

Internet-Draft
A/V Transport Payloads Working Group
Intended status: Standards Track
Expires: September 12, 2013

J. Lindsay
H.Foerster
APT Ltd
March 12, 2013

RTP Payload Format for Standard apt-X and Enhanced apt-X Codecs
draft-ietf-payload-rtp-aptx-00

Abstract

This document specifies a scheme for packetizing Standard apt-X, or Enhanced apt-X, encoded audio data into Real-time Transport Protocol (RTP) packets. The document describes a payload format that permits transmission of multiple related audio channels in a single RTP payload, and a means of establishing Standard apt-X and Enhanced apt-X connections through the Session Description Protocol (SDP).

Status of this Memo

This Internet-Draft is submitted to IETF in full conformance with the provisions of [BCP 78](#) and [BCP 79](#).

Comments are solicited and should be addressed to the A/V Transport Payloads working group's mailing list at payload@ietf.org and/or the authors.

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>.

This Internet-Draft will expire on Sept 12, 2013 .

Submission Compliance for Internet-Drafts.

This Internet-Draft is submitted in full conformance with the provisions of [BCP 78](#) and [BCP 79](#).

Copyright and License Notice

Copyright (c) 2012 IETF Trust and the persons identified as the document authors. All rights reserved.

This document is subject to [BCP 78](#) and the IETF Trust's Legal Provisions Relating to IETF Documents (<http://trustee.ietf.org/license-info>) in effect on the date of publication of this document. Please review these documents carefully, as they describe your rights and restrictions with respect to this document. Code Components extracted from this document must include Simplified BSD License text as described in Section 4.e of the Trust Legal Provisions and are provided without warranty as described in the Simplified BSD License.

Table of Contents

1.	Introduction	4
2.	Conventions	5
3.	Standard apt-X and Enhanced apt-X Codecs	6
4.	Payload Format Capabilities	8
4.1.	Use of Forward Error Correction (FEC)	8
5.	Payload Format	9
5.1.	RTP Header Usage	9
5.2.	Payload Structure	10
5.3.	Default Packetization Interval	11
5.4.	Implementation Considerations	11
5.5.	Payload Example	11
6.	Payload Format Parameters	14
6.1.	Media Type Definition	14
6.2.	Mapping to SDP	16
6.2.1.	SDP Usage Example	16
6.2.2.	Offer/Answer Considerations	17
7.	IANA Considerations	18
8.	Security Considerations	19
9.	Acknowledgements	20
10.	References	21
10.1.	Normative References	21
10.2.	Informative References	21
11.	Authors' Addresses	22

1. Introduction

This document specifies the payload format for packetization of audio data, encoded with the Standard apt-X or Enhanced apt-X audio coding algorithms, into the Real-time Transport Protocol (RTP). [[RFC3550](#)].

The document outlines some conventions, a brief description of the operating principles of the audio codecs, and the payload format capabilities. The RTP payload format is detailed and a relevant example of the format is provided. The media type, its mappings to SDP [[RFC4566](#)] and its usage in the SDP offer/answer model are also specified. Finally, some security considerations are outlined.

This document registers a media type (audio/aptx) for the RTP payload format for the Standard apt-X and Enhanced apt-X audio codecs.

2. Conventions

This document uses the normal IETF bit-order representation. Bit fields in figures are read left to right and then down. The leftmost bit in each field is the most significant. The numbering starts from 0 and ascends, where bit 0 will be the most significant.

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 [RFC 2119](#) [[RFC2119](#)].

3. Standard apt-X and Enhanced apt-X Codecs

Standard apt-X and Enhanced apt-X are proprietary audio coding algorithms, licensed by CSR plc and widely deployed in a variety of audio processing equipment. For commercial reasons, the detailed internal operations of these algorithms are not described in standards or reference documents. However, the data interfaces to implementations of these algorithms are very simple, and allow easy RTP packetization of data coded with the algorithms, without a detailed knowledge of the actual coded audio stream syntax.

Both the Standard apt-X and Enhanced apt-X coding algorithms are based on Adaptive Differential Pulse Code Modulation principles. They produce a constant coded bit rate that is scaled according to the sample frequency of the uncoded audio. This constant rate is $1/4$ of the bit rate of the uncoded audio, irrespective of the resolution (number of bits) used to represent an uncoded audio sample. For example, a 1.536 Mbit/s stereo audio stream, composed of 2 channels of 16-bit Pulse Code Modulated (PCM) audio that is sampled at a frequency of 48 kHz, is encoded at 384 kbit/s.

