Network Working Group
Request for Comments: 3551
Obsoletes: 1890
Category: Standards Track

H. Schulzrinne
Columbia University
S. Casner
Packet Design

July 2003

RTP Profile for Audio and Video Conferences with Minimal Control

Status of this Memo

This document specifies an Internet standards track protocol for the Internet community, and requests discussion and suggestions for improvements. Please refer to the current edition of the "Internet Official Protocol Standards" (STD 1) for the standardization state and status of this protocol. Distribution of this memo is unlimited.

Copyright Notice

Abstract

This document describes a profile called "RTP/AVP" for the use of the real-time transport protocol (RTP), version 2, and the associated control protocol, RTCP, within audio and video multiparticipant conferences with minimal control. It provides interpretations of generic fields within the RTP specification suitable for audio and video conferences. In particular, this document defines a set of default mappings from payload type numbers to encodings.

This document also describes how audio and video data may be carried within RTP. It defines a set of standard encodings and their names when used within RTP. The descriptions provide pointers to reference implementations and the detailed standards. This document is meant as an aid for implementors of audio, video and other real-time multimedia applications.

This memorandum obsoletes RFC 1890. It is mostly backwards-compatible except for functions removed because two interoperable implementations were not found. The additions to RFC 1890 codify existing practice in the use of payload formats under this profile and include new payload formats defined since RFC 1890 was published.

Table of Contents

1. Introduction **4**

1.1 Terminology 4

2. RTP and RTCP Packet Forms and Protocol Behavior **4**

3. Registering Additional Encodings **6**

4. Audio **7**

4.1 Encoding-Independent Rules 7

4.2 Operating Recommendations 9

4.3 Guidelines for Sample-Based Audio Encodings 9

4.4 Guidelines for Frame-Based Audio Encodings 9

4.5 Audio Encodings 10

4.5.1 DVI4 11

4.5.2 G722 12

4.5.3 G723 12

4.5.4 G726-40, G726-32, G726-24, and G726-16 15

4.5.5 G728 16

4.5.6 G729 16

4.5.7 G729D and G729E 18

4.5.8 GSM 20

4.5.9 GSM-EFR 23

4.5.10 L8 23

4.5.11 L16 23

4.5.12 LPC 23

4.5.13 MPA 23

4.5.14 PCMA and PCMU 24

4.5.15 QCELP 24

4.5.16 RED 24

4.5.17 VDVI 24

5. Video **25**

5.1 CelB 26

5.2 JPEG 26

5.3 H261 26

5.4 H263 26

5.5 H263-1998 26

5.6 MPV 26

5.7 MP2T 27

5.8 nv 27

6. Payload Type Definitions **27**

7. RTP over TCP and Similar Byte Stream Protocols **29**

8. Port Assignment **29**

9. Changes from RFC 1890 **30**

10. Security Considerations **32**

11. IANA Considerations **33**

12. References **33**

12.1 Normative References 33

12.2 Informative References 33

13. Current Locations of Related Resources **34**

14. Acknowledgments **36**

15. Intellectual Property Rights Statement **36**

16. Authors' Addresses **37**

17. Full Copyright Statement **38**

1. Introduction

This profile defines aspects of RTP left unspecified in the RTP Version 2 protocol definition (RFC 3550) [1]. This profile is intended for the use within audio and video conferences with minimal session control. In particular, no support for the negotiation of parameters or membership control is provided. The profile is expected to be useful in sessions where no negotiation or membership control are used (e.g., using the static payload types and the membership indications provided by RTCP), but this profile may also be useful in conjunction with a higher-level control protocol.

Use of this profile may be implicit in the use of the appropriate applications; there may be no explicit indication by port number, protocol identifier or the like. Applications such as session directories may use the name for this profile specified in Section 11.

Other profiles may make different choices for the items specified here.

This document also defines a set of encodings and payload formats for audio and video. These payload format descriptions are included here only as a matter of convenience since they are too small to warrant separate documents. Use of these payload formats is NOT REQUIRED to use this profile. Only the binding of some of the payload formats to static payload type numbers in Tables 4 and 5 is normative.

1.1 Terminology

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 [2] and indicate requirement levels for implementations compliant with this RTP profile.

This document defines the term *media type* as dividing encodings of audio and video content into three classes: audio, video and audio/video (interleaved).

2. RTP and RTCP Packet Forms and Protocol Behavior

The section "RTP Profiles and Payload Format Specifications" of RFC 3550 enumerates a number of items that can be specified or modified in a profile. This section addresses these items. Generally, this profile follows the default and/or recommended aspects of the RTP specification.

RTP data header: The standard format of the fixed RTP data header is used (one marker bit).

Payload types: Static payload types are defined in Section 6.

RTP data header additions: No additional fixed fields are appended to the RTP data header.

RTP data header extensions: No RTP header extensions are defined, but applications operating under this profile MAY use such extensions. Thus, applications SHOULD NOT assume that the RTP header X bit is always zero and SHOULD be prepared to ignore the header extension.

If a header extension is defined in the future, that definition MUST specify the contents of the first 16 bits in such a way that multiple different extensions can be identified.

RTCP packet types: No additional RTCP packet types are defined by this profile specification.

RTCP report interval: The suggested constants are to be used for the RTCP report interval calculation. Sessions operating under this profile MAY specify a separate parameter for the RTCP traffic bandwidth rather than using the default fraction of the session bandwidth. The RTCP traffic bandwidth MAY be divided into two separate session parameters for those participants which are active data senders and those which are not. Following the recommendation in the RTP specification [1] that 1/4 of the RTCP bandwidth be dedicated to data senders, the RECOMMENDED default values for these two parameters would be 1.25% and 3.75%, respectively. For a particular session, the RTCP bandwidth for non-data-senders MAY be set to zero when operating on unidirectional links or for sessions that don't require feedback on the quality of reception. The RTCP bandwidth for data senders SHOULD be kept non-zero so that sender reports can still be sent for inter-media synchronization and to identify the source by CNAME. The means by which the one or two session parameters for RTCP bandwidth are specified is beyond the scope of this memo.

SR/RR extension: No extension section is defined for the RTCP SR or RR packet.

SDES use: Applications MAY use any of the SDES items described in the RTP specification. While CNAME information MUST be sent every reporting interval, other items SHOULD only be sent every third reporting interval, with NAME sent seven out of eight times within that slot and the remaining SDES items cyclically taking up the eighth slot, as defined in Section 6.2.2 of the RTP specification. In other words, NAME is sent in RTCP packets 1, 4, 7, 10, 13, 16, 19, while, say, EMAIL is used in RTCP packet 22.

Security: The RTP default security services are also the default under this profile.

String-to-key mapping: No mapping is specified by this profile.

Congestion: RTP and this profile may be used in the context of enhanced network service, for example, through Integrated Services (RFC 1633) [4] or Differentiated Services (RFC 2475) [5], or they may be used with best effort service.

If enhanced service is being 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, RTP receivers SHOULD 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 and 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.

The comparison to TCP cannot be specified exactly, but is intended as an "order-of-magnitude" comparison in timescale and throughput. The timescale on which TCP throughput is measured is the round-trip time of the connection. In essence, this requirement states that it is not acceptable to deploy an application (using RTP or any other transport protocol) on the best-effort Internet which consumes bandwidth arbitrarily and does not compete fairly with TCP within an order of magnitude.

Underlying protocol: The profile specifies the use of RTP over unicast and multicast UDP as well as TCP. (This does not preclude the use of these definitions when RTP is carried by other lower-layer protocols.)

Transport mapping: The standard mapping of RTP and RTCP to transport-level addresses is used.

Encapsulation: This profile leaves to applications the specification of RTP encapsulation in protocols other than UDP.

3. Registering Additional Encodings

This profile lists a set of encodings, each of which is comprised of a particular media data compression or representation plus a payload format for encapsulation within RTP. Some of those payload formats are specified here, while others are specified in separate RFCs. It is expected that additional encodings beyond the set listed here will be created in the future and specified in additional payload format RFCs.

This profile also assigns to each encoding a short name which MAY be used by higher-level control protocols, such as the Session Description Protocol (SDP), RFC 2327 [6], to identify encodings selected for a particular RTP session.

In some contexts it may be useful to refer to these encodings in the form of a MIME content-type. To facilitate this, RFC 3555 [7] provides registrations for all of the encodings names listed here as MIME subtype names under the "audio" and "video" MIME types through the MIME registration procedure as specified in RFC 2048 [8].

Any additional encodings specified for use under this profile (or others) may also be assigned names registered as MIME subtypes with the Internet Assigned Numbers Authority (IANA). This registry provides a means to insure that the names assigned to the additional encodings are kept unique. RFC 3555 specifies the information that is required for the registration of RTP encodings.

In addition to assigning names to encodings, this profile also assigns static RTP payload type numbers to some of them. However, the payload type number space is relatively small and cannot accommodate assignments for all existing and future encodings. During the early stages of RTP development, it was necessary to use statically assigned payload types because no other mechanism had been specified to bind encodings to payload types. It was anticipated that non-RTP means beyond the scope of this memo (such as directory services or invitation protocols) would be specified to establish a dynamic mapping between a payload type and an encoding. Now, mechanisms for defining dynamic payload type bindings have been specified in the Session Description Protocol (SDP) and in other protocols such as ITU-T Recommendation H.323/H.245. These mechanisms associate the registered name of the encoding/payload format, along with any additional required parameters, such as the RTP timestamp clock rate and number of channels, with a payload type number. This association is effective only for the duration of the RTP session in which the dynamic payload type binding is made. This association applies only to the RTP session for which it is made, thus the numbers can be re-used for different encodings in different sessions so the number space limitation is avoided.

