Internet Engineering Task Force
INTERNET-DRAFT
draft-ietf-avt-profile-new-00.ps

AVT WG
Schulzrinne
Columbia U.

March 27, 1997
Expires: September 9, 1997

RTP Profile for Audio and Video Conferences with Minimal Control

Status of this Memo

This document is an Internet-Draft. 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".

To learn the current status of any Internet-Draft, please check the "1id-abstracts.txt" listing contained in the Internet-Drafts Shadow Directories on ftp.is.co.za (Africa), nic.nordu.net (Europe), munnari.oz.au (Pacific Rim), ds.internic.net (US East Coast), or ftp.isi.edu (US West Coast).

Distribution of this document is unlimited.

Abstract

This memo 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.

The 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. However, the encoding definitions are independent of the particular transport mechanism used. 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.

Changes

This draft revises RFC 1890. It is fully backwards-compatible with RFC 1890 and codifies existing practice. It is intended that this draft form the basis of a new RFC to obsolete RFC 1890 as it moves to Draft Standard..

Besides wording clarifications and filling in RFC numbers for payload type definitions, this draft adds payload types 4, 13, 16, 17, 18 and 34. The PostScript version of this draft contains change bars.

Note to RFC editor: This section is to be removed before publication as an RFC. All RFC TBD should be filled in with the number of the RTP specification RFC submitted for DS status.

1 Introduction

This profile defines aspects of RTP left unspecified in the RTP Version 2 protocol definition (RFC XXXX). 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 occurs by use of the appropriate applications; there is no explicit indication by port number, protocol identifier or the like. Applications such as session directories should refer to this profile as "RTP/AVP".

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

This document also defines a set of payload formats for audio.

This draft 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 Specification" of RFC TBD 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.

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 is sent every reporting interval, other items should be sent only 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: A user-provided string ("pass phrase") is hashed with the MD5 algorithm to a 16-octet digest. An n-bit key is extracted from the digest by taking the first n bits from the digest. If several keys are needed with a total length of 128 bits or less (as for triple DES), they are extracted in order from that digest. The octet ordering is specified in RFC 1423, Section 2.2. (Note that some DES implementations require that the 56-bit key be expanded into 8 octets by inserting an odd parity bit in the most significant bit of the octet to go with each 7 bits of the key.)

It is suggested that pass phrases are restricted to ASCII letters, digits, the hyphen, and white space to reduce the the chance of transcription errors when conveying keys by phone, fax, telex or email.

The pass phrase may be preceded by a specification of the encryption algorithm. Any characters up to the first slash (ASCII 0x2f) are taken as the name of the encryption algorithm. The encryption format specifiers should be drawn from RFC 1423 or any additional identifiers registered with IANA. If no slash is present, DES-CBC is assumed as default. The encryption algorithm specifier is case sensitive.

The pass phrase typed by the user is transformed to a canonical form before applying the hash algorithm. For that purpose, we define 'white space' to be the ASCII space, formfeed, newline, carriage return, tab, or vertical tab as well as all characters contained in the Unicode space characters table. The transformation consists of the following steps: (1) convert the input string to the ISO 10646 character set, using the UTF-8 encoding as specified in Annex P to ISO/IEC 10646-1:1993 (ASCII characters require no mapping, but ISO 8859-1 characters do); (2) remove leading and trailing white space characters; (3) replace one or more contiguous white space characters by a single space (ASCII or UTF-8 0x20); (4) convert all letters to lower case and replace sequences of characters and non-spacing accents with a single character, where possible. A minimum length of 16 key characters (after applying the transformation) should be enforced by the application, while applications must allow up to 256 characters of input.

Underlying protocol: The profile specifies the use of RTP over unicast and multicast UDP. (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: No encapsulation of RTP packets is specified.

3 Registering Payload Types

This profile defines a set of standard encodings and their payload types when used within RTP. Other encodings and their payload types are to be registered with the Internet Assigned Numbers Authority (IANA). When registering a new encoding/payload type, the following information should be provided:

- name and description of encoding, in particular the RTP timestamp clock rate; the names defined here are 3 or 4 characters long to allow a compact representation if needed;
- indication of who has change control over the encoding (for example, ISO, CCITT/ITU, other international standardization bodies, a consortium or a particular company or group of companies);
- any operating parameters or profiles;
- a reference to a further description, if available, for example (in order of preference) an RFC, a published paper, a patent filing, a technical report, documented source code or a computer manual;
- for proprietary encodings, contact information (postal and email address);
- the payload type value for this profile, if necessary (see below).