Standard apt-X and Enhanced apt-X do not enforce a coded frame structure, and the coded data forms a continuous coded sample stream with each coded sample capable of regenerating 4 PCM samples when decoded. The Standard apt-X algorithm encodes 4 successive 16-bit PCM samples from each audio channel into a single 16-bit coded sample per audio channel. The Enhanced apt-X algorithm encodes 4 successive 16-bit or 24-bit PCM samples from each audio channel and respectively produces a single 16-bit or 24-bit coded sample per channel. The same RTP packetisation rules apply for each of these algorithmic variations.

- Standard apt-X and Enhanced apt-X coded data streams can optionally carry synchronisation information and an auxiliary data channel within the coded audio data without additional overhead. These mechanisms can, for instance, be used when the IP system is cascaded with another transportation system and the decoder is acting as a simple bridge between the two systems. Since auxiliary data channel and synchronisation information are carried within the coded audio data without additional overhead, RTP payload format rules do not change if they are present. Out-of-band signalling is required however to notify the receiver end when autosync and auxiliary data have been embedded in the apt-X stream.

Embedded auxiliary data is typically used to transport non-audio data, and timecode information for synchronisation with video. The bit rate of the auxiliary data channel is $1/4$ of the sample frequency. For example with a single audio channel encoded at $F_s =$

48kHz, an auxiliary data bit rate of 12 kbit/s can be embedded.

apt-X further provides a means of stereo pairing apt-X channels so that the embedded autosync and auxiliary data channel can be shared across the channel pair. In the case of a 1.536 Mbit/s stereo audio stream, composed of 2 channels of 16-bit PCM audio that is sampled at 48 kHz, a byte of auxiliary data would typically be fed into the Standard or Enhanced apt-X encoder once every 32 uncoded left channel samples. By default apt-X channels pairing is not enabled. Out-of-band signalling is required to notify the receiver when the option is being used.

Standard apt-X and Enhanced apt-X decoders that have not be set up with the correct embedded autosync, auxiliary data and stereo pairing information will playout uncoded PCM samples with a loss of decoding quality. In the case of standard apt-X the loss of quality can be significant.

4. Payload Format Capabilities

This RTP payload format carries an integer number of Standard apt-X or Enhanced apt-X coded audio samples. When multiple related audio channels are being conveyed within the payload, each channel contributes the same integer number of apt-X coded audio samples to the total carried by the payload.

4.1. Use of Forward Error Correction (FEC)

Standard apt-X and Enhanced apt-X do not inherently provide any mechanism for adding redundancy or error-control coding into the coded audio stream. Generic forward error correction schemes for RTP such as [RFC 2198](#) [[RFC2198](#)] and [RFC 5109](#) [[RFC5109](#)] can be used to add redundant information to Standard apt-X and Enhanced apt-X RTP packet streams, making them more resilient to packet losses at the expense of a higher bit rate.

5. Payload Format

The Standard apt-X and Enhanced apt-X algorithms encode 4 successive PCM samples from each audio channel and produce a single compressed sample for each audio channel. The encoder MUST be presented with an integer number S of input audio samples, where S is an arbitrary multiple of 4. The encoder will produce exactly S/4 coded audio samples. Since each coded audio sample is either 16 or 24 bits, the amount of coded audio data produced upon each invocation of the encoding process will be an integer number of bytes. RTP packetization of the encoded data SHALL be on a byte-by-byte basis.

5.1. RTP Header Usage

Utilisation of the Standard apt-X and Enhanced apt-X coding algorithms does not create any special requirements with respect to the contents of the RTP packet header. Other RTP packet header fields are defined as follows.

- o V - As per [[RFC3550](#)]
- o P - As per [[RFC3550](#)]
- o X - As per [[RFC3550](#)]
- o CC - As per [[RFC3550](#)]
- o M - This payload format defines no use for this bit. Senders SHOULD set this bit to zero in each outgoing packet.
- o PT - A dynamic payload type, i.e. one within the range [96..127], MUST be used. [[RFC3551](#)]
- o SN - As per [[RFC3550](#)]
- o Timestamp - As per [[RFC3550](#)]. The RTP timestamp reflects the instant at which the first audio sample in the packet was sampled, that is, the oldest information in the packet.

Header field abbreviations are defined as follows.