This profile reserves payload type numbers in the range 96-127 exclusively for dynamic assignment. Applications SHOULD first use values in this range for dynamic payload types. Those applications which need to define more than 32 dynamic payload types MAY bind codes below 96, in which case it is RECOMMENDED that unassigned payload type numbers be used first. However, the statically assigned payload types are default bindings and MAY be dynamically bound to new encodings if needed. Redefining payload types below 96 may cause incorrect operation if an attempt is made to join a session without obtaining session description information that defines the dynamic payload types.

Dynamic payload types SHOULD NOT be used without a well-defined mechanism to indicate the mapping. Systems that expect to interoperate with others operating under this profile SHOULD NOT make their own assignments of proprietary encodings to particular, fixed payload types.

This specification establishes the policy that no additional static payload types will be assigned beyond the ones defined in this document. Establishing this policy avoids the problem of trying to create a set of criteria for accepting static assignments and encourages the implementation and deployment of the dynamic payload type mechanisms.

The final set of static payload type assignments is provided in Tables 4 and 5.

4. Audio

4.1 Encoding-Independent Rules

Since the ability to suppress silence is one of the primary motivations for using packets to transmit voice, the RTP header carries both a sequence number and a timestamp to allow a receiver to distinguish between lost packets and periods of time when no data was transmitted. Discontiguous transmission (silence suppression) MAY be used with any audio payload format. Receivers MUST assume that senders may suppress silence unless this is restricted by signaling specified elsewhere. (Even if the transmitter does not suppress silence, the receiver should be prepared to handle periods when no data is present since packets may be lost.)

Some payload formats (see Sections 4.5.3 and 4.5.6) define a “silence insertion descriptor” or “comfort noise” frame to specify parameters for artificial noise that may be generated during a period of silence to approximate the background noise at the source. For other payload formats, a generic Comfort Noise (CN) payload format is specified in RFC 3389 [9]. When the CN payload format is used with another payload format, different values in the RTP payload type field distinguish comfort-noise packets from those of the selected payload format.

For applications which send either no packets or occasional comfort-noise packets during silence, the first packet of a talkspurt, that is, the first packet after a silence period during which packets have not been transmitted contiguously, SHOULD be distinguished by setting the marker bit in the RTP data header to one. The marker bit in all other packets is zero. The beginning of a talkspurt MAY be used to adjust the playout delay to reflect changing network delays. Applications without silence suppression MUST set the marker bit to zero.

The RTP clock rate used for generating the RTP timestamp is independent of the number of channels and the encoding; it usually equals the number of sampling periods per second. For N-channel encodings, each sampling period (say, 1/8,000 of a second) generates N samples. (This terminology is standard, but somewhat confusing, as the total number of samples generated per second is then the sampling rate times the channel count.)

If multiple audio channels are used, channels are numbered left-to-right, starting at one. In RTP audio packets, information from lower-numbered channels precedes that from higher-numbered channels. For more than two channels, the convention followed by the AIFF-C audio interchange format SHOULD be followed [3], using the following notation, unless some other convention is specified for a particular encoding or payload format:

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		Fl	Fr	Fc	Sl	Sr	
6		l	lc	c	r	rc	S

> Note: RFC 1890 defined two conventions for the ordering of four audio channels. Since the ordering is indicated implicitly by the number of channels, this was ambiguous. In this revision, the order described as "quadrophonic" has been eliminated to remove the ambiguity. This choice was based on the observation that quadrophonic consumer audio format did not become popular whereas surround-sound subsequently has.

Samples for all channels belonging to a single sampling instant MUST be within the same packet. The interleaving of samples from different channels depends on the encoding. General guidelines are given in Section 4.3 and 4.4.

The sampling frequency SHOULD be drawn from the set: 8,000, 11,025, 16,000, 22,050, 24,000, 32,000, 44,100 and 48,000 Hz. (Older Apple Macintosh computers had a native sample rate of 22,254.54 Hz, which can be converted to 22,050 with acceptable quality by dropping 4 samples in a 20 ms frame.) However, most audio encodings are defined for a more restricted set of sampling frequencies. Receivers SHOULD be prepared to accept multi-channel audio, but MAY choose to only play a single channel.

4.2 Operating Recommendations

The following recommendations are default operating parameters. Applications SHOULD be prepared to handle other values. The ranges given are meant to give guidance to application writers, allowing a set of applications conforming to these guidelines to interoperate without additional negotiation. These guidelines are not intended to restrict operating parameters for applications that can negotiate a set of interoperable parameters, e.g., through a conference control protocol.

For packetized audio, the default packetization interval SHOULD have a duration of 20 ms or one frame, whichever is longer, unless otherwise noted in Table 1 (column "ms/packet"). The packetization interval determines the minimum end-to-end delay; longer packets introduce less header overhead but higher delay and make packet loss more noticeable. For non-interactive applications such as lectures or for links with severe bandwidth constraints, a higher packetization delay MAY be used. A receiver SHOULD accept packets representing between 0 and 200 ms of audio data. (For framed audio encodings, a receiver SHOULD accept packets with a number of frames equal to 200 ms divided by the frame duration, rounded up.) This restriction allows reasonable buffer sizing for the receiver.

4.3 Guidelines for Sample-Based Audio Encodings

In *sample-based* encodings, each audio sample is represented by a fixed number of bits. Within the compressed audio data, codes for individual samples may span octet boundaries. An RTP audio packet may contain any number of audio samples, subject to the constraint that the number of bits per sample times the number of samples per packet yields an integral octet count. *Fractional encodings* produce less than one octet per sample.

The duration of an audio packet is determined by the number of samples in the packet.

For sample-based encodings producing one or more octets per sample, samples from different channels sampled at the same sampling instant SHOULD be packed in consecutive octets. For example, for a two-channel encoding, the octet sequence is (left channel, first sample), (right channel, first sample), (left channel, second sample), (right channel, second sample), ... For multi-octet encodings, octets SHOULD be transmitted in network byte order (i.e., most significant octet first).

The packing of sample-based encodings producing less than one octet per sample is encoding-specific.

The RTP timestamp reflects the instant at which the first sample in the packet was sampled, that is, the oldest information in the packet.

4.4 Guidelines for Frame-Based Audio Encodings

Frame-based encodings encode a fixed-length block of audio into another block of compressed data, typically also of fixed length. For frame-based encodings, the sender MAY choose to combine several such frames into a single RTP packet. The receiver can tell the number of frames contained in an RTP packet, if all the frames have the same length, by dividing the RTP payload length by the audio frame size which is defined as part of the encoding. This does not work when carrying frames of different sizes unless the frame sizes are relatively prime. If not, the frames MUST indicate their size.

For frame-based codecs, the channel order is defined for the whole block. That is, for two-channel audio, right and left samples SHOULD be coded independently, with the encoded frame for the left channel preceding that for the right channel.

All frame-oriented audio codecs SHOULD be able to encode and decode several consecutive frames within a single packet. Since the frame size for the frame-oriented codecs is given, there is no need to use a separate designation for the same encoding, but with different number of frames per packet.

RTP packets SHALL contain a whole number of frames, with frames inserted according to age within a packet, so that the oldest frame (to be played first) occurs immediately after the RTP packet header. The RTP timestamp reflects the instant at which the first sample in the first frame was sampled, that is, the oldest information in the packet.

4.5 Audio Encodings

name of encoding	sample/frame	bits/sample	sampling rate	ms/frame	default ms/packet
DVI4	sample	4	var.		20
G722	sample	8	16,000		20
G723	frame	N/A	8,000	30	30
G726-40	sample	5	8,000		20
G726-32	sample	4	8,000		20
G726-24	sample	3	8,000		20
G726-16	sample	2	8,000		20
G728	frame	N/A	8,000	2.5	20
G729	frame	N/A	8,000	10	20
G729D	frame	N/A	8,000	10	20
G729E	frame	N/A	8,000	10	20
GSM	frame	N/A	8,000	20	20
GSM-EFR	frame	N/A	8,000	20	20
L8	sample	8	var.		20
L16	sample	16	var.		20
LPC	frame	N/A	8,000	20	20
MPA	frame	N/A	var.	var.	
PCMA	sample	8	var.		20
PCMU	sample	8	var.		20
QCELP	frame	N/A	8,000	20	20
VDVI	sample	var.	var.		20

Table 1: Properties of Audio Encodings (N/A: not applicable; var.: variable)

The characteristics of the audio encodings described in this document are shown in Table 1; they are listed in order of their payload type in Table 4. While most audio codecs are only specified for a fixed sampling rate, some sample-based algorithms (indicated by an entry of "var." in the sampling rate column of Table 1) may be used with different sampling rates, resulting in different coded bit rates. When used with a sampling rate other than that for which a static payload type is defined, non-RTP means beyond the scope of this memo MUST be used to define a dynamic payload type and MUST indicate the selected RTP timestamp clock rate, which is usually the same as the sampling rate for audio.

4.5.1 DVI4

DVI4 uses an adaptive delta pulse code modulation (ADPCM) encoding scheme that was specified by the Interactive Multimedia Association (IMA) as the "IMA ADPCM wave type". However, the encoding defined here as DVI4 differs in three respects from the IMA specification:

- The RTP DVI4 header contains the predicted value rather than the first sample value contained the IMA ADPCM block header.
- IMA ADPCM blocks contain an odd number of samples, since the first sample of a block is contained just in the header (uncompressed), followed by an even number of compressed samples. DVI4 has an even number of compressed samples only, using the 'predict' word from the header to decode the first sample.
- For DVI4, the 4-bit samples are packed with the first sample in the four most significant bits and the second sample in the four least significant bits. In the IMA ADPCM codec, the samples are packed in the opposite order.

Each packet contains a single DVI block. This profile only defines the 4-bit-per-sample version, while IMA also specified a 3-bit-per-sample encoding.