Note that not all encodings to be used by RTP need to be assigned a static payload type. Non-RTP means beyond the scope of this memo (such as directory services or invitation protocols) may be used to establish a dynamic mapping between a payload type drawn from the range $96 - 127$ and an encoding. For implementor convenience, this profile contains descriptions of encodings which do not currently have a static payload type assigned to them.

Note that 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 assign proprietary encodings to particular, fixed payload types in the range reserved for dynamic payload types.

The available payload type space is relatively small. Thus, new static payload types are assigned only if the following conditions are met:

- The encoding is of interest to the Internet community at large.
- It offers benefits compared to existing encodings and/or is required for interoperation with existing, widely deployed conferencing or multimedia systems.
- The description is sufficient to build a decoder.

The four-character encoding names are those those by the Session Description Protocol (SDP) (RFC XXXX) [**?**].

4 Audio

4.1 Encoding-Independent Rules

For applications which send no packets during silence, the first packet of a talkspurt, that is, the first packet after a silence period, is distinguished by setting the marker bit in the RTP data header. The beginning of a talkspurt may be used to adjust the playout delay to reflect changing network delays. Applications without silence suppression set the bit to zero.

The RTP clock rate used for generating the RTP timestamp is independent of the number of channels and the encoding; it equals the number of sampling periods per second. For N-channel encodings, each sampling period (say, $1/8000$ 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 [1], using the following notation:

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	quadrophonic	Fl	Fr	Rl	Rr		
4		l	c	r	S		
5		Fl	Fr	Fc	Sl	Sr	
6		l	lc	c	r	rc	S

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: 8000, 11025, 16000, 22050, 24000, 32000, 44100 and 48000 Hz. (The Apple Macintosh computers have native sample rates of 22254.54 and 11127.27, which can be converted to 22050 and 11025 with acceptable quality by dropping 4 or 2 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 links with severe bandwidth constraints, a higher packetization delay may be appropriate. A receiver should accept packets representing between 0 and 200 ms of audio data. (For framed audio encodings, a receiver should accept packets with 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 are 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 are 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.

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 since the audio frame duration (in octets) is defined as part of the encoding, as long as all frames have the same length measured in octets. This does not work when carrying frames of different sizes unless the frame sizes are relatively prime.

For frame-based codecs, the channel order is defined for the whole block. That is, for two-channel audio, right and left samples are 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 capturing time of the first sample in the first frame, that is, the oldest information in the packet.

4.5 Audio Encodings

encoding	sample/frame	bits/sample	ms/frame	ms/packet
1016	frame	N/A	30	30
DVI4	sample	4		20
G721	sample	4		20
G722	sample	8		20
G723	frame	N/A	30	30
G728	frame	N/A	2.5	20
G729	frame	N/A	10	20
GSM	frame	N/A	20	20
L8	sample	8		20
L16	sample	16		20
LPC	frame	N/A	20	20
MPA	frame	N/A		20
PCMA	sample	8		20
PCMU	sample	8		20
VDVI	sample	var.		20

Table 1: Properties of Audio Encodings

The characteristics of standard audio encodings are shown in Table 1 and their payload types are listed in Table 4.

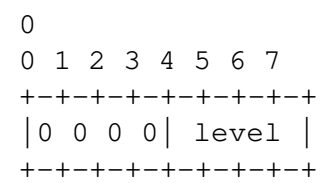
4.5.1 1016

Encoding 1016 is a frame based encoding using code-excited linear prediction (CELP) and is specified in Federal Standard FED-STD 1016 [2, 3, 4, 5].

The U. S. DoD's Federal-Standard-1016 based 4800 bps code excited linear prediction voice coder version 3.2 (CELP 3.2) Fortran and C simulation source codes are available for worldwide distribution at no charge (on DOS diskettes, but configured to compile on Sun SPARC stations) from: Bob Fenichel, National Communications System, Washington, D.C. 20305, phone +1-703-692-2124, fax +1-703-746-4960.

4.5.2 CN

The G.764-based VAD (voice activity detector) noise level packet contains a single-octet message to the receiver to play comfort noise at the absolute dBmO level specified by the G.764 level index. This message would normally be sent once at the beginning of a silence period (which also indicates the transition from speech to silence), but rate of noise level updates is implementation specific. The mapping of the index to absolute noise levels measured on the transmit side is given in Table 2, with the level index packed into the least significant bits of the noise-level payload, as shown below.