V - Version Number

P - Padding

X - Extensions

CC - Count of contributing sources

M - Marker

PT - Payload Type

PS - Payload Structure

5.2. Payload Structure

The RTP payload data for Standard apt-X and Enhanced apt-X MUST be structured as follows.

Standard and Enhanced apt-X coded samples are packed contiguously into payload octets in "network byte order", also known as big-endian order and starting with the most significant bit. Coded samples are packed into the packet in time sequence beginning with the oldest coded sample. An integer number of coded samples MUST be within the same packet.

When multiple channels of Standard and E-APTX coded audio, such as in a stereo program, are multiplexed into a single RTP stream, the coded samples from each channel, at a single sampling instant, are interleaved into a coded sample block according to the following standard audio channel ordering, [RFC3551]. Coded sample blocks are then packed into the packet in time sequence beginning with the oldest coded sample block.

l left
r right
c center
S surround
F front
R rear

channels	description	channel					
		1	2	3	4	5	6
2	stereo	l	r				
3		l	r	c			
4		l	c	r	S		
5		F1	Fr	Fc	S1	Sr	
6		l	lc	c	r	rc	S

For the two-channel encoding example, the sample sequence is (left channel, first sample), (right channel, first sample), (left channel, second sample), (right channel, second sample). Coded Samples for all

channels, belonging to a single coded sampling instant, MUST be contained in the same packet. All channels in the same RTP stream MUST be sampled at the same frequency.

5.3. Default Packetization Interval

The default packetization interval MUST have a duration of 4 ms. When an integer number of coded samples per channel cannot be contained within this 4ms interval, the default packet interval MUST be rounded down to the nearest packet interval that can contain a complete integer set of coded samples. For example when encoding audio with either Standard or Enhanced apt-X, sampled at 11025 Hz, 22050 Hz, or 44100 Hz, the packetization interval MUST be rounded down to 3.99 ms.

The packetization interval sets limits on the end-to-end delay; shorter packets minimize the audio delay through a system at the expense of increased bandwidth while longer packets introduce less header overhead but increase delay and make packet loss more noticeable. A default packet interval of 4 ms maintains an acceptable ratio of payload to header bytes and minimizes the end-to-end delay to allow viable interactive apt-X based applications. All implementations MUST support this default packetization interval.

5.4. Implementation Considerations

An application implementing this payload format MUST understand all the payload parameters that are defined in this specification. Any mapping of these parameters to a signaling protocol MUST support all parameters. Implementation can always decide whether they are capable of communicating based on the entities defined in this specification.

5.5. Payload Example

As an example payload format, consider the transmission of an arbitrary 5.1 audio signal consisting of 6 channels of 24-bit PCM data, sampled at a rate of 48 kHz and packetized on a RTP packet interval of 4ms. The total bit rate before audio coding is $6 * 24 * 48000 = 6.912$ Mbits/s. Applying Enhanced apt-X coding, with a coded sample size of 24 bits, results in a transmitted coded bit rate of 1/4 of the uncoded bit rate, i.e. 1.728 Mbit/s. On packet intervals of 4 ms, packets contain 864 bytes of encoded data that contain 48 Enhanced apt-X coded samples per channel.

For the example format, the diagram below shows how coded samples from each channel are packed into a sample block and how sample blocks 1, 2, and 48 are subsequently packed into the RTP packet.

C:

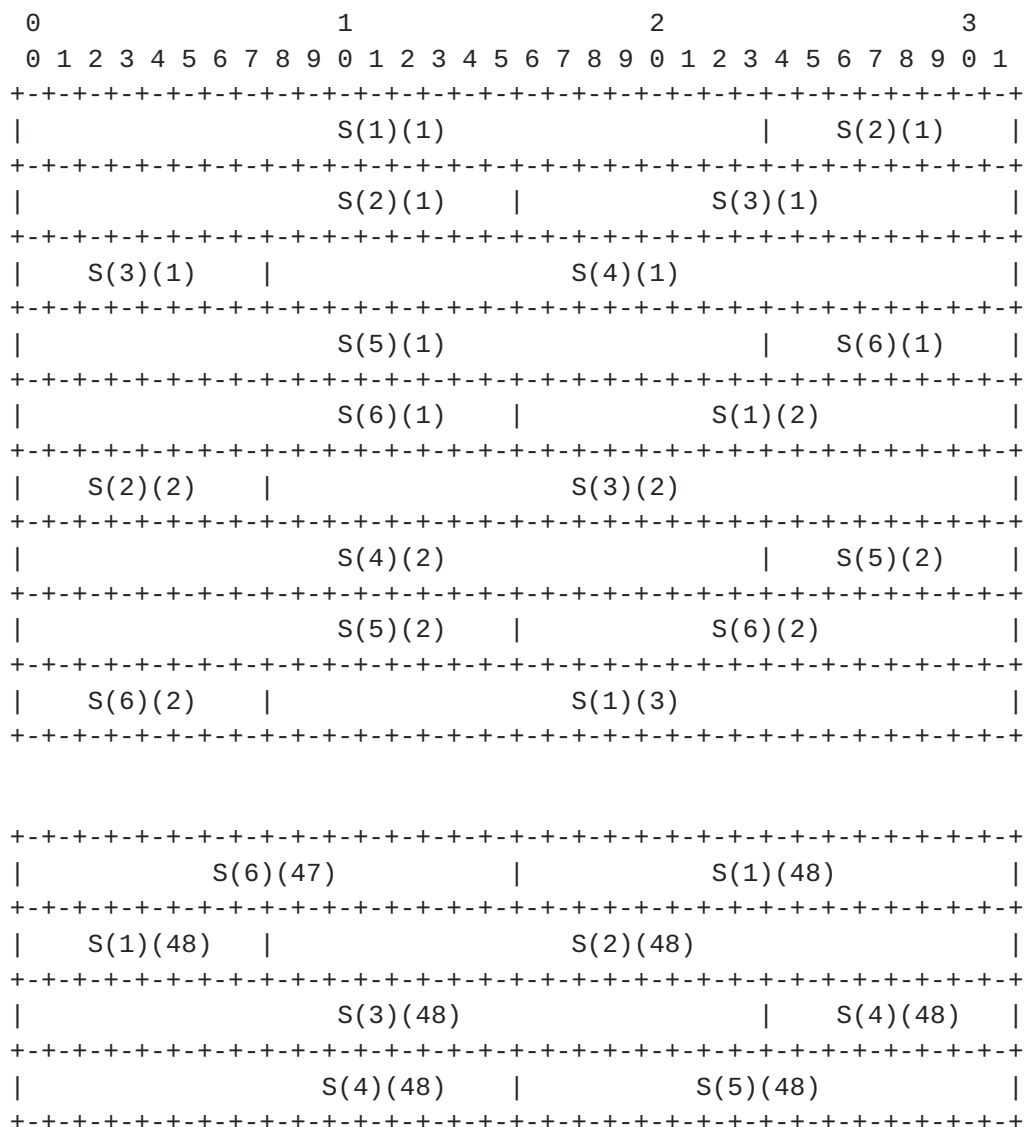
Channel index: Left (l) = 1, left centre (lc) = 2, centre (c) = 3, right (r) = 4, right centre (rc) = 5, surround (S) = 6.

T:

Sample Block time index: The first sample block is 1, the final sample is 48.

S(C)(T):

The Tth sample from channel C



S(5)(48)												S(6)(48)											
+	-	+	-	+	-	+	-	+	-	+	-	+	-	+	-	+	-	+	-	+	-	+	-

For the example format, the diagram below indicates the order that coded bytes are packed into the packet payload in terms of sample byte significance. The following abbreviations are used.

MSB:

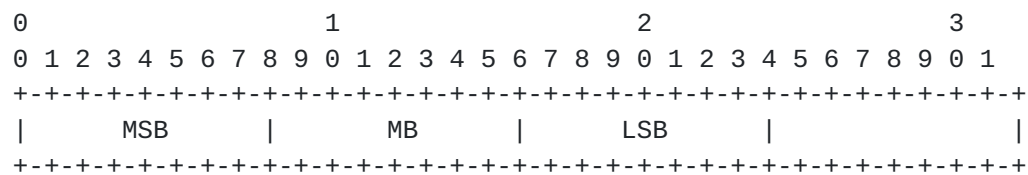
Most Significant Byte of a 24-bit coded sample

MB:

Middle Byte of a 24-bit coded sample

LSB:

Least Significant Byte of a 24-bit coded sample



6. Payload Format Parameters

This RTP payload format is identified using the media type audio/aptx, which is registered in accordance with [RFC 4855](#) [[RFC4855](#)] and using the template of [RFC 4288](#) [[RFC4288](#)]

6.1. Media Type Definition

Registration of media subtype audio/aptx.

