Network Working Group   J. Sjoberg
Request for Comments: 4352   M. Westerlund
Category: Standards Track   Ericsson
                                                        A. Lakaniemi
                                                           S. Wenger
                                                               Nokia
                                                        January 2006

                      RTP Payload Format for the
         Extended Adaptive Multi-Rate Wideband (AMR-WB+) Audio Codec

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.

Abstract

   This document specifies a Real-time Transport Protocol (RTP) payload
   format for Extended Adaptive Multi-Rate Wideband (AMR-WB+) encoded
   audio signals. The AMR-WB+ codec is an audio extension of the AMR-WB
   speech codec. It encompasses the AMR-WB frame types and a number of
   new frame types designed to support high-quality music and speech. A
   media type registration for AMR-WB+ is included in this
   specification.

Table of Contents

1. Introduction

   This document specifies the payload format for packetization of
   Extended Adaptive Multi-Rate Wideband (AMR-WB+) [1] encoded audio
   signals into the Real-time Transport Protocol (RTP) [3]. The payload
   format supports the transmission of mono or stereo audio, aggregating
   multiple frames per payload, and mechanisms enhancing the robustness
   of the packet stream against packet loss.

   The AMR-WB+ codec is an extension of the Adaptive Multi-Rate Wideband
   (AMR-WB) speech codec. New features include extended audio bandwidth
   to enable high quality for non-speech signals (e.g., music), native
   support for stereophonic audio, and the option to operate on, and
   switch between, several internal sampling frequencies (ISFs). The
   primary usage scenario for AMR-WB+ is the transport over IP.
   Therefore, interworking with other transport networks, as discussed
   for AMR-WB in [7], is not a major concern and hence not addressed in
   this memo.

   The expected key application for AMR-WB+ is streaming. To make the
   packetization process on a streaming server as efficient as possible,
   an octet-aligned payload format is desirable. Therefore, a
   bandwidth-efficient mode (as defined for AMR-WB in [7]) is not
   specified herein; the bandwidth savings of the bandwidth-efficient
   mode would be very small anyway, since all extension frame types are
   octet aligned.

   The stereo encoding capability of AMR-WB+ renders the support for
   multi-channel transport at RTP payload format level, as specified for
   AMR-WB [7], obsolete. Therefore, this feature is not included in
   this memo.

   This specification does not include a definition of a file format for
   AMR-WB+. Instead, it refers to the ISO-based 3GP file format [14],
   which supports AMR-WB+ and provides all functionality required. The
   3GP format also supports storage of AMR, AMR-WB, and many other
   multi-media formats, thereby allowing synchronized playback.

   The rest of the document is organized as follows: Background
   information on the AMR-WB+ codec, and design principles, can be found
   in Section 3. The payload format itself is specified in Section 4.
   Sections 5 and 6 discuss congestion control and security
   considerations, respectively. In Section 7, a media type
   registration is provided.

2. Definitions

2.1. Glossary

    3GPP   - Third Generation Partnership Project
    AMR   - Adaptive Multi-Rate (Codec)
    AMR-WB   - Adaptive Multi-Rate Wideband (Codec)
    AMR-WB+ - Extended Adaptive Multi-Rate Wideband (Codec)
    CN   - Comfort Noise
    DTX   - Discontinuous Transmission
    FEC   - Forward Error Correction
    FT   - Frame Type
    ISF   - Internal Sampling Frequency
    SCR   - Source-Controlled Rate Operation
    SID   - Silence Indicator (the frames containing only CN
               parameters)
    TFI   - Transport Frame Index
    TS   - Timestamp
    VAD   - Voice Activity Detection
    UED   - Unequal Error Detection
    UEP   - Unequal Error Protection

2.2. 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].

