video streams.

1. Introduction

   This document describes a bundled packetization scheme for MPEG-2
   encoded audio and video streams using the Real-time Transport
   Protocol (RTP), version 2 [1].

   The MPEG-2 International standard consists of three layers: audio,
   video and systems [2]. The audio and the video layers define the
   syntax and semantics of the corresponding "elementary streams." The
   systems layer supports synchronization and interleaving of multiple
   compressed streams, buffer initialization and management, and time
   identification. RFC 2250 [3] describes packetization techniques to
   transport individual audio and video elementary streams as well as
   the transport stream, which is defined at the system layer, using the
   RTP.

The bundled packetization scheme is needed because it has several
advantages over other schemes for some important applications
including video-on-demand (VOD) where, audio and video are always
used together. Its advantages over independent packetization of
audio and video are:

   1. Uses a single port per "program" (i.e. bundled A/V). This may
   increase the number of streams that can be served e.g., from a VOD
   server. Also, it eliminates the performance hit when two ports are
   used for the separate audio and video streams on the client side.

   2. Provides implicit synchronization of audio and video. This is
   particularly convenient when the A/V data is stored in an
   interleaved format at the server.

   3. Reduces the header overhead. Since using large packets increases
   the effects of losses and delay, audio only packets need to be
   smaller increasing the overhead. An A/V bundled format can provide
   about 1% overall overhead reduction. Considering the high bitrates
   used for MPEG-2 encoded material, e.g. 4 Mbps, the number of bits
   saved, e.g. 40 Kbps, may provide noticeable audio or video quality
   improvement.

   4. May reduce overall receiver buffer size. Audio and video streams
   may experience different delays when transmitted separately. The
   receiver buffers need to be designed for the longest of these
   delays. For example, let's assume that using two buffers, each with
   a size B, is sufficient with probability P when each stream is
   transmitted individually. The probability that the same buffer size
   will be sufficient when both streams need to be received is P times
   the conditional probability of B being sufficient for the second
   stream given that it was sufficient for the first one. This
   conditional probability is, generally, less than one requiring use
   of a larger buffer size to achieve the same probability level.

   5. May help with the control of the overall bandwidth used by an
   A/V program.

And, the advantages over packetization of the transport layer streams
are:

   1. Reduced overhead. It does not contain systems layer information
   which is redundant for the RTP (essentially they address similar
   issues).

2. Easier error recovery. Because of the structured packetization
consistent with the application layer framing (ALF) principle, loss
concealment and error recovery can be made simpler and more
effective.

2. Encapsulation of Bundled MPEG Video and Audio

Video encapsulation follows rules similar to the ones described in
[3] for encapsulation of MPEG elementary streams. Specifically,

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.

In addition to these, it is required that:

4. Each packet must contain an integral number of video slices.

It is the application's responsibility to adjust the slice sizes and
the number of slices put in each RTP packet so that lower level
fragmentation does not occur. This approach simplifies the receivers
while somewhat increasing the complexity of the transmitter's
packetizer. Considering that a slice can be as small as a single
macroblock, it is possible to prevent fragmentation for most of the
cases. If a packet size exceeds the path maximum transmission unit
(path-MTU) [4], this payload type depends on the lower protocol
layers for fragmentation although, this may cause problems with
packet classification for integrated services (e.g. with RSVP).

The video data is followed by a sufficient number of integral audio
frames to cover the duration of the video segment included in a
packet. For example, if the first packet contains three 1/900
seconds long slices of video, and Layer I audio coding is used at a
44.1kHz sampling rate, only one audio frame covering 384/44100
seconds of audio need be included in this packet. Since the length of
this audio frame (8.71 msec.) is longer than that of the video
segment contained in this packet (3.33 msec), the next few packets
may not contain any audio frames until the packet in which the
covered video time extends outside the length of the previously
transmitted audio frames. Alternatively, it is possible, in this
proposal, to repeat the latest audio frame in "no-audio" packets for

    packet loss resilience. Again, it is the application's responsibility
    to adjust the bundled packet size according to the minimum MTU size
    to prevent fragmentation.

2.1. RTP Fixed Header for BMPEG Encapsulation

    The following RTP header fields are used:

      Payload Type: A distinct payload type number, which may be dynamic,
      should be assigned to BMPEG.

      M Bit: Set for packets containing end of a picture.

      timestamp: 32-bit 90 kHz timestamp representing sampling time of
      the MPEG picture. May not be monotonically increasing if B pictures
      are present. Same for all packets belonging to the same picture.
      For packets that contain only a sequence, extension and/or GOP
      header, the timestamp is that of the subsequent picture.