MIME media type name: audio

MIME subtype name: aptx

Required parameters:

rate:

RTP timestamp clock rate, which is equal to the sampling rate in Hz. -- RECOMMENDED values for rate are 8000, 11025, 16000, 22050, 24000, 32000, 44100 and 48000 samples per second. Other values are permissible.

channels:

The number of logical audio channels that are present in the audio stream.

variant:

The variant of apt-X (i.e. Standard or Enhanced) that is being used. The following variants can be signalled:

variant=standard

variant=enhanced

bitresolution:

The number of bits used by the algorithm to encode 4 PCM samples. This value MAY only be set to 16 for Standard apt-X and 16 or 24 for Enhanced apt-X.

Optional parameters:

ptime:

The recommended length of time (in milliseconds) represented by the media in a packet. Defaults to 4 ms. See [Section 6 of \[RFC4566\]](#).

maxptime:

The maximum length of time (in milliseconds) that can be

encapsulated in a packet. See [Section 6 of \[RFC4566\]](#).

stereo-channel-pairs:

Defines audio channels that are stereo paired in the stream. See [Section 3](#). Each pair of audio channels is defined as two comma-separated values that correspond to channel numbers in the range 1..channels. Each stereo channel pair is preceded by a '{', and followed by a '}'. Pairs of audio channels are separated by a comma. A channel MUST NOT be paired with more than one other channel. Absence of this parameter signals that each channel has been independently encoded.

embedded-autosync-channels:

Defines channels that carry embedded autosync. embedded-autosync-channels is defined as a list of comma-separated values that correspond to channel numbers in the range 1..channels. When a channel is stereo paired, embedded autosync is shared across channels in the pair. Only the first channel as defined in stereo-channel-pairs MUST be specified in the embedded-autosync-channels list.

embedded-aux-channels:

Defines channels that carry embedded auxiliary data. embedded-aux-channel is defined as a list of comma-separated values that correspond to channel numbers in the range 1..channels. When a channel is stereo paired, embedded auxiliary data is shared across channels in the pair. Only the second channel as defined in stereo-channel-pairs MUST be specified in the embedded-autosync-channels list.

Encoding considerations: This type is only defined for transfer via RTP [[RFC3550](#)].

Security considerations: See [Section 5 of \[RFC4855\]](#) and [Section 4 of \[RFC4856\]](#).

Interoperability considerations: none

Published specification: RFC XXXX

Applications which use this media type: Audio streaming

Additional information: none

Person & email address to contact for further information: John Lindsay email:lindsay@worldcastsystems.com

Intended usage: COMMON

Author/Change controller: John Lindsay

Lindsay & Foerster

Expires September 12, 2013

[Page 15]

6.2. Mapping to SDP

The information carried in the media type specification has a specific mapping to fields in the Session Description Protocol (SDP) [RFC4566] that is commonly used to describe RTP sessions. When SDP is used to describe sessions the media type mappings are as follows.

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

The subtype name ("aptx") goes in SDP "a=rtpmap" as the encoding name.

The parameter "rate" also goes in "a=rtpmap" as clock rate.

The parameter "channels" also goes in "a=rtpmap" as channel count.

The required parameters "variant" and "bitresolution" MUST be included in the SDP "a=fmtp" attribute and MUST follow the delivery-method that applies.

The optional parameters "stereo-channel-pairs", "embedded-autosync-channels", "embedded-aux-channels", and "maxptime" when present, MUST be included in the SDP "a=fmtp" attribute and MUST follow the delivery-method that applies.

The parameter "ptime", when present, goes in a separate SDP attribute field and is signalled as "a=ptime:<value>", where <value> is the number of milliseconds of audio represented by one RTP packet. See [Section 6 of \[RFC4566\]](#).

6.2.1. SDP Usage Examples

Some example SDP session descriptions utilizing apt-X encodings follow. In these examples, long a=fmtp lines are folded to meet the column width constraints of this document.