The RTP header for the comfort noise packet should be constructed as if the VAD noise were an independent codec, but sharing the media clock and sequence number space with the associated voice codec. Thus, the RTP timestamp designates the beginning of the silence period, using the timestamp frequency of the payload type immediately preceding the CN packet. The RTP packet should not have the marker bit set.

Note: dBrnc0 is the noise power measured in dBrnC, but referenced to the zero-level transmission level point (TLP). Typically, the two-wire interface in telephony is at the zero-level TLP of 0 dBm. dBrnC is the power level of noise with C-message weighting expressed in decibels relative to reference noise. Reference noise power is -90 dBm or 1 pW. (dBm is the power level in decibels relative to 1 mW, with an impedance of 600 Ohms.) The C-message weighting is described in [6, p. 36]. To obtain dBmC0 levels, subtract 90 dB from the values listed.

4.5.3 DVI4

DVI4 is specified, with pseudo-code, in [7] as the IMA ADPCM wave type.

However, the encoding defined here as DVI4 differs in three respects from this recommendation:

- The header contains the predicted value rather than the first sample value.
- 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

Index	Noise Level (dBrncO)
0	Idle Code
1	16.6
2	19.7
3	22.6
4	24.9
5	26.9
6	29.0
7	31.0
8	32.8
9	34.6
10	36.2
11	37.9
12	39.7
13	41.6
14	43.8
15	46.6

Table 2: G.764 noise level mapping

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 little-endian order.

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

Packing of samples for multiple channels is for further study.

The document *IMA Recommended Practices for Enhancing Digital Audio Compatibility in Multimedia Systems (version 3.0)* contains the algorithm description. It is available from

Interactive Multimedia Association
48 Maryland Avenue, Suite 202
Annapolis, MD 21401-8011

USA
phone: +1 410 626-1380

4.5.4 G721

G721 is specified in ITU recommendation G.721. Reference implementations for G.721 are available as part of the CCITT/ITU-T Software Tool Library (STL) from the ITU General Secretariat, Sales Service, Place du Nations, CH-1211 Geneve 20, Switzerland. The library is covered by a license.

4.5.5 G722

G722 is specified in ITU-T recommendation G.722, "7 kHz audio-coding within 64 kbit/s".

4.5.6 G723

G.723.1 is specified in ITU recommendation G.723.1, "Dual-rate speech coder for multimedia communications transmitting at 5.3 and 6.3 kbit/s". 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.

4.5.7 G726-32

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 8000 samples/sec to and from a 32 kbit/s channel. The conversion is applied to the PCM stream using an Adaptive Differential Pulse Code Modulation (ADPCM) transcoding technique. G.726 is a backwards-compatible superset of G.721, a recommendation which is no longer in force. G.726 also describes codecs operating at 40 (5 bits/sample), 24 (3 bits/sample) and 16 kb/s (2 bits/sample). These are labeled G726-40, G726-24 and G726-16, respectively.

No header information shall be included as part of the audio data. The 4-bit code words of the G.726 encoding MUST be packed into octets as follows: the first code word is placed in the four least significant bits of the first octet, with the least significant bit of the code word in the least significant bit of the octet; the second code word is placed in the four most significant bits of the first octet, with the most significant bit of the code word in the most significant bit of the octet. Subsequent pairs of the code words shall be packed in the same way into successive octets, with the first code word of each pair placed in the least significant four bits of the octet. It is prefered that the voice sample be extended with silence such that the encoded value comprises an even number of code words.

4.5.8 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-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.

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--->
<--B1--><--B2--><--B3--><--B4--><--B5-->
<--------------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.

4.5.9 G729

G.729 and G.729A are defined in ITU-T Recommendation G.729, “Coding of Speech at 8 kbit/s using Conjugate Structure-Algebraic Code Excited Linear Predictive (CS-ACELP) Coding” and its Annex A, respectively. These two audio codecs are compatible with each other on the wire so there is no need to distinguish further between them. The codecs were optimized to represent speech with a high quality; G.729A achieves this with very low complexity.