3. Background of AMR-WB+ and Design Principles

    The Extended Adaptive Multi-Rate Wideband (AMR-WB+) [1] audio codec
    is designed to compress speech and audio signals at low bit-rate and
    good quality. The codec is specified by the Third Generation
    Partnership Project (3GPP). The primary target applications are 1)
    the packet-switched streaming service (PSS) [13], 2) multimedia
    messaging service (MMS) [18], and 3) multimedia broadcast and
    multicast service (MBMS) [19]. However, due to its flexibility and
    robustness, AMR-WB+ is also well suited for streaming services in
    other highly varying transport environments, for example, the
    Internet.

3.1. The AMR-WB+ Audio Codec

    3GPP originally developed the AMR-WB+ audio codec for streaming and
    messaging services in Global System for Mobile communications (GSM)
    and third generation (3G) cellular systems. The codec is designed as
    an audio extension of the AMR-WB speech codec. The extension adds
    new functionality to the codec in order to provide high audio quality

for a wide range of signals including music. Stereophonic operation
has also been added. A new, high-efficiency hybrid stereo coding
algorithm enables stereo operation at bit-rates as low as 6.2 kbit/s.

The AMR-WB+ codec includes the nine frame types specified for AMR-WB,
extended by new bit-rates ranging from 5.2 to 48 kbit/s. The AMR-WB
frame types can employ only a 16000 Hz sampling frequency and operate
only on monophonic signals. The newly introduced extension frame
types, however, can operate at a number of internal sampling
frequencies (ISFs), both in mono and stereo. Please see Table 24 in
[1] for details. The output sampling frequency of the decoder is
limited to 8, 16, 24, 32, or 48 kHz.

An overview of the AMR-WB+ encoding operations is provided as
follows. The encoder receives the audio sampled at, for example, 48
kHz. The encoding process starts with pre-processing and resampling
to the user-selected ISF. The encoding is performed on equally sized
super-frames. Each super-frame corresponds to 2048 samples per
channel, at the ISF. The codec carries out a number of encoding
decisions for each super-frame, thereby choosing between different
encoding algorithms and block lengths, so as to achieve a fidelity-
optimized encoding adapted to the signal characteristics of the
source. The stereo encoding (if used) executes separately from the
monophonic core encoding, thus enabling the selection of different
combinations of core and stereo encoding rates. The resulting
encoded audio is produced in four transport frames of equal length.
Each transport frame corresponds to 512 samples at the ISF and is
individually usable by the decoder, provided that its position in the
super-frame structure is known.

The codec supports 13 different ISFs, ranging from 12.8 to 38.4 kHz,
as described by Table 24 of [1]. The high number of ISFs allows a
trade-off between the audio bandwidth and the target bit-rate. As
encoding is performed on 2048 samples at the ISF, the duration of a
super-frame and the effective bit-rate of the frame type in use
varies.

The ISF of 25600 Hz has a super-frame duration of 80 ms. This is the
'nominal' value used to describe the encoding bit-rates henceforth.
Assuming this normalization, the ISF selection results in bit-rate
variations from 1/2 up to 3/2 of the nominal bit-rate.

The encoding for the extension modes is performed as one monophonic
core encoding and one stereo encoding. The core encoding is executed
by splitting the monophonic signal into a lower and a higher
frequency band. The lower band is encoded employing either algebraic
code excited linear prediction (ACELP) or transform coded excitation
(TCX). This selection can be made once per transport frame, but must

obey certain limitations of legal combinations within the super-
frame. The higher band is encoded using a low-rate parametric
bandwidth extension approach.

The stereo signal is encoded employing a similar frequency band
decomposition; however, here the signal is divided into three bands
that are individually parameterized.

The total bit-rate produced by the extension is the result of the
combination of the encoder's core rate, stereo rate, and ISF. The
extension supports 8 different core encoding rates, producing bit-
rates between 10.4 and 24.0 kbit/s; see Table 22 in [1]. There are
16 stereo encoding rates generating bit-rates between 2.0 and 8.0
kbit/s; see Table 23 in [1]. The frame type uniquely identifies the
AMR-WB modes, 4 fixed extension rates (see below), 24 combinations of
core and stereo rates for stereo signals, and the 8 core rates for
mono signals, as listed in Table 25 in [1]. This implies that the
AMR-WB+ supports encoding rates between 10.4 and 32 kbit/s, assuming
an ISF of 25600 Hz.