The "header" word for each channel has the following structure:

```
int16  predict;  /* predicted value of first sample
                    from the previous block (L16 format) */
u_int8 index;    /* current index into stepsize table */
u_int8 reserved; /* set to zero by sender, ignored by receiver */
```

Each octet following the header contains two 4-bit samples, thus the number of samples per packet MUST be even because there is no means to indicate a partially filled last octet.

Packing of samples for multiple channels is for further study.

The IMA ADPCM algorithm was described in the document *IMA Recommended Practices for Enhancing Digital Audio Compatibility in Multimedia Systems (version 3.0).* However, the Interactive Multimedia Association ceased operations in 1997. Resources for an archived copy of that document and a software implementation of the RTP DVI4 encoding are listed in Section 13.

4.5.2 G722

G722 is specified in ITU-T Recommendation G.722, "7 kHz audio-coding within 64 kbit/s". The G.722 encoder produces a stream of octets, each of which SHALL be octet-aligned in an RTP packet. The first bit transmitted in the G.722 octet, which is the most significant bit of the higher sub-band sample, SHALL correspond to the most significant bit of the octet in the RTP packet.

Even though the actual sampling rate for G.722 audio is 16,000 Hz, the RTP clock rate for the G722 payload format is 8,000 Hz because that value was erroneously assigned in RFC 1890 and must remain unchanged for backward compatibility. The octet rate or sample-pair rate is 8,000 Hz.

4.5.3 G723

G723 is specified in ITU Recommendation G.723.1, "Dual-rate speech coder for multimedia communications transmitting at 5.3 and 6.3 kbit/s". The G.723.1 5.3/6.3 kbit/s codec was defined by the ITU-T as a mandatory codec for ITU-T H.324 GSTN videophone terminal applications. The algorithm has a floating point specification in Annex B to G.723.1, a silence compression algorithm in Annex A to G.723.1 and a scalable channel coding scheme for wireless applications in G.723.1 Annex C.

This Recommendation specifies a coded representation that can be used for compressing the speech signal component of multi-media services at a very low bit rate. Audio is encoded in 30 ms frames, with an additional delay of 7.5 ms due to look-ahead. A G.723.1 frame can be one of three sizes: 24 octets (6.3 kb/s frame), 20 octets (5.3 kb/s frame), or 4 octets. These 4-octet frames are called SID frames (Silence Insertion Descriptor) and are used to specify comfort noise parameters. There is no restriction on how 4, 20, and 24 octet frames are intermixed. The least significant two bits of the first octet in the frame determine the frame size and codec type:

bits	content	octets/frame
00	high-rate speech (6.3 kb/s)	24
01	low-rate speech (5.3 kb/s)	20
10	SID frame	4
11	reserved	

It is possible to switch between the two rates at any 30 ms frame boundary. Both (5.3 kb/s and 6.3 kb/s) rates are a mandatory part of the encoder and decoder. Receivers MUST accept both data rates and MUST accept SID frames unless restriction of these capabilities has been signaled. The MIME registration for G723 in RFC 3555 [7] specifies parameters that MAY be used with MIME or SDP to restrict to a single data rate or to restrict the use of SID frames. This coder was optimized to represent speech with near-toll quality at the above rates using a limited amount of complexity.

The packing of the encoded bit stream into octets and the transmission order of the octets is specified in Rec. G.723.1 and is the same as that produced by the G.723 C code reference implementation. For the 6.3 kb/s data rate, this packing is illustrated as follows, where the header (HDR) bits are always "0 0" as shown in Fig. 1 to indicate operation at 6.3 kb/s, and the Z bit is always set to zero. The diagrams show the bit packing in "network byte order", also known as big-endian order. The bits of each 32-bit word are numbered 0 to 31, with the most significant bit on the left and numbered 0. The octets (bytes) of each word are transmitted most significant octet first. The bits of each data field are numbered in the order of the bit stream representation of the encoding (least significant bit first). The vertical bars indicate the boundaries between field fragments.

```
 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
+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
|    LPC    |HDR|      LPC      |      LPC      |    ACL0   |LPC|
|           |   |               |               |           |   |
|0 0 0 0 0 0|0 0|1 1 1 1 0 0 0 0|2 2 1 1 1 1 1 1|0 0 0 0 0 0|2 2|
|5 4 3 2 1 0|   |3 2 1 0 9 8 7 6|1 0 9 8 7 6 5 4|5 4 3 2 1 0|3 2|
+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
|  ACL2   |ACL|A| GAIN0 |ACL|ACL|    GAIN0      |    GAIN1      |
|         | 1 |C|       | 3 | 2 |               |               |
|0 0 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|
|4 3 2 1 0|1 0|6|3 2 1 0|1 0|6 5|1 0 9 8 7 6 5 4|7 6 5 4 3 2 1 0|
+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
| GAIN2 | GAIN1 |      GAIN2    |     GAIN3     | GRID  | GAIN3 |
|       |       |               |               |       |       |
|0 0 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|1 1 0 0|
|3 2 1 0|1 0 9 8|1 0 9 8 7 6 5 4|7 6 5 4 3 2 1 0|3 2 1 0|1 0 9 8|
+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
|   MSBPOS    |Z|POS|  MSBPOS   |     POS0      |POS|   POS0    |
|             | | 0 |           |               | 1 |           |
|0 0 0 0 0 0 0|0|0 0|1 1 1 0 0 0|0 0 0 0 0 0 0 0|0 0|1 1 1 1 1 1|
|6 5 4 3 2 1 0| |1 0|2 1 0 9 8 7|9 8 7 6 5 4 3 2|1 0|5 4 3 2 1 0|
+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
|     POS1      | POS2  | POS1  |     POS2      | POS3  | POS2  |
|               |       |       |               |       |       |
|0 0 0 0 0 0 0 0|0 0 0 0|1 1 1 1|1 1 0 0 0 0 0 0|0 0 0 0|1 1 1 1|
|9 8 7 6 5 4 3 2|3 2 1 0|3 2 1 0|1 0 9 8 7 6 5 4|3 2 1 0|5 4 3 2|
+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
|     POS3      |   PSIG0   |POS|PSIG2|  PSIG1  |  PSIG3  |PSIG2|
|               |           | 3 |     |         |         |     |
|1 1 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|
|1 0 9 8 7 6 5 4|5 4 3 2 1 0|3 2|2 1 0|4 3 2 1 0|4 3 2 1 0|5 4 3|
+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
```

Figure 1: G.723 (6.3 kb/s) bit packing

For the 5.3 kb/s data rate, the header (HDR) bits are always “0 1”, as shown in Fig. 2, to indicate operation at 5.3 kb/s.

```
 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
+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
|    LPC    |HDR|      LPC      |      LPC      |    ACL0   |LPC|
|           |   |               |               |           |   |
|0 0 0 0 0 0|0 1|1 1 1 1 0 0 0 0|2 2 1 1 1 1 1 1|0 0 0 0 0 0|2 2|
|5 4 3 2 1 0|   |3 2 1 0 9 8 7 6|1 0 9 8 7 6 5 4|5 4 3 2 1 0|3 2|
+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
|  ACL2   |ACL|A| GAIN0 |ACL|ACL|     GAIN0     |     GAIN1     |
|         | 1 |C|       | 3 | 2 |               |               |
|0 0 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|
|4 3 2 1 0|1 0|6|3 2 1 0|1 0|6 5|1 0 9 8 7 6 5 4|7 6 5 4 3 2 1 0|
+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
| GAIN2 | GAIN1 |     GAIN2     |     GAIN3     | GRID  | GAIN3 |
|       |       |               |               |       |       |
|0 0 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|1 1 0 0|
|3 2 1 0|1 0 9 8|1 0 9 8 7 6 5 4|7 6 5 4 3 2 1 0|4 3 2 1|1 0 9 8|
+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
|     POS0      | POS1  | POS0  |     POS1      |     POS2      |
|               |       |       |               |               |
|0 0 0 0 0 0 0 0|0 0 0 0|1 1 0 0|1 1 0 0 0 0 0 0|0 0 0 0 0 0 0 0|
|7 6 5 4 3 2 1 0|3 2 1 0|1 0 9 8|1 0 9 8 7 6 5 4|7 6 5 4 3 2 1 0|
+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
| POS3  | POS2  |     POS3      | PSIG1 | PSIG0 | PSIG3 | PSIG2 |
|       |       |               |       |       |       |       |
|0 0 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|
|3 2 1 0|1 0 9 8|1 0 9 8 7 6 5 4|3 2 1 0|3 2 1 0|3 2 1 0|3 2 1 0|
+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
```

Figure 2: G.723 (5.3 kb/s) bit packing

The packing of G.723.1 SID (silence) frames, which are indicated by the header (HDR) bits having the pattern “1 0”, is depicted in Fig. 3.

```
 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
+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
|    LPC    |HDR|      LPC      |      LPC      |    GAIN   |LPC|
|           |   |               |               |           |   |
|0 0 0 0 0 0|1 0|1 1 1 1 0 0 0 0|2 2 1 1 1 1 1 1|0 0 0 0 0 0|2 2|
|5 4 3 2 1 0|   |3 2 1 0 9 8 7 6|1 0 9 8 7 6 5 4|5 4 3 2 1 0|3 2|
+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
```

Figure 3: G.723 SID mode bit packing

4.5.4 G726-40, G726-32, G726-24, and G726-16

ITU-T Recommendation G.726 describes, among others, the algorithm recommended for conversion of a single 64 kbit/s A-law or mu-law PCM channel encoded at 8,000 samples/sec to and from a 40, 32, 24, or 16 kbit/s channel. The conversion is applied to the PCM stream using an Adaptive Differential Pulse Code Modulation (ADPCM) transcoding technique. The ADPCM representation consists of a series of codewords with a one-to-one correspondence to the samples in the PCM stream. The G726 data rates of 40, 32, 24, and 16 kbit/s have codewords of 5, 4, 3, and 2 bits, respectively.