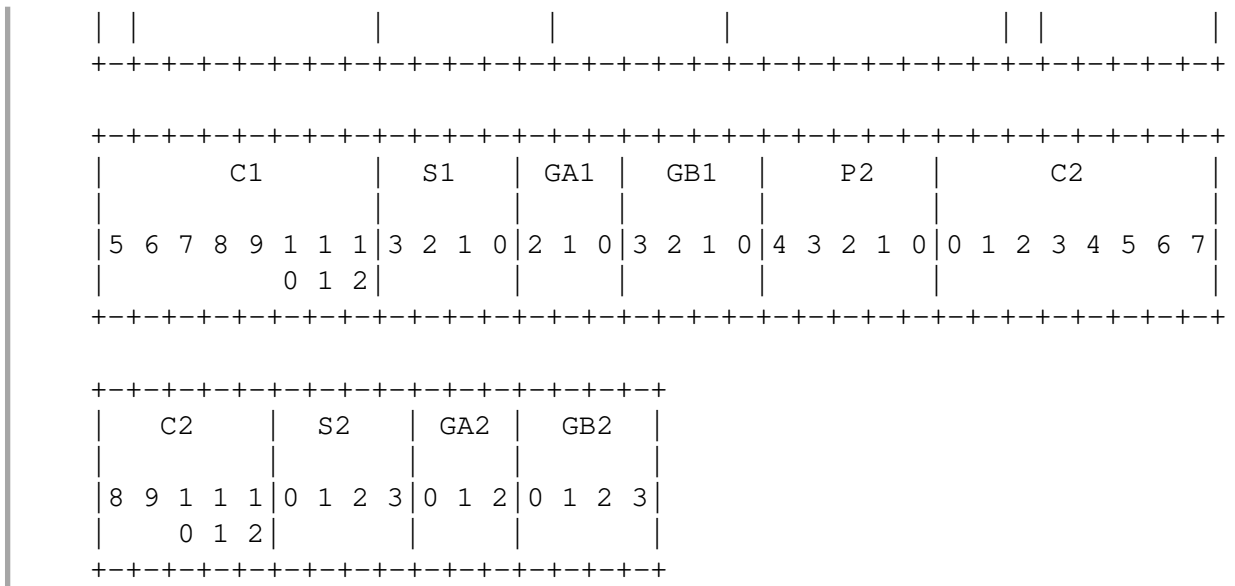
A voice activity detector (VAD) and comfort noise generator (CNG) is defined in G.729 Annex B (G.729B). It can be used in conjunction with either G.729 or G.729A. A G.729 or G.729A frame contains 10 octets, while the G.729B comfort noise frame contains 4 octets.

An RTP packet may consist of zero or more G.729 or G.729A frames, followed by zero or one G.729B payload.

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. Bits are numbered as Internet order, that is, the most significant bit is bit 0.

```
 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|
```



4.5.10 GSM

GSM (group speciale mobile) denotes the European GSM 06.10 provisional standard for full-rate speech transcoding, prI-ETS 300 036, which is based on RPE/LTP (residual pulse excitation/long term prediction) coding at a rate of 13 kb/s [8, 9, 10]. 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.

General Packaging Issues The GSM standard specifies the bit stream produced by the codec, but does not specify how these bits should be packed for transmission. Some software implementations of the GSM codec use a different packing than that specified here.

In the GSM encoding used by RTP, the bits are 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 (4-7) and the 4 most significant bits of F1 (2-5) in the 4 least significant bits (0-3). The second octet contains the 2 least bits of F1 in bits 6-7, and F2 in bits 0-5, and so on. The order of the fields in the frame is as follows:

GSM variable names and numbers So if F.i signifies the ith bit of the field F, and bit 0 is the most significant bit, and the bits of every octet are numbered from 0 to 7 from most to least significant, then in the RTP encoding we have:

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 3: 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

4.5.11 L8

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

4.5.12 L16

L16 denotes uncompressed audio data, using 16-bit signed representation with 65535 equally divided steps between minimum and maximum signal level, ranging from -32768 to 32767. The value is represented in two's complement notation and network byte order.

4.5.13 LPC

LPC designates an experimental linear predictive encoding contributed by Ron Frederick, Xerox PARC, which is based on an implementation written by Ron Zuckerman, Motorola, 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.

4.5.14 MPA

MPA denotes MPEG-I or MPEG-II 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 2038 [11].

Sampling rate and channel count are contained in the payload. MPEG-I audio supports sampling rates of 32000, 44100, and 48000 Hz (ISO/IEC 11172-3, section 1.1; "Scope"). MPEG-II additionally supports ISO/IEC 11172-3 Audio...").