Different ISFs allow for additional freedom in the produced bit-rates
and audio quality. The selection of an ISF changes the available
audio bandwidth of the reconstructed signal, and also the total bit-
rate. The bit-rate for a given combination of frame type and ISF is
determined by multiplying the frame type's bit-rate with the used
ISF's bit-rate factor; see Table 24 in [1].

The extension also has four frame types which have fixed ISFs.
Please see frame types 10-13 in Table 21 in [1]. These four pre-
defined frame types have a fixed input sampling frequency at the
encoder, which can be set at either 16 or 24 kHz. Like the AMR-WB
frame types, transport frames encoded utilizing these frame types
represent exactly 20 ms of the audio signal. However, they are also
part of 80 ms super-frames. Frame types 0-13 (AMR-WB and fixed
extension rates), as listed in Table 21 in [1], do not require an
explicit ISF indication. The other frame types, 14-47, require the
ISF employed to be indicated.

The 32 different frame types of the extension, in combination with 13
ISFs, allows for a great flexibility in bit-rate and selection of
desired audio quality. A number of combinations exist that produce
the same codec bit-rate. For example, a 32 kbit/s audio stream can
be produced by utilizing frame type 41 (i.e., 25.6 kbit/s) and the
ISF of 32kHz (5/4 * (19.2+6.4) = 32 kbit/s), or frame type 47 and the
ISF of 25.6 kHz (1 * (24 + 8) = 32 kbit/s). Which combination is
more beneficial for the perceived audio quality depends on the
content. In the above example, the first case provides a higher
audio bandwidth, while the second one spends the same number of bits

on somewhat narrower audio bandwidth but provides higher fidelity.
Encoders are free to select the combination they deem most
beneficial.

Since a transport frame always corresponds to 512 samples at the used
ISF, its duration is limited to the range 13.33 to 40 ms; see Table
1. An RTP Timestamp clock rate of 72000 Hz, as mandated by this
specification, results in AMR-WB+ transport frame lengths of 960 to
2880 timestamp ticks, depending solely on the selected ISF.

Index	ISF	Duration(ms)	Duration(TS Ticks @ 72 kHz)
0	N/A	20	1440
1	12800	40	2880
2	14400	35.55	2560
3	16000	32	2304
4	17067	30	2160
5	19200	26.67	1920
6	21333	24	1728
7	24000	21.33	1536
8	25600	20	1440
9	28800	17.78	1280
10	32000	16	1152
11	34133	15	1080
12	36000	14.22	1024
13	38400	13.33	960

Table 1: Normative number of RTP Timestamp Ticks for each
         Transport Frame depending on ISF (ISF and Duration in
         ms are rounded)

The encoder is free to change both the ISF and the encoding frame
type (both mono and stereo) during a session. For the extension
frame types with index 10-13 and 16-47, the ISF and frame type
changes are constrained to occur at super-frame boundaries. This
implies that, for the frame types mentioned, the ISF is constant
throughout a super-frame. This limitation does not apply for frame
types with index 0-9, 14, and 15; i.e., the original AMR-WB frame
types.

A number of features of the AMR-WB+ codec require special
consideration from a transport point of view, and solutions that
could perhaps be viewed as unorthodox. First, there are constraints
on the RTP timestamping, due to the relationship of the frame
duration and the ISFs. Second, each frame of encoded audio must
maintain information about its frame type, ISF, and position in the
super-frame.

3.2. Multi-rate Encoding and Rate Adaptation

   The multi-rate encoding capability of AMR-WB+ is designed to preserve
   high audio quality under a wide range of bandwidth requirements and
   transmission conditions.

