

Colin Perkins
University College London

RTP Payload Format for Interleaved Media

[draft-ietf-avt-interleaving-01.txt](#)

Status of this Memo

This document is an Internet-Draft and is in full conformance with all provisions of [Section 10 of RFC2026](#).

Internet-Drafts are working documents of the Internet Engineering Task Force (IETF), its areas, and its working groups. Note that other groups may also distribute working documents as Internet-Drafts.

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.

The list of current Internet-Drafts can be accessed at
<http://www.ietf.org/ietf/1id-abstracts.txt>

The list of Internet-Draft Shadow Directories can be accessed at
<http://www.ietf.org/shadow.html>.

Comments are solicited and should be addressed to the author and/or the audio/video transport working group's mailing list rem-conf@es.net.

Abstract

This memo defines an interleaving scheme for RTP streams. This scheme is derived from the RTP payload format for redundant audio data [4] and hence is targetted primarily at streamed audio, although it may be of use in other scenarios.

1 Introduction

The need for loss resilient transport of media streams within RTP has been recognised for a number of years, and various channel coding schemes capable of providing such transport have been proposed. These schemes have, to date, focused on the addition of FEC data to media streams, however FEC schemes are not the only form of error resilience which may be employed. This memo focuses on a transport mechanism for interleaved media, providing an alternative which is of use when bandwidth efficiency is required and latency is not an issue.

Page 1

INTERNET-DRAFT

20 February 1999

2 Discussion

The interleaving process resequences codec frames before transmission, so that originally adjacent frames are separated by a guaranteed distance in the transmitted stream and returned to their original order at the receiver. Interleaving disperses the effect of packet losses. If, for example, frames are 20ms in length and packets 80ms (ie: 4 frames per packet), then the first packet could contain frames 1, 5, 9, 13; the second packet would contain frames 2, 6, 10, 14; and so on. An example is illustrated in figure 1.

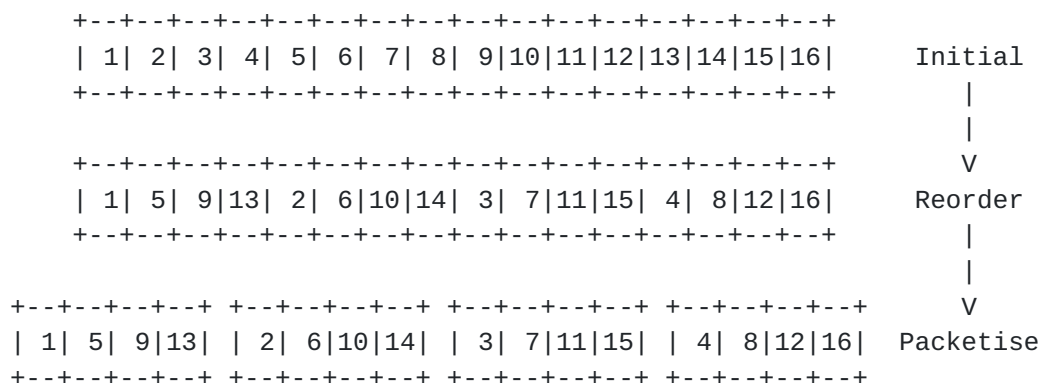


Figure 1: The interleaving process

It can be seen that the loss of a single packet from an interleaved

stream results in multiple small gaps in the reconstructed stream, as opposed to the single large gap which would occur in a non-interleaved stream. The size of the gap is dependent on the degree of interleaving used, and can be made arbitrarily small at the expense of additional latency. In many cases it is easier to reconstruct or repair a stream with such loss patterns, than it is to repair a non-interleaved stream, although this is clearly media and codec dependent.

The obvious disadvantage of interleaving is that it increases latency. The major advantage of interleaving is that it provides increased error resilience yet does not increase the bandwidth requirements of a stream.

If each RTP packet contains a single codec frame, it is a simple matter for the receiver to reconstruct an interleaved stream; frames are decoded in the order specified by the RTP timestamp. It should be noted that the timestamps of these packets will not be monotonically increasing, an effect which will cause RTP header compression [5] to fail for such a stream.

If multiple frames are packed into each RTP packet, the RTP timestamp is not sufficient for the receiver to reconstruct the media stream. It is also necessary to convey the order in which frames are packetised. This information can be communicated explicitly, by timestamping each

Page 2

INTERNET-DRAFT

20 February 1999

frame, or implicitly by informing the receiver of the interleaving function by non-RTP means.

It is more bandwidth efficient to implicitly transport this information, since this allows frames to be packed into RTP packets with no additional headers.

The use of explicit timestamps on each frame allows for the decoder to be unaware of the interleaving function being used, and allows for a common decoder for both redundant and interleaved media. Use of a common payload format also allows for the codec to transparently change, since the payload type of each frame is conveyed.

It is our belief that the benefits of a common decoder model outweigh the bandwidth overhead incurred, hence this document defines a payload format with explicit timestamps on each frame.

3 Payload format definition

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 [3].