2.2. BMPEG Specific 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
+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
| P |N|MBZ|    Audio Length   | |        Audio Offset          |
+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
                            MBZ
```
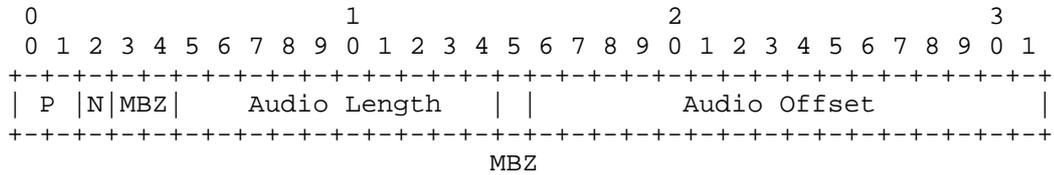
    P: Picture type (2 bits). I (0), P (1), B (2).

    N: Header data changed (1 bit). Set if any part of the video
    sequence, extension, GOP and picture header data is different than
    that of the previously sent headers. It gets reset when all the
    header data gets repeated (see Appendix 1).

    MBZ: Must be zero. Reserved for future use.

    Audio Length: (10 bits) Length of the audio data in this packet in
    bytes. Start of the audio data is found by subtracting "Audio
    Length" from the total length of the received packet.

    Audio Offset: (16 bits) The offset between the start of the audio
    frame and the RTP timestamp for this packet in number of audio
    samples (for multi-channel sources, a set of samples covering all
    channels is counted as one sample for this purpose.)

Audio offset is a signed integer in two's complement form. It allows
a ~ +/- 750 msec offset at 44.1 KHz audio sampling rate. For a very
low video frame rate (e.g., a frame per second), this offset may not
be sufficient and this payload format may not be usable.

If B frames are present, audio frames are not re-ordered along with
video. Instead, they are packetized along with video frames in
their transmission order (e.g., an audio segment packetized with a
video segment corresponding to a P picture may belong to a B
picture, which will be transmitted later and should be rendered at
the same time with this audio segment.) Even though the video
segments are reordered, the audio offset for a particular audio
segment is still relative to the RTP timestamp in the packet
containing that audio segment.

Since a special picture counter, such as the "temporal reference
(TR)" field of [3], is not included in this payload format, lost GOP
headers may not be detected. The only effect of this may be
incorrect decoding of the B pictures immediately following the lost
GOP header for some edited video material.

3. Security Considerations

RTP packets using the payload format defined in this specification
are subject to the security considerations discussed in the RTP
specification [1]. 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.

This payload type does not exhibit any significant non-uniformity in
the receiver side computational complexity for packet processing to
cause a potential denial-of-service threat.

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

Appendix 1. Error Recovery

   Packet losses can be detected from a combination of the sequence
   number and the timestamp fields of the RTP fixed header. The extent
   of the loss can be determined from the timestamp, the slice number
   and the horizontal location of the first slice in the packet. The
   slice number and the horizontal location can be determined from the
   slice header and the first macroblock address increment, which are
   located at fixed bit positions.

   If lost data consists of slices all from the same picture, new data
   following the loss may simply be given to the video decoder which
   will normally repeat missing pixels from a previous picture. The next
   audio frame must be played at the appropriate time determined by the
   timestamp and the audio offset contained in the received packet.
   Appropriate audio frames (e.g., representing background noise) may
   need to be fed to the audio decoder in place of the lost audio frames
   to keep the lip-synch and/or to conceal the effects of the losses.

   If the received new data after a loss is from the next picture (i.e.
   no complete picture loss) and the N bit is not set, previously
   received headers for the particular picture type (determined from the
   P bits) can be given to the video decoder followed by the new data.
   If N is set, data deletion until a new picture start code is
   advisable unless headers are made available to the receiver through
   some other channel.

   If data for more than one picture is lost and headers are not
   available, unless N is zero and 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, resynchronization to a new video
   sequence header is advisable.

   In all cases of heavy packet losses, if the correct headers for the
   missing Pictures are available, they can be given to the video
   decoder and the received data can be used irrespective of the N bit
   value or the number of lost pictures.

Appendix 2. Resynchronization

   As described in [3], use of frequent video sequence headers makes it
   possible to join in a program at arbitrary times. Also, it reduces
   the resynchronization time after severe losses.

References

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

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

    [3] Hoffman, D., Fernando, G., Goyal, V., and M. Civanlar, "RTP
        Payload Format for MPEG1/MPEG2 Video", RFC 2250, January 1998.

    [4] Mogul, J., and S. Deering, "Path MTU Discovery", RFC 1191,
        November 1990.

Authors' Addresses

    M. Reha Civanlar
    AT&T Labs-Research
    100 Schultz Drive
    Red Bank, NJ 07701
    USA

    EMail: civanlar@research.att.com

    Glenn L. Cash
    AT&T Labs-Research
    100 Schultz Drive
    Red Bank, NJ 07701
    USA

    EMail: glenn@research.att.com

    Barry G. Haskell
    AT&T Labs-Research
    100 Schultz Drive
    Red Bank, NJ 07701
    USA

    EMail: bgh@research.att.com

Full Copyright Statement