   AMR-WB+ enables seamless switching between frame types that use the
   same number of audio channels and the same ISF. Every AMR-WB+ codec
   implementation is required to support all frame types defined by the
   codec and must be able to handle switching between any two frame
   types. Switching between frame types employing a different number of
   audio channels or a different ISF must also be supported, but it may
   not be completely seamless. Therefore, it is recommended to perform
   such switching infrequently and, if possible, during periods of
   silence.

3.3. Voice Activity Detection and Discontinuous Transmission

   AMR-WB+ supports the same algorithms as AMR-WB for voice activity
   detection (VAD) and generation of comfort noise (CN) parameters
   during silence periods. However, these functionalities can only be
   used in conjunction with the AMR-WB frame types (FT=0-8). This
   option allows reducing the number of transmitted bits and packets
   during silence periods to a minimum. The operation of sending CN
   parameters at regular intervals during silence periods is usually
   called discontinuous transmission (DTX) or source controlled rate
   (SCR) operation. The AMR-WB+ frames containing CN parameters are
   called Silence Indicator (SID) frames. More details about the VAD
   and DTX functionality are provided in [4] and [5].

3.4. Support for Multi-Channel Session

   Some of the AMR-WB+ frame types support the encoding of stereophonic
   audio. Because of this native support for a two-channel stereophonic
   signal, it does not seem necessary to support multi-channel transport
   with separate codec instances, as specified in the AMR-WB RTP payload
   [7]. The codec has the capability of stereo to mono downmixing as
   part of the decoding process. Thus, a receiver that is only capable
   of playout of monophonic audio must still be able to decode and play
   signals originally encoded and transmitted as stereo. However, to
   avoid spending bits on a stereo encoding that is not going to be
   utilized, a mechanism is defined in this specification to signal
   mono-only audio.

3.5. Unequal Bit-Error Detection and Protection

   The audio bits encoded in each AMR-WB frame are sorted according to
   their different perceptual sensitivity to bit errors. In cellular
   systems, for example, this property can be exploited to achieve
   better voice quality, by using unequal error protection and detection
   (UEP and UED) mechanisms. However, the bits of the extension frame
   types of the AMR-WB+ codec do not have a consistent perceptual
   significance property and are not sorted in this order. Thus, UEP or
   UED is meaningless with the extension frame types. If there is a
   need to use UEP or UED for AMR-WB frame types, it is recommended that
   RFC 3267 [7] be used.

3.6. Robustness against Packet Loss

   The payload format supports two mechanisms to improve robustness
   against packet loss: simple forward error correction (FEC) and frame
   interleaving.

3.6.1. Use of Forward Error Correction (FEC)

   Generic forward error correction within RTP is defined, for example,
   in RFC 2733 [11]. Audio redundancy coding is defined in RFC 2198
   [12]. Either scheme can be used to add redundant information to the
   RTP packet stream and make it more resilient to packet losses, at the
   expense of a higher bit rate. Please see either RFC for a discussion
   of the implications of the higher bit rate to network congestion.

   In addition to these media-unaware mechanisms, this memo specifies an
   AMR-WB+ specific form of audio redundancy coding, which may be
   beneficial in terms of packetization overhead.