The 16 and 24 kbit/s encodings do not provide toll quality speech. They are designed for used in overloaded Digital Circuit Multiplication Equipment (DCME). ITU-T G.726 recommends that the 16 and 24 kbit/s encodings should be alternated with higher data rate encodings to provide an average sample size of between 3.5 and 3.7 bits per sample.

The encodings of G.726 are here denoted as G726-40, G726-32, G726-24, and G726-16. Prior to 1990, G721 described the 32 kbit/s ADPCM encoding, and G723 described the 40, 32, and 16 kbit/s encodings. Thus, G726-32 designates the same algorithm as G721 in RFC 1890.

A stream of G726 codewords contains no information on the encoding being used, therefore transitions between G726 encoding types are not permitted within a sequence of packed codewords. Applications MUST determine the encoding type of packed codewords from the RTP payload identifier.

No payload-specific header information SHALL be included as part of the audio data. A stream of G726 codewords MUST be packed into octets as follows: the first codeword is placed into the first octet such that the least significant bit of the codeword aligns with the least significant bit in the octet, the second codeword is then packed so that its least significant bit coincides with the least significant unoccupied bit in the octet. When a complete codeword cannot be placed into an octet, the bits overlapping the octet boundary are placed into the least significant bits of the next octet. Packing MUST end with a completely packed final octet. The number of codewords packed will therefore be a multiple of 8, 2, 8, and 4 for G726-40, G726-32, G726-24, and G726-16, respectively. An example of the packing scheme for G726-32 codewords is as shown, where bit 7 is the least significant bit of the first octet, and bit A3 is the least significant bit of the first codeword:

```
 0                   1
 0 1 2 3 4 5 6 7 8 9 0 1 2 3 4 5
+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-
|B B B B|A A A A|D D D D|C C C C| ...
|0 1 2 3|0 1 2 3|0 1 2 3|0 1 2 3|
+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-
```

An example of the packing scheme for G726-24 codewords follows, where again bit 7 is the least significant bit of the first octet, and bit A2 is the least significant bit of the first codeword:

```
 0                   1                   2
 0 1 2 3 4 5 6 7 8 9 0 1 2 3 4 5 6 7 8 9 0 1 2 3
+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-
|C C|B B B|A A A|F|E E E|D D D|C|H H H|G G G|F F| ...
|1 2|0 1 2|0 1 2|2|0 1 2|0 1 2|0|0 1 2|0 1 2|0 1|
+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-
```

Note that the "little-endian" direction in which samples are packed into octets in the G726-16, -24, -32 and -40 payload formats specified here is consistent with ITU-T Recommendation X.420, but is the opposite of what is specified in ITU-T Recommendation I.366.2 Annex E for ATM AAL2 transport. A second set of RTP payload formats matching the packetization of I.366.2 Annex E and identified by MIME subtypes AAL2-G726-16, -24, -32 and -40 will be specified in a separate document.

4.5.5 G728

G728 is specified in ITU-T Recommendation G.728, "Coding of speech at 16 kbit/s using low-delay code excited linear prediction".

A G.278 encoder translates 5 consecutive audio samples into a 10-bit codebook index, resulting in a bit rate of 16 kb/s for audio sampled at 8,000 samples per second. The group of five consecutive samples is called a vector. Four consecutive vectors, labeled V1 to V4 (where V1 is to be played first by the receiver), build one G.728 frame. The four vectors of 40 bits are packed into 5 octets, labeled B1 through B5. B1 SHALL be placed first in the RTP packet.

Referring to the figure below, the principle for bit order is "maintenance of bit significance". Bits from an older vector are more significant than bits from newer vectors. The MSB of the frame goes to the MSB of B1 and the LSB of the frame goes to LSB of B5.

```
          1         2         3        3
0         0         0         0        9
++++++++++++++++++++++++++++++++++++++++
<---V1---><---V2---><---V3---><---V4---> vectors
<--B1--><--B2--><--B3--><--B4--><--B5--> octets
<------------- frame 1 ---------------->
```

In particular, B1 contains the eight most significant bits of V1, with the MSB of V1 being the MSB of B1. B2 contains the two least significant bits of V1, the more significant of the two in its MSB, and the six most significant bits of V2. B1 SHALL be placed first in the RTP packet and B5 last.

4.5.6 G729

G729 is specified in ITU-T Recommendation G.729, "Coding of speech at 8 kbit/s using conjugate structure-algebraic code excited linear prediction (CS-ACELP)". A reduced-complexity version of the G.729 algorithm is specified in Annex A to Rec. G.729. The speech coding algorithms in the main body of G.729 and in G.729 Annex A are fully interoperable with each other, so there is no need to further distinguish between them. An implementation that signals or accepts use of G729 payload format may implement either G.729 or G.729A unless restricted by additional signaling specified elsewhere related specifically to the encoding rather than the payload format. The G.729 and G.729 Annex A codecs were optimized to represent speech with high quality, where G.729 Annex A trades some speech quality for an approximate 50% complexity reduction [10]. See the next Section (4.5.7) for other data rates added in later G.729 Annexes. For all data rates, the sampling frequency (and RTP timestamp clock rate) is 8,000 Hz.

A voice activity detector (VAD) and comfort noise generator (CNG) algorithm in Annex B of G.729 is RECOMMENDED for digital simultaneous voice and data applications and can be used in conjunction with G.729 or G.729 Annex A. A G.729 or G.729 Annex A frame contains 10 octets, while the G.729 Annex B comfort noise frame occupies 2 octets. Receivers MUST accept comfort noise frames if restriction of their use has not been signaled. The MIME registration for G729 in RFC 3555 [7] specifies a parameter that MAY be used with MIME or SDP to restrict the use of comfort noise frames.

A G729 RTP packet may consist of zero or more G.729 or G.729 Annex A frames, followed by zero or one G.729 Annex B frames. The presence of a comfort noise frame can be deduced from the length of the RTP payload. The default packetization interval is 20 ms (two frames), but in some situations it may be desirable to send 10 ms packets. An example would be a transition from speech to comfort noise in the first 10 ms of the packet. For some applications, a longer packetization interval may be required to reduce the packet rate.

```
  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
 +-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
 |L|      L1     |    L2   |    L3   |       P1      |P|    C1   |
 |0|             |         |         |               |0|         |
 | |0 1 2 3 4 5 6|0 1 2 3 4|0 1 2 3 4|0 1 2 3 4 5 6 7| |0 1 2 3 4|
 +-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
 |       C1      |  S1   | GA1 |  GB1  |    P2   |      C2       |
 |          1 1 1|       |     |       |         |               |
 |5 6 7 8 9 0 1 2|0 1 2 3|0 1 2|0 1 2 3|0 1 2 3 4|0 1 2 3 4 5 6 7|
 +-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
 |   C2    |  S2   | GA2 |  GB2  |
 |    1 1 1|       |     |       |
 |8 9 0 1 2|0 1 2 3|0 1 2|0 1 2 3|
 +-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
```

Figure 4: G.729 and G.729A bit packing

The transmitted parameters of a G.729/G.729A 10-ms frame, consisting of 80 bits, are defined in Recommendation G.729, Table 8/G.729. The mapping of the these parameters is given below in Fig. 4. The diagrams show the bit packing in “network byte order”, also known as big-endian order. The bits of each 32-bit word are numbered 0 to 31, with the most significant bit on the left and numbered 0. The octets (bytes) of each word are transmitted most significant octet first.

The bits of each data field are numbered in the order as produced by the G.729 C code reference implementation.

The packing of the G.729 Annex B comfort noise frame is shown in Fig. 5.

```
 0                   1
 0 1 2 3 4 5 6 7 8 9 0 1 2 3 4 5
+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
|L|  LSF1    |  LSF2 |   GAIN  |R|
|S|          |       |         |E|
|F|          |       |         |S|
|0|0 1 2 3 4|0 1 2 3|0 1 2 3 4|V|    RESV = Reserved (zero)
+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
```

Figure 5: G.729 Annex B bit packing

4.5.7 G729D and G729E

Annexes D and E to ITU-T Recommendation G.729 provide additional data rates. Because the data rate is not signaled in the bitstream, the different data rates are given distinct RTP encoding names which are mapped to distinct payload type numbers. G729D indicates a 6.4 kbit/s coding mode (G.729 Annex D, for momentary reduction in channel capacity), while G729E indicates an 11.8 kbit/s mode (G.729 Annex E, for improved performance with a wide range of narrow-band input signals, e.g., music and background noise). Annex E has two operating modes, backward adaptive and forward adaptive, which are signaled by the first two bits in each frame (the most significant two bits of the first octet).

The voice activity detector (VAD) and comfort noise generator (CNG) algorithm specified in Annex B of G.729 may be used with Annex D and Annex E frames in addition to G.729 and G.729 Annex A frames. The algorithm details for the operation of Annexes D and E with the Annex B CNG are specified in G.729 Annexes F and G. Note that Annexes F and G do not introduce any new encodings. Receivers MUST accept comfort noise frames if restriction of their use has not been signaled. The MIME registrations for G729D and G729E in RFC 3555 [7] specify a parameter that MAY be used with MIME or SDP to restrict the use of comfort noise frames.

For G729D, an RTP packet may consist of zero or more G.729 Annex D frames, followed by zero or one G.729 Annex B frame. Similarly, for G729E, an RTP packet may consist of zero or more G.729 Annex E frames, followed by zero or one G.729 Annex B frame. The presence of a comfort noise frame can be deduced from the length of the RTP payload.

A single RTP packet must contain frames of only one data rate, optionally followed by one comfort noise frame. The data rate may be changed from packet to packet by changing the payload type number. G.729 Annexes D, E and H describe what the encoding and decoding algorithms must do to accommodate a change in data rate.