Example 1: An standard apt-X stream that encodes two independent 44.1kHz 16-bit PCM channels into a 4ms RTP packet.

```
m=audio 5004 RTP/AVP 98
a=rtpmap:98 aptx/44100/2
a=fmtp:98 variant=standard; bitresolution=16;
a=ptime:4
```


Example 2: An enhanced apt-X stream that encodes two 48kHz 24-bit stereo channels into a 4ms RTP packet and that carries both an embedded autosync and auxiliary data channel.

```
m=audio 5004 RTP/AVP 98
a=rtpmap:98 aptx/48000/2
a=fmtp:98 variant=enhanced; bitresolution=24;
stereo-channel-pairs={1,2}; embedded-autosync-channels=1;
embedded-aux-channels=2

a=ptime:4
```

Example 3: An enhanced apt-X stream that encodes six 44.1kHz 24-bit channels into a 6ms RTP packet. Channels 1,2 and 3,4 are stereo pairs. Both stereo pairs carry both an embedded autosync and auxiliary data channel.

```
m=audio 5004 RTP/AVP 98
a=rtpmap:98 aptx/44100/6
a=fmtp:98 variant=enhanced; bitresolution=24;
stereo-channel-pairs={1,2},{3,4}; embedded-autosync-channels=1,3;
embedded-aux-channels=2,4
a=ptime:6
```

6.2.2. Offer/Answer Considerations

The only negotiable parameter is the delivery method. All other parameters are declarative. The offer, as described in [\[RFC3264\]](#), may contain a large number of delivery methods per single fmtp attribute, the answerer MUST remove every delivery method and configuration uri not supported. Apart from this exceptional case, all parameters MUST NOT be altered on answer.

7. IANA Considerations

One media type (audio/aptx) has been defined and needs registration in the media types registry. See [Section 7.1](#)

8. Security Considerations

RTP packets using the payload format defined in this specification are subject to the security considerations discussed in the RTP specification [[RFC3550](#)], and any appropriate RTP profile (for example [[RFC3551](#)]). This implies that confidentiality of the media streams is achieved by encryption. Because the audio coding used with this payload format is applied end-to-end, encryption may be performed after audio coding so there is no conflict between the two operations. A potential denial-of-service threat exists for audio coding 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, the Standard apt-X and Enhanced apt-X audio coding algorithms do 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 [[RFC3376](#)] and in multicast routing protocols to allow a receiver to select which sources are allowed to reach it.

9. Acknowledgements

This specification was facilitated by earlier documents produced by Greg Massey, David Trainer, James Hunter and Derrick Rea along with practical tests carried out by Paul McCambridge of APT Ltd.

10. References

10.1. Normative References

- [RFC2119] Bradner, S., "Key words for use in RFCs to Indicate Requirement Levels", [BCP 14](#), [RFC 2119](#), March 1997.
- [RFC3264] Rosenberg, J. and H. Schulzrinne, "An Offer/Answer Model with Session Description Protocol (SDP)", [RFC 3264](#), June 2002.
- [RFC3550] Schulzrinne, H., Casner, S., Frederick, R., and V. Jacobson, "RTP: A Transport Protocol for Real-Time Applications", STD 64, [RFC 3550](#), July 2003.
- [RFC4566] Handley, M., Jacobson, V., and C. Perkins, "SDP: Session Description Protocol", [RFC 4566](#), July 2006.
- [RFC3551] H. Schulzrinne, "RTP profile for audio and video conferences with minimal control", [RFC 3551](#), July 2003.

10.2. Informative References

- [RFC2198] Perkins, C., Kouvelas, I., Hodson, O., Hardman, V., Handley, M., Bolot, J., Vega-Garcia, A., and S. Fosse-Parisis, "RTP Payload for Redundant Audio Data", [RFC 2198](#), September 1997.
- [RFC3376] Cain, B., Deering, S., Kouvelas, I., Fenner, B., and A. Thyagarajan, "Internet Group Management Protocol, Version 3", [RFC 3376](#), October 2002.
- [RFC4288] Freed, N. and J. Klensin, "Media Type Specifications and Registration Procedures", [BCP 13](#), [RFC 4288](#), December 2005.
- [RFC4855] Casner, S., "Media Type Registration of RTP Payload Formats", [RFC 4855](#), February 2007.
- [RFC4856] Casner, S., "Media Type Registration of Payload Formats in the RTP Profile for Audio and Video Conferences", [RFC 4856](#), February 2007.
- [RFC5109] Li, A., "RTP Payload Format for Generic Forward Error Correction", [RFC 5109](#), December 2007.

11. Authors' Addresses

John Lindsay
APT Ltd
729 Springfield Road
Belfast
Northern Ireland
BT12 7FP
UK

Phone: +44 2890 677200
Email: Lindsay@worldcastsystems.com

Hartmut Foerster
APT Ltd
729 Springfield Road
Belfast
Northern Ireland
BT12 7FP
UK

Phone: +44 2890 677200
Email: foerster@worldcastsystems.com