   Conceptually, previously transmitted transport frames are aggregated
   together with new ones. A sliding window is used to group the frames
   to be sent in each payload. Figure 1 below shows an example.

```
   --+--------+--------+--------+--------+--------+--------+--------+--
     | f(n-2) | f(n-1) |  f(n)  | f(n+1) | f(n+2) | f(n+3) | f(n+4) |
   --+--------+--------+--------+--------+--------+--------+--------+--


     <---- p(n-1) ---->
              <----- p(n) ----->
                       <---- p(n+1) ---->
                                <---- p(n+2) ---->
                                         <---- p(n+3) ---->
                                                  <---- p(n+4) ---->
```
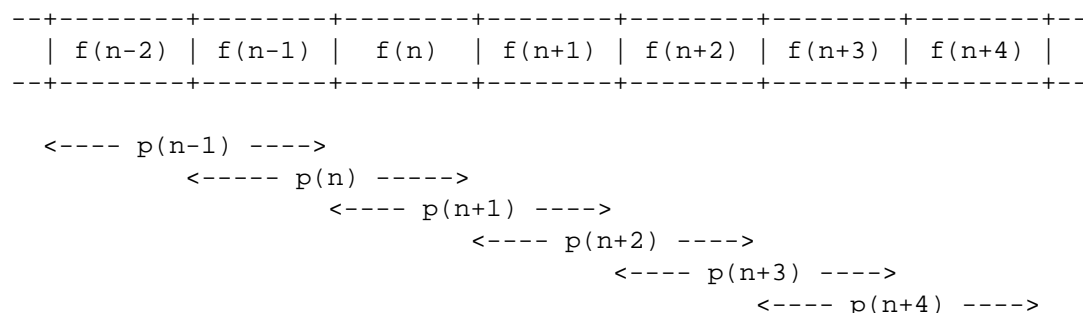
   Figure 1: An example of redundant transmission

Here, each frame is retransmitted once in the following RTP payload
packet. F(n-2)...f(n+4) denote a sequence of audio frames, and
p(n-1)...p(n+4) a sequence of payload packets.

The mechanism described does not require signaling at the session
setup. In other words, the audio sender can choose to use this
scheme without consulting the receiver. For a certain timestamp, the
receiver may receive multiple copies of a frame containing encoded
audio data or frames indicated as NO_DATA. The cost of this scheme
is bandwidth and the receiver delay necessary to allow the redundant
copy to arrive.

This redundancy scheme provides a functionality similar to the one
described in RFC 2198, but it works only if both original frames and
redundant representations are AMR-WB+ frames. When the use of other
media coding schemes is desirable, one has to resort to RFC 2198.

The sender is responsible for selecting an appropriate amount of
redundancy based on feedback about the channel conditions, e.g., in
the RTP Control Protocol (RTCP) [3] receiver reports. The sender is
also responsible for avoiding congestion, which may be exacerbated by
redundancy (see Section 5 for more details).

3.6.2. Use of Frame Interleaving

   To decrease protocol overhead, the payload design allows several
   audio transport frames to be encapsulated into a single RTP packet.
   One of the drawbacks of such an approach is that in case of packet
   loss several consecutive frames are lost. Consecutive frame loss
   normally renders error concealment less efficient and usually causes
   clearly audible and annoying distortions in the reconstructed audio.
   Interleaving of transport frames can improve the audio quality in
   such cases by distributing the consecutive losses into a number of
   isolated frame losses, which are easier to conceal. However,
   interleaving and bundling several frames per payload also increases
   end-to-end delay and sets higher buffering requirements. Therefore,
   interleaving is not appropriate for all use cases or devices.
   Streaming applications should most likely be able to exploit
   interleaving to improve audio quality in lossy transmission
   conditions.

   Note that this payload design supports the use of frame interleaving
   as an option. The usage of this feature needs to be negotiated in
   the session setup.

   The interleaving supported by this format is rather flexible. For
   example, a continuous pattern can be defined, as depicted in Figure
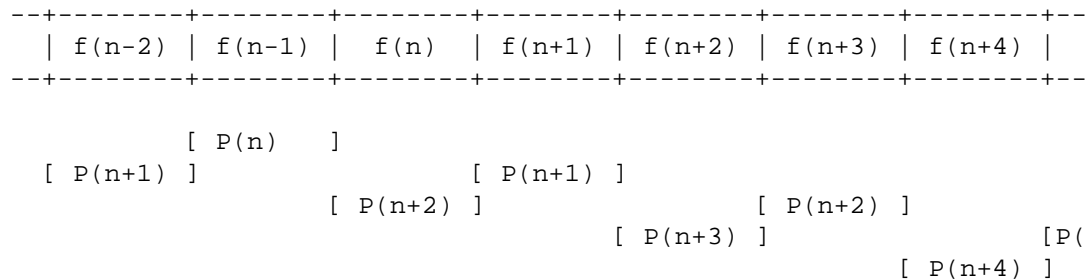   2.

```
--+--------+--------+--------+--------+--------+--------+--------+--
  | f(n-2) | f(n-1) |  f(n)  | f(n+1) | f(n+2) | f(n+3) | f(n+4) |
--+--------+--------+--------+--------+--------+--------+--------+--

          [ P(n)   ]
   [ P(n+1) ]                      [ P(n+1) ]
                      [ P(n+2) ]                      [ P(n+2) ]
                                     [ P(n+3) ]                      [P(
                                                       [ P(n+4) ]
```