For G729D, the bits of a G.729 Annex D frame are formatted as shown below in Fig. 6 (cf. Table D.1/G.729). The frame length is 64 bits.

```
 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
+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
|L|      L1     |    L2   |    L3   |       P1      |     C1    |
|0|             |         |         |               |           |
| |0 1 2 3 4 5 6|0 1 2 3 4|0 1 2 3 4|0 1 2 3 4 5 6 7|0 1 2 3 4 5|
+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
| C1  |S1 | GA1 | GB1 |  P2   |        C2       |S2 | GA2 | GB2 |
|     |   |     |     |       |                 |   |     |     |
|6 7 8|0 1|0 1 2|0 1 2|0 1 2 3|0 1 2 3 4 5 6 7 8|0 1|0 1 2|0 1 2|
+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
```

Figure 6: G.729 Annex D bit packing

The net bit rate for the G.729 Annex E algorithm is 11.8 kbit/s and a total of 118 bits are used. Two bits are appended as "don't care" bits to complete an integer number of octets for the frame. For G729E, the bits of a data frame are formatted as shown in the next two diagrams (cf. Table E.1/G.729). The fields for the G729E forward adaptive mode are packed as shown in Fig. 7.

```
 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|L|      L1     |    L2   |    L3   |       P1      |P| C0_1|
|   |0|             |         |         |               |0|     |
|   | |0 1 2 3 4 5 6|0 1 2 3 4|0 1 2 3 4|0 1 2 3 4 5 6 7| |0 1 2|
+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
|       |    C1_1     |     C2_1    |    C3_1     |    C4_1     |
|       |             |             |             |             |
|3 4 5 6|0 1 2 3 4 5 6|0 1 2 3 4 5 6|0 1 2 3 4 5 6|0 1 2 3 4 5 6|
+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
| GA1 |  GB1  |    P2   |   C0_2      |     C1_2    |    C2_2   |
|     |       |         |             |             |           |
|0 1 2|0 1 2 3|0 1 2 3 4|0 1 2 3 4 5 6|0 1 2 3 4 5 6|0 1 2 3 4 5|
+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
| |    C3_2     |     C4_2    | GA2 | GB2   |DC |
| |             |             |     |       |   |
|6|0 1 2 3 4 5 6|0 1 2 3 4 5 6|0 1 2|0 1 2 3|0 1|
+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
```

Figure 7: G.729 Annex E (forward adaptive mode) bit packing

The fields for the G729E backward adaptive mode are packed as shown in Fig. 8.

```
  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 1|       P1      |P|       C0_1              |     C1_1      |
 |   |               |0|                    1 1 1|               |
 |   |0 1 2 3 4 5 6 7|0|0 1 2 3 4 5 6 7 8 9 0 1 2|0 1 2 3 4 5 6 7|
 +-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
 |   | C2_1        | C3_1        | C4_1        |GA1  | GB1   |P2 |
 |   |             |             |             |     |       |   |
 |8 9|0 1 2 3 4 5 6|0 1 2 3 4 5 6|0 1 2 3 4 5 6|0 1 2|0 1 2 3|0 1|
 +-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
 |     |         C0_2            |       C1_2        |    C2_2   |
 |     |                    1 1 1|                   |           |
 |2 3 4|0 1 2 3 4 5 6 7 8 9 0 1 2|0 1 2 3 4 5 6 7 8 9|0 1 2 3 4 5|
 +-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
 | |    C3_2     |     C4_2     | GA2 | GB2   |DC |
 | |             |             |     |       |   |
 |6|0 1 2 3 4 5 6|0 1 2 3 4 5 6|0 1 2|0 1 2 3|0 1|
 +-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
```

Figure 8: G.729 Annex E (backward adaptive mode) bit packing

4.5.8 GSM

GSM (Group Speciale Mobile) denotes the European GSM 06.10 standard for full-rate speech transcoding, ETS 300 961, which is based on RPE/LTP (residual pulse excitation/long term prediction) coding at a rate of 13 kb/s [11, 12, 13]. The text of the standard can be obtained from:

ETSI (European Telecommunications Standards Institute)
ETSI Secretariat: B.P.152
F-06561 Valbonne Cedex
France
Phone: +33 92 94 42 00
Fax: +33 93 65 47 16

Blocks of 160 audio samples are compressed into 33 octets, for an effective data rate of 13,200 b/s.

4.5.8.1 General Packaging Issues

The GSM standard (ETS 300 961) specifies the bit stream produced by the codec, but does not specify how these bits should be packed for transmission. The packetization specified here has subsequently been adopted in ETSI Technical Specification TS 101 318. Some software implementations of the GSM codec use a different packing than that specified here.

field	field name	bits	field	field name	bits
1	LARc[0]	6	39	xmc[22]	3
2	LARc[1]	6	40	xmc[23]	3
3	LARc[2]	5	41	xmc[24]	3
4	LARc[3]	5	42	xmc[25]	3
5	LARc[4]	4	43	Nc[2]	7
6	LARc[5]	4	44	bc[2]	2
7	LARc[6]	3	45	Mc[2]	2
8	LARc[7]	3	46	xmaxc[2]	6
9	Nc[0]	7	47	xmc[26]	3
10	bc[0]	2	48	xmc[27]	3
11	Mc[0]	2	49	xmc[28]	3
12	xmaxc[0]	6	50	xmc[29]	3
13	xmc[0]	3	51	xmc[30]	3
14	xmc[1]	3	52	xmc[31]	3
15	xmc[2]	3	53	xmc[32]	3
16	xmc[3]	3	54	xmc[33]	3
17	xmc[4]	3	55	xmc[34]	3
18	xmc[5]	3	56	xmc[35]	3
19	xmc[6]	3	57	xmc[36]	3
20	xmc[7]	3	58	xmc[37]	3
21	xmc[8]	3	59	xmc[38]	3
22	xmc[9]	3	60	Nc[3]	7
23	xmc[10]	3	61	bc[3]	2
24	xmc[11]	3	62	Mc[3]	2
25	xmc[12]	3	63	xmaxc[3]	6
26	Nc[1]	7	64	xmc[39]	3
27	bc[1]	2	65	xmc[40]	3
28	Mc[1]	2	66	xmc[41]	3
29	xmaxc[1]	6	67	xmc[42]	3
30	xmc[13]	3	68	xmc[43]	3
31	xmc[14]	3	69	xmc[44]	3
32	xmc[15]	3	70	xmc[45]	3
33	xmc[16]	3	71	xmc[46]	3
34	xmc[17]	3	72	xmc[47]	3
35	xmc[18]	3	73	xmc[48]	3
36	xmc[19]	3	74	xmc[49]	3
37	xmc[20]	3	75	xmc[50]	3
38	xmc[21]	3	76	xmc[51]	3

Table 2: Ordering of GSM variables

Octet	Bit 0	Bit 1	Bit 2	Bit 3	Bit 4	Bit 5	Bit 6	Bit 7
0	1	1	0	1	LARc0.0	LARc0.1	LARc0.2	LARc0.3
1	LARc0.4	LARc0.5	LARc1.0	LARc1.1	LARc1.2	LARc1.3	LARc1.4	LARc1.5
2	LARc2.0	LARc2.1	LARc2.2	LARc2.3	LARc2.4	LARc3.0	LARc3.1	LARc3.2
3	LARc3.3	LARc3.4	LARc4.0	LARc4.1	LARc4.2	LARc4.3	LARc5.0	LARc5.1
4	LARc5.2	LARc5.3	LARc6.0	LARc6.1	LARc6.2	LARc7.0	LARc7.1	LARc7.2
5	Nc0.0	Nc0.1	Nc0.2	Nc0.3	Nc0.4	Nc0.5	Nc0.6	bc0.0
6	bc0.1	Mc0.0	Mc0.1	xmaxc00	xmaxc01	xmaxc02	xmaxc03	xmaxc04
7	xmaxc05	xmc0.0	xmc0.1	xmc0.2	xmc1.0	xmc1.1	xmc1.2	xmc2.0
8	xmc2.1	xmc2.2	xmc3.0	xmc3.1	xmc3.2	xmc4.0	xmc4.1	xmc4.2
9	xmc5.0	xmc5.1	xmc5.2	xmc6.0	xmc6.1	xmc6.2	xmc7.0	xmc7.1
10	xmc7.2	xmc8.0	xmc8.1	xmc8.2	xmc9.0	xmc9.1	xmc9.2	xmc10.0
11	xmc10.1	xmc10.2	xmc11.0	xmc11.1	xmc11.2	xmc12.0	xmc12.1	xcm12.2
12	Nc1.0	Nc1.1	Nc1.2	Nc1.3	Nc1.4	Nc1.5	Nc1.6	bc1.0
13	bc1.1	Mc1.0	Mc1.1	xmaxc10	xmaxc11	xmaxc12	xmaxc13	xmaxc14
14	xmax15	xmc13.0	xmc13.1	xmc13.2	xmc14.0	xmc14.1	xmc14.2	xmc15.0
15	xmc15.1	xmc15.2	xmc16.0	xmc16.1	xmc16.2	xmc17.0	xmc17.1	xmc17.2
16	xmc18.0	xmc18.1	xmc18.2	xmc19.0	xmc19.1	xmc19.2	xmc20.0	xmc20.1
17	xmc20.2	xmc21.0	xmc21.1	xmc21.2	xmc22.0	xmc22.1	xmc22.2	xmc23.0
18	xmc23.1	xmc23.2	xmc24.0	xmc24.1	xmc24.2	xmc25.0	xmc25.1	xmc25.2
19	Nc2.0	Nc2.1	Nc2.2	Nc2.3	Nc2.4	Nc2.5	Nc2.6	bc2.0
20	bc2.1	Mc2.0	Mc2.1	xmaxc20	xmaxc21	xmaxc22	xmaxc23	xmaxc24
21	xmaxc25	xmc26.0	xmc26.1	xmc26.2	xmc27.0	xmc27.1	xmc27.2	xmc28.0
22	xmc28.1	xmc28.2	xmc29.0	xmc29.1	xmc29.2	xmc30.0	xmc30.1	xmc30.2
23	xmc31.0	xmc31.1	xmc31.2	xmc32.0	xmc32.1	xmc32.2	xmc33.0	xmc33.1
24	xmc33.2	xmc34.0	xmc34.1	xmc34.2	xmc35.0	xmc35.1	xmc35.2	xmc36.0
25	Xmc36.1	xmc36.2	xmc37.0	xmc37.1	xmc37.2	xmc38.0	xmc38.1	xmc38.2
26	Nc3.0	Nc3.1	Nc3.2	Nc3.3	Nc3.4	Nc3.5	Nc3.6	bc3.0
27	bc3.1	Mc3.0	Mc3.1	xmaxc30	xmaxc31	xmaxc32	xmaxc33	xmaxc34
28	xmaxc35	xmc39.0	xmc39.1	xmc39.2	xmc40.0	xmc40.1	xmc40.2	xmc41.0
29	xmc41.1	xmc41.2	xmc42.0	xmc42.1	xmc42.2	xmc43.0	xmc43.1	xmc43.2
30	xmc44.0	xmc44.1	xmc44.2	xmc45.0	xmc45.1	xmc45.2	xmc46.0	xmc46.1
31	xmc46.2	xmc47.0	xmc47.1	xmc47.2	xmc48.0	xmc48.1	xmc48.2	xmc49.0
32	xmc49.1	xmc49.2	xmc50.0	xmc50.1	xmc50.2	xmc51.0	xmc51.1	xmc51.2