4.5.15 PCMA

PCMA is specified in CCITT/ITU-T recommendation G.711. Audio data is encoded as eight bits per sample, after logarithmic scaling. Code to convert between linear and A-law companded data is available in [7]. A detailed description is given by Jayant and Noll [12].

4.5.16 PCMU

PCMU is specified in CCITT/ITU-T recommendation G.711. Audio data is encoded as eight bits per sample, after logarithmic scaling. Code to convert between linear and mu-law companded data is available in [7]. PCMU is the encoding used for the Internet media type audio/basic. A detailed description is given by Jayant and Noll [12].

4.5.17 RED

The redundant audio payload format "RED" is specified by RFC XXX. 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 the previous packetization interval. This allows an approximation of the data from lost packets to be recovered upon decoding of the following packet, giving much improved sound quality when compared with silence substitution for lost packets.

4.5.18 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.

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 video encodings are currently defined, with their abbreviated names used for identification:

5.1 CelB

The CELL-B encoding is a proprietary encoding proposed by Sun Microsystems. The byte stream format is described in RFC 2029 [13].

5.2 JPEG

The encoding is specified in ISO Standards 10918-1 and 10918-2. The RTP payload format is as specified in RFC 2035 [14].

5.3 H261

The encoding is specified in CCITT/ITU-T standard H.261. The packetization and RTP-specific properties are described in RFC 2032 [15].

5.4 MPV

MPV designates the use MPEG-I and MPEG-II 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 2038

[11], Section 3.

5.5 MP2T

MP2T designates the use of MPEG-II transport streams, for either audio or video. The encapsulation is described in RFC 2038 [11], Section 2. See the description of the MPA audio encoding for contact information.

5.6 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
Xerox Palo Alto Research Center
3333 Coyote Hill Road
Palo Alto, CA 94304
United States
electronic mail: `frederic@parc.xerox.com`

6 Payload Type Definitions

Table 4 defines this profile's static payload type values for the PT field of the RTP data header. A new RTP payload format specification may be registered with the IANA by name, and may also be assigned a static payload type value from the range marked 'unassigned' in Table 4 subject to the constraints listed in Section 3.

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. The payload type range marked 'reserved' has been set aside so that RTCP and RTP packets can be reliably distinguished (see Section "Summary of Protocol Constants" of the RTP protocol specification).

An RTP source emits a single RTP payload type at any given instant. The interleaving or multiplexing of several RTP media types within a single RTP session is not allowed, but multiple RTP sessions may be used in parallel to send multiple media types. An RTP source may change payload types during a session.

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. A single RTP session consists of payload types of one and only media type.

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 of the applications used, negotiated by a conference control protocol or established by agreement between the human participants. The media types in Table 4 are marked as "A" for audio, "V" for video and "AV" for combined audio/video streams.

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

All current video encodings use a 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 are possible. 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.

The standard video encodings and their payload types are listed in Table 4.

7 Port Assignment

As specified in the RTP protocol definition, RTP data is to be carried on an even UDP port number and the corresponding RTCP packets are to 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 accomodate port number allocation practice within 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.

References

[1] Apple Computer, "Audio interchange file format AIFF-C," Aug. 1991. (also ftp://ftp.sgi.com/sgi/aiff-c.9.26.91.ps.Z).

[2] Office of Technology and Standards, "Telecommunications: Analog to digital conversion of radio voice by 4,800 bit/second code excited linear prediction (celp)," Federal Standard FS-1016, GSA, Room 6654; 7th & D Street SW; Washington, DC 20407 (+1-202-708-9205), 1990.

[3] J. P. Campbell, Jr., T. E. Tremain, and V. C. Welch, "The proposed Federal Standard 1016 4800 bps voice coder: CELP," *Speech Technology*, vol. 5, pp. 58–64, April/May 1990.

[4] J. P. Campbell, Jr., T. E. Tremain, and V. C. Welch, "The federal standard 1016 4800 bps CELP voice coder," *Digital Signal Processing*, vol. 1, no. 3, pp. 145–155, 1991.

[5] J. P. Campbell, Jr., T. E. Tremain, and V. C. Welch, "The dod 4.8 kbps standard (proposed federal standard 1016)," in *Advances in Speech Coding* (B. Atal, V. Cuperman, and A. Gersho, eds.), ch. 12, pp. 121–133, Kluwer Academic Publishers, 1991.