Figure 2: An example of interleaving pattern that has constant delay

In Figure 2 the consecutive frames, denoted f(n-2) to f(n+4), are
aggregated into packets P(n) to P(n+4), each packet carrying two
frames. This approach provides an interleaving pattern that allows
for constant delay in both the interleaving and deinterleaving
processes. The deinterleaving buffer needs to have room for at least
three frames, including the one that is ready to be consumed. The
storage space for three frames is needed, for example, when f(n) is
the next frame to be decoded: since frame f(n) was received in packet
P(n+2), which also carried frame f(n+3), both these frames are stored
in the buffer. Furthermore, frame f(n+1) received in the previous
packet, P(n+1), is also in the deinterleaving buffer. Note also that
in this example the buffer occupancy varies: when frame f(n+1) is the
next one to be decoded, there are only two frames, f(n+1) and f(n+3),
in the buffer.

3.7. AMR-WB+ Audio over IP Scenarios

   Since the primary target application for the AMR-WB+ codec is
   streaming over packet networks, the most relevant usage scenario for
   this payload format is IP end-to-end between a server and a terminal,
   as shown in Figure 3.

```
          +----------+                              +----------+
          |          |    IP/UDP/RTP/AMR-WB+         |          |
          |  SERVER  |<---------------------------->| TERMINAL |
          |          |                              |          |
          +----------+                              +----------+
```
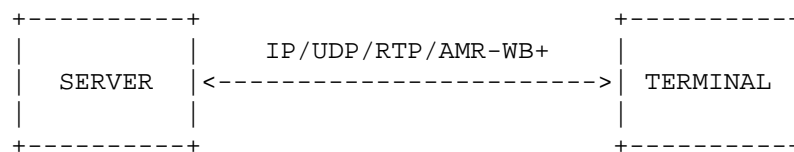
                 Figure 3: Server to terminal IP scenario

3.8. Out-of-Band Signaling

   Some of the options of this payload format remain constant throughout
   a session. Therefore, they can be controlled/negotiated at the
   session setup. Throughout this specification, these options and
   variables are denoted as "parameters to be established through out-

of-band means". In Section 7, all the parameters are formally
specified in the form of media type registration for the AMR-WB+
encoding. The method used to signal these parameters at session
setup or to arrange prior agreement of the participants is beyond the
scope of this document; however, Section 7.2 provides a mapping of
the parameters into the Session Description Protocol (SDP) [6] for
those applications that use SDP.

4. RTP Payload Format for AMR-WB+

   The main emphasis in the payload design for AMR-WB+ has been to
   minimize the overhead in typical use cases, while providing full
   flexibility with a slightly higher overhead. In order to keep the
   specification reasonably simple, we refrained from defining frame-
   specific parameters for each frame type. Instead, a few common
   parameters were specified that cover all types of frames.

   The payload format has two modes: basic mode and interleaved mode.
   The main structural difference between the two modes is the extension
   of the table of content entries with frame displacement fields when
   operating in the interleaved mode. The basic mode supports
   aggregation of multiple consecutive frames in a payload. The
   interleaved mode supports aggregation of multiple frames that are
   non-consecutive in time. In both modes it is possible to have frames
   encoded with different frame types in the same payload. The ISF must
   remain constant throughout the payload of a single packet.