Table 3: GSM payload format

In the GSM packing used by RTP, the bits SHALL be packed beginning from the most significant bit. Every 160 sample GSM frame is coded into one 33 octet (264 bit) buffer. Every such buffer begins with a 4 bit signature (0xD), followed by the MSB encoding of the fields of the frame. The first octet thus contains 1101 in the 4 most significant bits (0-3) and the 4 most significant bits of F1 (0-3) in the 4 least significant bits (4-7). The second octet contains the 2 least significant bits of F1 in bits 0-1, and F2 in bits 2-7, and so on. The order of the fields in the frame is described in Table 2.

4.5.8.2 GSM Variable Names and Numbers

In the RTP encoding we have the bit pattern described in Table 3, where F.i signifies the ith bit of the field F, bit 0 is the most significant bit, and the bits of every octet are numbered from 0 to

7 from most to least significant.

4.5.9 GSM-EFR

GSM-EFR denotes GSM 06.60 enhanced full rate speech transcoding, specified in ETS 300 726 which is available from ETSI at the address given in Section 4.5.8. This codec has a frame length of 244 bits. For transmission in RTP, each codec frame is packed into a 31 octet (248 bit) buffer beginning with a 4-bit signature 0xC in a manner similar to that specified here for the original GSM 06.10 codec. The packing is specified in ETSI Technical Specification TS 101 318.

4.5.10 L8

L8 denotes linear audio data samples, using 8-bits of precision with an offset of 128, that is, the most negative signal is encoded as zero.

4.5.11 L16

L16 denotes uncompressed audio data samples, using 16-bit signed representation with 65,535 equally divided steps between minimum and maximum signal level, ranging from -32,768 to 32,767. The value is represented in two's complement notation and transmitted in network byte order (most significant byte first).

The MIME registration for L16 in RFC 3555 [7] specifies parameters that MAY be used with MIME or SDP to indicate that analog pre-emphasis was applied to the signal before quantization or to indicate that a multiple-channel audio stream follows a different channel ordering convention than is specified in Section 4.1.

4.5.12 LPC

LPC designates an experimental linear predictive encoding contributed by Ron Frederick, which is based on an implementation written by Ron Zuckerman posted to the Usenet group comp.dsp on June 26, 1992. The codec generates 14 octets for every frame. The framesize is set to 20 ms, resulting in a bit rate of 5,600 b/s.

4.5.13 MPA

MPA denotes MPEG-1 or MPEG-2 audio encapsulated as elementary streams. The encoding is defined in ISO standards ISO/IEC 11172-3 and 13818-3. The encapsulation is specified in RFC 2250 [14].

The encoding may be at any of three levels of complexity, called Layer I, II and III. The selected layer as well as the sampling rate and channel count are indicated in the payload. The RTP timestamp clock rate is always 90,000, independent of the sampling rate. MPEG-1 audio supports sampling rates of 32, 44.1, and 48 kHz (ISO/IEC 11172-3, section 1.1; "Scope"). MPEG-2 supports sampling rates of 16, 22.05 and 24 kHz. The number of samples per frame is fixed, but the frame size will vary with the sampling rate and bit rate.

The MIME registration for MPA in RFC 3555 [7] specifies parameters that MAY be used with MIME or SDP to restrict the selection of layer, channel count, sampling rate, and bit rate.

4.5.14 PCMA and PCMU

PCMA and PCMU are specified in ITU-T Recommendation G.711. Audio data is encoded as eight bits per sample, after logarithmic scaling. PCMU denotes mu-law scaling, PCMA A-law scaling. A detailed description is given by Jayant and Noll [15]. Each G.711 octet SHALL be octet-aligned in an RTP packet. The sign bit of each G.711 octet SHALL correspond to the most significant bit of the octet in the RTP packet (i.e., assuming the G.711 samples are handled as octets on the host machine, the sign bit SHALL be the most significant bit of the octet as defined by the host machine format). The 56 kb/s and 48 kb/s modes of G.711 are not applicable to RTP, since PCMA and PCMU MUST always be transmitted as 8-bit samples.

See Section 4.1 regarding silence suppression.

4.5.15 QCELP

The Electronic Industries Association (EIA) & Telecommunications Industry Association (TIA) standard IS-733, "TR45: High Rate Speech Service Option for Wideband Spread Spectrum Communications Systems", defines the QCELP audio compression algorithm for use in wireless CDMA applications. The QCELP CODEC compresses each 20 milliseconds of 8,000 Hz, 16-bit sampled input speech into one of four different size output frames: Rate 1 (266 bits), Rate 1/2 (124 bits), Rate 1/4 (54 bits) or Rate 1/8 (20 bits). For typical speech patterns, this results in an average output of 6.8 kb/s for normal mode and 4.7 kb/s for reduced rate mode. The packetization of the QCELP audio codec is described in [16].

4.5.16 RED

The redundant audio payload format "RED" is specified by RFC 2198 [17]. It defines a means by which multiple redundant copies of an audio packet may be transmitted in a single RTP stream. Each packet in such a stream contains, in addition to the audio data for that packetization interval, a (more heavily compressed) copy of the data from a previous packetization interval. This allows an approximation of the data from lost packets to be recovered upon decoding of a subsequent packet, giving much improved sound quality when compared with silence substitution for lost packets.

4.5.17 VDVI

VDVI is a variable-rate version of DVI4, yielding speech bit rates of between 10 and 25 kb/s. It is specified for single-channel operation only. Samples are packed into octets starting at the most-significant bit. The last octet is padded with 1 bits if the last sample does not fill the last octet. This padding is distinct from the valid codewords. The receiver needs to detect the padding because there is no explicit count of samples in the packet.

It uses the following encoding:

DVI4 codeword	VDVI bit pattern
0	00
1	010
2	1100
3	11100
4	111100
5	1111100
6	11111100
7	11111110
8	10
9	011
10	1101
11	11101
12	111101
13	1111101
14	11111101
15	11111111

5. Video

The following sections describe the video encodings that are defined in this memo and give their abbreviated names used for identification. These video encodings and their payload types are listed in Table 5.

All of these video encodings use an RTP timestamp frequency of 90,000 Hz, the same as the MPEG presentation time stamp frequency. This frequency yields exact integer timestamp increments for the typical 24 (HDTV), 25 (PAL), and 29.97 (NTSC) and 30 Hz (HDTV) frame rates and 50, 59.94 and 60 Hz field rates. While 90 kHz is the RECOMMENDED rate for future video encodings used within this profile, other rates MAY be used. However, it is not sufficient to use the video frame rate (typically between 15 and 30 Hz) because that does not provide adequate resolution for typical synchronization requirements when calculating the RTP timestamp corresponding to the NTP timestamp in an RTCP SR packet. The timestamp resolution MUST also be sufficient for the jitter estimate contained in the receiver reports.

For most of these video encodings, the RTP timestamp encodes the sampling instant of the video image contained in the RTP data packet. If a video image occupies more than one packet, the timestamp is the same on all of those packets. Packets from different video images are distinguished by their different timestamps.

Most of these video encodings also specify that the marker bit of the RTP header SHOULD be set to one in the last packet of a video frame and otherwise set to zero. Thus, it is not necessary to wait for a following packet with a different timestamp to detect that a new frame should be displayed.

5.1 CelB

The CELL-B encoding is a proprietary encoding proposed by Sun Microsystems. The byte stream format is described in RFC 2029 [18].

5.2 JPEG

The encoding is specified in ISO Standards 10918-1 and 10918-2. The RTP payload format is as specified in RFC 2435 [19].

5.3 H261

The encoding is specified in ITU-T Recommendation H.261, "Video codec for audiovisual services at p x 64 kbit/s". The packetization and RTP-specific properties are described in RFC 2032 [20].

5.4 H263

The encoding is specified in the 1996 version of ITU-T Recommendation H.263, "Video coding for low bit rate communication". The packetization and RTP-specific properties are described in RFC 2190 [21]. The H263-1998 payload format is RECOMMENDED over this one for use by new implementations.