The payload format for redundant audio data [4] provides an efficient means by which multiple frames of audio data may be combined within a single RTP packet. Whilst that payload format was defined to allow transport of media specific FEC data, it is also possible to use it to convey interleaved data.

Interleaved frames are packed into an RTP packet using the same payload format as redundant frames. Unlike redundant audio, each frame is sent once only, with the timestamp offset fields in the payload header used to indicate the ordering of interleaved frames.

Frames MUST be packed into packets such that the frame with the earliest timestamp takes the place of the primary encoding, with the other frames taking the place of the redundant encodings. This is because the timestamp offset field in the payload header is unsigned and gives the delay relative to the primary encoding.

Frames SHOULD be packetised such that each packet contains a frame with the maximum timestamp offset required by the interleaver. If this packet would not ordinarily contain a frame with this offset, a dummy frame with this offset and zero length SHOULD be inserted. This requirement is made to allow simple decoder design: it allows the decoder buffering requirement to be identified with the receipt of any packet.

The interleaving function to be used is a function of the encoder only and is not defined here. The decoder does not need to be aware of the interleaving function.

The assignment of an RTP payload type for this payload format is outside the scope of this document, and will not be specified here. It is expected that the RTP profile for a particular class of applications

will assign a payload type for this format, or if that is not done then a payload type in the dynamic range SHOULD be chosen.

4 Example Packet

Assume the interleaving function illustrated in figure 1, using the GSM codec with 20ms frames. The format of the packets would be as illustrated in figure 2.

```

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=2|P|X| CC=0 |M|          PT   |          sequence number   |
+---+---+---+---+---+---+---+---+---+---+---+---+---+---+---+---+
|          timestamp of initial frame          |
+---+---+---+---+---+---+---+---+---+---+---+---+---+---+---+---+
|          synchronization source (SSRC) identifier          |
+---+---+---+---+---+---+---+---+---+---+---+---+---+---+---+---+
|1| block PT=3 |   timestamp offset (=1920)| block length (=33)|
+---+---+---+---+---+---+---+---+---+---+---+---+---+---+---+---+
|1| block PT=3 |   timestamp offset (=1280)| block length (=33)|
+---+---+---+---+---+---+---+---+---+---+---+---+---+---+---+---+
|1| block PT=3 |   timestamp offset (=640) | block length (=33)|
+---+---+---+---+---+---+---+---+---+---+---+---+---+---+---+---+
|0| block PT=3 |          |
+---+---+---+---+---+---+---+---+---+---+---+---+---+---+---+---+
|          |
/          4 frames of GSM encoded data follow          /
|          |
+---+---+---+---+---+---+---+---+---+---+---+---+---+---+---+---+

```

Figure 2: Example interleaved packet

5 Interaction with redundant audio

Whilst the payload format defined in this memo is not the most efficient possible in terms of bandwidth usage for an interleaved stream, the reuse of the payload format for redundant audio data provides a number of advantages which we now describe.

A decoder which can separate frames of data from interleaved/redundant media streams and order them according to both timestamp and quality,

and which select the frame with the highest quality for a particular

Page 4

INTERNET-DRAFT

20 February 1999

time interval should be able to decode both interleaved and redundant media streams with no change.

This allows for dual usage: if low-latency transmission is desired, and some bandwidth overhead is acceptable, then the sender should choose redundant transmission. If latency is not an issue interleaving should be chosen. The decoder can render either stream with no change, resulting in a system suitable for both interactive and non-interactive scenarios.

In addition, packets are sent with predictable sequence numbers and timestamps, such that RTP header compression works correctly with an interleaved stream using this format.

6 Security considerations

There are no additional security considerations beyond those noted for RTP [[1](#)], the RTP profile for audio/video conferences [[2](#)] and the RTP payload format for redundant audio [[4](#)].

7 Acknowledgements

The author wishes to thank Orion Hodson for his helpful comments.

8 Author's addresses

Colin Perkins
Department of Computer Science
University College London
Gower Street

London WC1E 6BT
United Kingdom

Email: c.perkins@cs.ucl.ac.uk

9 References

- [1] H. Schulzrinne, S. Casner, R. Frederick and V. Jacobson, ``RTP: A transport protocol for real-time applications'', [RFC1889](#), January 1996.
- [2] H. Schulzrinne, ``RTP Profile for Audio and Video Conferences with Minimal Control'', [RFC1890](#), January 1996.
- [3] S. Bradner, ``Key words for use in RFCs to indicate requirement levels'', [RFC2119](#), March 1997.

Page 5

INTERNET-DRAFT

20 February 1999

- [4] C. S. Perkins, I. Kouvelas, O. Hodson, V. Hardman, M. Handley, J.-C. Bolot, A. Vega-Garcia and S. Fosse-Parisis, ``RTP Payload for Redundant Audio Data'', [RFC2198](#), November 1997.
- [5] S. Casner and V. Jacobson, ``Compressing IP/UDP/RTP Headers for Low-Speed Serial Links'', [RFC2508](#), February 1999.

Page 6