PT	encoding name	media type	clock rate (Hz)	channels (audio)
0	PCMU	A	8000	1
1	1016	A	8000	1
2	G721	A	8000	1
3	GSM	A	8000	1
4	G.723.1	A	8000	1
5	DVI4	A	8000	1
6	DVI4	A	16000	1
7	LPC	A	8000	1
8	PCMA	A	8000	1
9	G722	A	16000	1
10	L16	A	44100	2
11	L16	A	44100	1
12	G723	A	8000	1
13	CN	A		
14	MPA	A	90000	(see text)
15	G728	A	8000	1
16	DVI4	A	11025	1
17	DVI4	A	22050	1
18	G729	A	8000	1
19–22	unassigned	A		
24	unassigned	V		
25	CelB	V	90000	
26	JPEG	V	90000	
27	unassigned	V		
28	nv	V	90000	
29	unassigned	V		
30	unassigned	V		
31	H261	V	90000	
32	MPV	V	90000	
33	MP2T	AV	90000	
34	H263	V	90000	
35–71	unassigned	?		
72–76	reserved	N/A	N/A	N/A
77	RED	A	N/A	N/A
78–95	unassigned	?		
96–127	dynamic	?		

Table 4: Payload types (PT) for standard audio and video encodings

[6] J. Bellamy, *Digital Telephony*. New York: John Wiley & Sons, 1991.

[7] IMA Digital Audio Focus and Technical Working Groups, "Recommended practices for enhancing digital audio compatibility in multimedia systems (version 3.00)," tech. rep., Interactive Multimedia Association, Annapolis, Maryland, Oct. 1992.

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

[9] J. Degener, "Digital speech compression," *Dr. Dobb's Journal*, Dec. 1994.

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

[11] D. Hoffman, G. Fernando, and V. Goyal, "RTP payload format for MPEG1/MPEG2 video," Request for Comments (Proposed Standard) RFC 2038, Internet Engineering Task Force, Oct. 1996.

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

[13] M. Speer and D. Hoffman, "RTP payload format of sun's CellB video encoding," Request for Comments (Proposed Standard) RFC 2029, Internet Engineering Task Force, Oct. 1996.

[14] L. Berc, W. Fenner, R. Frederick, and S. McCanne, "RTP payload format for JPEG-compressed video," Request for Comments (Proposed Standard) RFC 2035, Internet Engineering Task Force, Oct. 1996.

[15] T. Turletti and C. Huitema, "RTP payload format for H.261 video streams," Request for Comments (Proposed Standard) RFC 2032, Internet Engineering Task Force, Oct. 1996.

8 Acknowledgements

The comments and careful review of Steve Casner are gratefully acknowledged. The GSM description was adopted from the *IMTC Voice over IP Forum Service Interoperability Implementation Agreement* (January 1997). Fred Burg helped with the G.729 description.

9 Address of Author

Henning Schulzrinne
Dept. of Computer Science
Columbia University
1214 Amsterdam Avenue
New York, NY 10027
USA
electronic mail: `schulzrinne@cs.columbia.edu`

Current Locations of Related Resources

UTF-8

Information on the UCS Transformation Format 8 (UTF-8) is available at

http://www.stonehand.com/unicode/standard/utf8.html

1016

An implementation is available at

ftp://ftp.super.org/pub/speech/celp_3.2a.tar.Z

DVI4

An implementation is available from Jack Jansen at

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

G721

An implementation is available at

ftp://gaia.cs.umass.edu/pub/hgschulz/ccitt/ccitt_tools.tar.Z

G723

Reference implementations for G.723.1 are available as part of the CCITT/ITU-T Software Tool Library (STL) from the ITU General Secretariat, Sales Service, Place du Nations, CH-1211 Geneve 20, Switzerland. The library is covered by a license.

The specification also contains C source code. Source code files are available at

http://www4.itu.ch/itudoc/itu-t/rec/g/g700-799/g723-1/723disk1_32415.html

and test vectors at

http://www4.itu.ch/itudoc/itu-t/rec/g/g700-799/g723-1/723disk2_32416.html

G729

Reference implementations for G.729, G.729A and G.729B are available as part of the ITU-T Software Tool Library from the ITU General Secretariat, Sales Service, Place de Nations, CH-1211 Geneve 20, Switzerland. The library is covered by a license.

GSM

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

ftp://ftp.cs.tu-berlin.de/pub/local/kbs/tubmik/gsm/

LPC

An implementation is available at

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