5.5 H263-1998

The encoding is specified in the 1998 version of ITU-T Recommendation H.263, "Video coding for low bit rate communication". The packetization and RTP-specific properties are described in RFC 2429 [22]. Because the 1998 version of H.263 is a superset of the 1996 syntax, this payload format can also be used with the 1996 version of H.263, and is RECOMMENDED for this use by new implementations. This payload format does not replace RFC 2190, which continues to be used by existing implementations, and may be required for backward compatibility in new implementations. Implementations using the new features of the 1998 version of H.263 MUST use the payload format described in RFC 2429.

5.6 MPV

MPV designates the use of MPEG-1 and MPEG-2 video encoding elementary streams as specified in ISO Standards ISO/IEC 11172 and 13818-2, respectively. The RTP payload format is as specified in RFC 2250 [14], Section 3.

The MIME registration for MPV in RFC 3555 [7] specifies a parameter that MAY be used with MIME or SDP to restrict the selection of the type of MPEG video.

5.7 MP2T

MP2T designates the use of MPEG-2 transport streams, for either audio or video. The RTP payload format is described in RFC 2250 [14], Section 2.

5.8 nv

The encoding is implemented in the program 'nv', version 4, developed at Xerox PARC by Ron Frederick. Further information is available from the author:

Ron Frederick
Blue Coat Systems Inc.
650 Almanor Avenue
Sunnyvale, CA 94085
United States

EMail: `ronf@bluecoat.com`

6. Payload Type Definitions

Tables 4 and 5 define this profile's static payload type values for the PT field of the RTP data header. In addition, payload type values in the range 96-127 MAY be defined dynamically through a conference control protocol, which is beyond the scope of this document. For example, a session directory could specify that for a given session, payload type 96 indicates PCMU encoding, 8,000 Hz sampling rate, 2 channels. Entries in Tables 4 and 5 with payload type "dyn" have no static payload type assigned and are only used with a dynamic payload type. Payload type 2 was assigned to G721 in RFC 1890 and to its equivalent successor G726-32 in draft versions of this specification, but its use is now deprecated and that static payload type is marked reserved due to conflicting use for the payload formats G726-32 and AAL2-G726-32 (see Section 4.5.4). Payload type 13 indicates the Comfort Noise (CN) payload format specified in RFC 3389 [9]. Payload type 19 is marked "reserved" because some draft versions of this specification assigned that number to an earlier version of the comfort noise payload format. The payload type range 72-76 is marked "reserved" so that RTCP and RTP packets can be reliably distinguished (see Section "Summary of Protocol Constants" of the RTP protocol specification).

The payload types currently defined in this profile are assigned to exactly one of three categories or *media types:* audio only, video only and those combining audio and video. The media types are marked in Tables 4 and 5 as "A", "V" and "AV", respectively. Payload types of different media types SHALL NOT be interleaved or multiplexed within a single RTP session, but multiple RTP sessions MAY be used in parallel to send multiple media types. An RTP source MAY change payload types within the same media type during a session. See the section "Multiplexing RTP Sessions" of RFC 3550 for additional explanation.

Session participants agree through mechanisms beyond the scope of this specification on the set of payload types allowed in a given session. This set MAY, for example, be defined by the capabilities

PT	encoding name	media type	clock rate (Hz)	channels
0	PCMU	A	8,000	1
1	reserved	A		
2	reserved	A		
3	GSM	A	8,000	1
4	G723	A	8,000	1
5	DVI4	A	8,000	1
6	DVI4	A	16,000	1
7	LPC	A	8,000	1
8	PCMA	A	8,000	1
9	G722	A	8,000	1
10	L16	A	44,100	2
11	L16	A	44,100	1
12	QCELP	A	8,000	1
13	CN	A	8,000	1
14	MPA	A	90,000	(see text)
15	G728	A	8,000	1
16	DVI4	A	11,025	1
17	DVI4	A	22,050	1
18	G729	A	8,000	1
19	reserved	A		
20	unassigned	A		
21	unassigned	A		
22	unassigned	A		
23	unassigned	A		
dyn	G726-40	A	8,000	1
dyn	G726-32	A	8,000	1
dyn	G726-24	A	8,000	1
dyn	G726-16	A	8,000	1
dyn	G729D	A	8,000	1
dyn	G729E	A	8,000	1
dyn	GSM-EFR	A	8,000	1
dyn	L8	A	var.	var.
dyn	RED	A		(see text)
dyn	VDVI	A	var.	1

Table 4: Payload types (PT) for audio encodings

PT	encoding name	media type	clock rate (Hz)
24	unassigned	V	
25	CelB	V	90,000
26	JPEG	V	90,000
27	unassigned	V	
28	nv	V	90,000
29	unassigned	V	
30	unassigned	V	
31	H261	V	90,000
32	MPV	V	90,000
33	MP2T	AV	90,000
34	H263	V	90,000
35-71	unassigned	?	
72-76	reserved	N/A	N/A
77-95	unassigned	?	
96-127	dynamic	?	
dyn	H263-1998	V	90,000

Table 5: Payload types (PT) for video and combined encodings

of the applications used, negotiated by a conference control protocol or established by agreement between the human participants.

Audio applications operating under this profile SHOULD, at a minimum, be able to send and/or receive payload types 0 (PCMU) and 5 (DVI4). This allows interoperability without format negotiation and ensures successful negotiation with a conference control protocol.

7. RTP over TCP and Similar Byte Stream Protocols

Under special circumstances, it may be necessary to carry RTP in protocols offering a byte stream abstraction, such as TCP, possibly multiplexed with other data. The application MUST define its own method of delineating RTP and RTCP packets (RTSP [23] provides an example of such an encapsulation specification).

8. Port Assignment

As specified in the RTP protocol definition, RTP data SHOULD be carried on an even UDP port number and the corresponding RTCP packets SHOULD be carried on the next higher (odd) port number.

Applications operating under this profile MAY use any such UDP port pair. For example, the port pair MAY be allocated randomly by a session management program. A single fixed port number pair cannot be required because multiple applications using this profile are likely to run on the same host, and there are some operating systems that do not allow multiple processes to use the same UDP port with different multicast addresses.

However, port numbers 5004 and 5005 have been registered for use with this profile for those applications that choose to use them as the default pair. Applications that operate under multiple profiles MAY use this port pair as an indication to select this profile if they are not subject to the constraint of the previous paragraph. Applications need not have a default and MAY require that the port pair be explicitly specified. The particular port numbers were chosen to lie in the range above 5000 to accommodate port number allocation practice within some versions of the Unix operating system, where port numbers below 1024 can only be used by privileged processes and port numbers between 1024 and 5000 are automatically assigned by the operating system.

9. Changes from RFC 1890

This RFC revises RFC 1890. It is mostly backwards-compatible with RFC 1890 except for functions removed because two interoperable implementations were not found. The additions to RFC 1890 codify existing practice in the use of payload formats under this profile. Since this profile may be used without using any of the payload formats listed here, the addition of new payload formats in this revision does not affect backwards compatibility. The changes are listed below, categorized into functional and non-functional changes.

Functional changes:

- Section 11, "IANA Considerations" was added to specify the registration of the name for this profile. That appendix also references a new Section 3 "Registering Additional Encodings" which establishes a policy that no additional registration of static payload types for this profile will be made beyond those added in this revision and included in Tables 4 and 5. Instead, additional encoding names may be registered as MIME subtypes for binding to dynamic payload types. Non-normative references were added to RFC 3555 [7] where MIME subtypes for all the listed payload formats are registered, some with optional parameters for use of the payload formats.
- Static payload types 4, 16, 17 and 34 were added to incorporate IANA registrations made since the publication of RFC 1890, along with the corresponding payload format descriptions for G723 and H263.
- Following working group discussion, static payload types 12 and 18 were added along with the corresponding payload format descriptions for QCELP and G729. Static payload type 13 was assigned to the Comfort Noise (CN) payload format defined in RFC 3389. Payload type 19 was marked reserved because it had been temporarily allocated to an earlier version of Comfort Noise present in some draft revisions of this document.
- The payload format for G721 was renamed to G726-32 following the ITU-T renumbering, and the payload format description for G726 was expanded to include the -16, -24 and -40 data rates. Because of confusion regarding draft revisions of this document, some implementations of these G726 payload formats packed samples into octets starting with the most significant bit

rather than the least significant bit as specified here. To partially resolve this incompatibility, new payload formats named AAL2-G726-16, -24, -32 and -40 will be specified in a separate document (see note in Section 4.5.4), and use of static payload type 2 is deprecated as explained in Section 6.

- Payload formats G729D and G729E were added following the ITU-T addition of Annexes D and E to Recommendation G.729. Listings were added for payload formats GSM-EFR, RED, and H263-1998 published in other documents subsequent to RFC 1890. These additional payload formats are referenced only by dynamic payload type numbers.
- The descriptions of the payload formats for G722, G728, GSM, VDVI were expanded.
- The payload format for 1016 audio was removed and its static payload type assignment 1 was marked "reserved" because two interoperable implementations were not found.
- Requirements for congestion control were added in Section 2.
- This profile follows the suggestion in the revised RTP spec that RTCP bandwidth may be specified separately from the session bandwidth and separately for active senders and passive receivers.
- The mapping of a user pass-phrase string into an encryption key was deleted from Section 2 because two interoperable implementations were not found.
- The "quadrophonic" sample ordering convention for four-channel audio was removed to eliminate an ambiguity as noted in Section 4.1.

Non-functional changes:

- In Section 4.1, it is now explicitly stated that silence suppression is allowed for all audio payload formats. (This has always been the case and derives from a fundamental aspect of RTP's design and the motivations for packet audio, but was not explicit stated before.) The use of comfort noise is also explained.
- In Section 4.1, the requirement level for setting of the marker bit on the first packet after silence for audio was changed from "is" to "SHOULD be", and clarified that the marker bit is set only when packets are intentionally not sent.
- Similarly, text was added to specify that the marker bit SHOULD be set to one on the last packet of a video frame, and that video frames are distinguished by their timestamps.
- RFC references are added for payload formats published after RFC 1890.
- The security considerations and full copyright sections were added.
- According to Peter Hoddie of Apple, only pre-1994 Macintosh used the 22254.54 rate and none the 11127.27 rate, so the latter was dropped from the discussion of suggested sampling frequencies.

- Table 1 was corrected to move some values from the "ms/packet" column to the "default ms/packet" column where they belonged.
- Since the Interactive Multimedia Association ceased operations, an alternate resource was provided for a referenced IMA document.
- A note has been added for G722 to clarify a discrepancy between the actual sampling rate and the RTP timestamp clock rate.
- Small clarifications of the text have been made in several places, some in response to questions from readers. In particular:
  - A definition for "media type" is given in Section 1.1 to allow the explanation of multiplexing RTP sessions in Section 6 to be more clear regarding the multiplexing of multiple media.
  - The explanation of how to determine the number of audio frames in a packet from the length was expanded.
  - More description of the allocation of bandwidth to SDES items is given.
  - A note was added that the convention for the order of channels specified in Section 4.1 may be overridden by a particular encoding or payload format specification.
  - The terms MUST, SHOULD, MAY, etc. are used as defined in RFC 2119.
- A second author for this document was added.

10. Security Considerations

Implementations using the profile defined in this specification are subject to the security considerations discussed in the RTP specification [1]. This profile does not specify any different security services. The primary function of this profile is to list a set of data compression encodings for audio and video media.

Confidentiality of the media streams is achieved by encryption. Because the data compression used with the payload formats described in this profile 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.

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, source pruning is implemented in IGMPv3 (RFC 3376) [24] and in multicast routing protocols to allow a receiver to select which sources are allowed to reach it.

11. IANA Considerations

The RTP specification establishes a registry of profile names for use by higher-level control protocols, such as the Session Description Protocol (SDP), RFC 2327 [6], to refer to transport methods. This profile registers the name "RTP/AVP".

Section 3 establishes the policy that no additional registration of static RTP payload types for this profile will be made beyond those added in this document revision and included in Tables 4 and 5. IANA may reference that section in declining to accept any additional registration requests. In Tables 4 and 5, note that types 1 and 2 have been marked reserved and the set of "dyn" payload types included has been updated. These changes are explained in Sections 6 and 9.

12. References

12.1 Normative References

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

[2] Bradner, S., "Key Words for Use in RFCs to Indicate Requirement Levels", BCP 14, RFC 2119, March 1997.

[3] Apple Computer, "Audio Interchange File Format AIFF-C", August 1991. (also ftp://ftp.sgi.com/sgi/aiff-c.9.26.91.ps.Z).

12.2 Informative References

[4] Braden, R., Clark, D. and S. Shenker, "Integrated Services in the Internet Architecture: an Overview", RFC 1633, June 1994.

[5] Blake, S., Black, D., Carlson, M., Davies, E., Wang, Z. and W. Weiss, "An Architecture for Differentiated Service", RFC 2475, December 1998.

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

[7] Casner, S. and P. Hoschka, "MIME Type Registration of RTP Payload Types", RFC 3555, July 2003.

[8] Freed, N., Klensin, J. and J. Postel, "Multipurpose Internet Mail Extensions (MIME) Part Four: Registration Procedures", BCP 13, RFC 2048, November 1996.

[9] Zopf, R., "Real-time Transport Protocol (RTP) Payload for Comfort Noise (CN)", RFC 3389, September 2002.

[10] Deleam, D. and J.-P. Petit, "Real-time implementations of the recent ITU-T low bit rate speech coders on the TI TMS320C54X DSP: results, methodology, and applications", in *Proc. of International Conference on Signal Processing, Technology, and Applications (ICSPAT)*, (Boston, Massachusetts), pp. 1656–1660, October 1996.

[11] Mouly, M. and M.-B. Pautet, *The GSM system for mobile communications.* Lassay-les-Chateaux, France: Europe Media Duplication, 1993.

[12] Degener, J., "Digital Speech Compression", *Dr. Dobb's Journal,* December 1994.

[13] Redl, S., Weber, M. and M. Oliphant, *An Introduction to GSM.* Boston: Artech House, 1995.

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

[15] Jayant, N. and P. Noll, *Digital Coding of Waveforms—Principles and Applications to Speech and Video.* Englewood Cliffs, New Jersey: Prentice-Hall, 1984.

[16] McKay, K., "RTP Payload Format for PureVoice(tm) Audio", RFC 2658, August 1999.

[17] Perkins, C., Kouvelas, I., Hodson, O., Hardman, V., Handley, M., Bolot, J.-C., Vega-Garcia, A. and S. Fosse-Parisis, "RTP Payload for Redundant Audio Data", RFC 2198, September 1997.

[18] Speer, M. and D. Hoffman, "RTP Payload Format of Sun's CellB Video Encoding", RFC 2029, October 1996.

[19] Berc, L., Fenner, W., Frederick, R., McCanne, S. and P. Stewart, "RTP Payload Format for JPEG-Compressed Video", RFC 2435, October 1998.

[20] Turletti, T. and C. Huitema, "RTP Payload Format for H.261 Video Streams", RFC 2032, October 1996.

[21] Zhu, C., "RTP Payload Format for H.263 Video Streams", RFC 2190, September 1997.

[22] Bormann, C., Cline, L., Deisher, G., Gardos, T., Maciocco, C., Newell, D., Ott, J., Sullivan, G., Wenger, S. and C. Zhu, "RTP Payload Format for the 1998 Version of ITU-T Rec. H.263 Video (H.263+)", RFC 2429, October 1998.

[23] Schulzrinne, H., Rao, A. and R. Lanphier, "Real Time Streaming Protocol (RTSP)", RFC 2326, April 1998.

[24] Cain, B., Deering, S., Kouvelas, I., Fenner, B. and A. Thyagarajan, "Internet Group Management Protocol, Version 3", RFC 3376, October 2002.

13. Current Locations of Related Resources

Note: Several sections below refer to the *ITU-T Software Tool Library* (STL). It is available from the ITU Sales Service, Place des Nations, CH-1211 Geneve 20, Switzerland (also check http://www.itu.int). The ITU-T STL is covered by a license defined in ITU-T Recommendation G.191, *"Software tools for speech and audio coding standardization".*

DVI4

An archived copy of the document *IMA Recommended Practices for Enhancing Digital Audio Compatibility in Multimedia Systems (version 3.0)*, which describes the IMA ADPCM algorithm, is available at:

http://www.cs.columbia.edu/~hgs/audio/dvi/

An implementation is available from Jack Jansen at

ftp://ftp.cwi.nl/local/pub/audio/adpcm.shar

G722

An implementation of the G.722 algorithm is available as part of the ITU-T STL, described above.

G723

The reference C code implementation defining the G.723.1 algorithm and its Annexes A, B, and C are available as an integral part of Recommendation G.723.1 from the ITU Sales Service, address listed above. Both the algorithm and C code are covered by a specific license. The ITU-T Secretariat should be contacted to obtain such licensing information.

G726

G726 is specified in the ITU-T Recommendation G.726, "40, 32, 24, and 16 kb/s Adaptive Differential Pulse Code Modulation (ADPCM)". An implementation of the G.726 algorithm is available as part of the ITU-T STL, described above.

G729

The reference C code implementation defining the G.729 algorithm and its Annexes A through I are available as an integral part of Recommendation G.729 from the ITU Sales Service, listed above. Annex I contains the integrated C source code for all G.729 operating modes. The G.729 algorithm and associated C code are covered by a specific license. The contact information for obtaining the license is available from the ITU-T Secretariat.

GSM

A reference implementation was written by Carsten Bormann and Jutta Degener (then at TU Berlin, Germany). It is available at

http://www.dmn.tzi.org/software/gsm/

Although the RPE-LTP algorithm is not an ITU-T standard, there is a C code implementation of the RPE-LTP algorithm available as part of the ITU-T STL. The STL implementation is an adaptation of the TU Berlin version.

LPC

An implementation is available at

ftp://parcftp.xerox.com/pub/net-research/lpc.tar.Z

PCMU, PCMA

An implementation of these algorithms is available as part of the ITU-T STL, described above.

14. Acknowledgments

The comments and careful review of Simao Campos, Richard Cox and AVT Working Group participants are gratefully acknowledged. The GSM description was adopted from the *IMTC Voice over IP Forum Service Interoperability Implementation Agreement* (January 1997). Fred Burg and Terry Lyons helped with the G.729 description.

15. Intellectual Property Rights Statement

The IETF takes no position regarding the validity or scope of any intellectual property 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; neither does it represent that it has made any effort to identify any such rights. Information on the IETF's procedures with respect to rights in standards-track and standards-related documentation can be found in BCP-11. Copies of claims of rights made available for publication 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 implementors or users of this specification can be obtained from the IETF Secretariat.

The IETF invites any interested party to bring to its attention any copyrights, patents or patent applications, or other proprietary rights which may cover technology that may be required to practice this standard. Please address the information to the IETF Executive Director.

16. Authors' Addresses

Henning Schulzrinne
Department of Computer Science
Columbia University
1214 Amsterdam Avenue
New York, NY 10027
United States

EMail: schulzrinne@cs.columbia.edu

Stephen L. Casner
Packet Design
3400 Hillview Avenue, Building 3
Palo Alto, CA 94304
United States

EMail: casner@acm.org

17. Full Copyright Statement

Acknowledgement

Funding for the RFC Editor function is currently provided by the Internet Society.