   The payload format is designed around the property of AMR-WB+ frames
   that the frames are consecutive in time and share the same frame
   duration (in the absence of an ISF change). This enables the
   receiver to derive the timestamp for an individual frame within a
   payload. In basic mode, the deriving process is based on the order
   of frames. In interleaved mode, it is based on the compact
   displacement fields. The frame timestamps are used to regenerate the
   correct order of frames after reception, identify duplicates, and
   detect lost frames that require concealment.

   The interleaving scheme of this payload format is significantly more
   flexible than the one specified in RFC 3267. The AMR and AMR-WB
   payload format is only capable of using periodic patterns with frames
   taken from an interleaving group at fixed intervals. The
   interleaving scheme of this specification, in contrast, allows for
   any interleaving pattern, as long as the distance in decoding order
   between any two adjacent frames is not more than 256 frames. Note
   that even at the highest ISF this allows an interleaving depth of up
   to 3.41 seconds.

To allow for error resiliency through redundant transmission, the
periods covered by multiple packets MAY overlap in time. A receiver
MUST be prepared to receive any audio frame multiple times. All
redundantly sent frames MUST use the same frame type and ISF, and
MUST have the same RTP timestamp, or MUST be a NO_DATA frame (FT=15).

The payload consists of octet-aligned elements (header, ToC, and
audio frames). Only the audio frames for AMR-WB frame types (0-9)
require padding for octet alignment. If additional padding is
desired, then the P bit in the RTP header MAY be set, and padding MAY
be appended as specified in [3].

4.1. RTP Header Usage

The format of the RTP header is specified in [3]. This payload
format uses the fields of the header in a manner consistent with that
specification.

The RTP timestamp corresponds to the sampling instant of the first
sample encoded for the first frame in the packet. The timestamp
clock frequency SHALL be 72000 Hz. This frequency allows the frame
duration to be integer RTP timestamp ticks for the ISFs specified in
Table 1. It also provides reasonable conversion factors to the
input/output audio sampling frequencies supported by the codec. See
Section 4.3.2.3 for guidance on how to derive the RTP timestamp for
any audio frame beyond the first one.

The RTP header marker bit (M) SHALL be set to 1 whenever the first
frame carried in the packet is the first frame in a talkspurt (see
the definition of talkspurt in Section 4.1 of [9]). For all other
packets, the marker bit SHALL be set to zero (M=0).

The assignment of an RTP payload type for the format defined in this
memo is outside the scope of this document. The RTP profile in use
either assigns a static payload type or mandates binding the payload
type dynamically.

The media type parameter "channels" is used to indicate the maximum
number of channels allowed for a given payload type. A payload type
where channels=1 (mono) SHALL only carry mono content. A payload
type for which channels=2 has been declared MAY carry both mono and
stereo content. Note that this definition is different from the one
in RFC 3551 [9]. As mentioned before, the AMR-WB+ codec handles the
support of stereo content and the (eventual) downmixing of stereo to
mono internally. This makes it unnecessary to negotiate for the
number of channels for reasons other than bit-rate efficiency.

4.2. Payload Structure

   The payload consists of a payload header, a table of contents, and
   the audio data representing one or more audio frames. The following
   diagram shows the general payload format layout:

   +----------------+-------------------+----------------
   | payload header | table of contents | audio data ...
   +----------------+-------------------+----------------

   Payloads containing more than one audio frame are called compound
   payloads.

   The following sections describe the variations taken by the payload
   format depending on the mode in use: basic mode or interleaved mode.

4.3. Payload Definitions

4.3.1. Payload Header

   The payload header carries data that is common for all frames in the
   payload. The structure of the payload header is described below.

    0 1 2 3 4 5 6 7
   +-+-+-+-+-+-+-+-+
   | ISF |TFI|L|
   +-+-+-+-+-+-+-+-+

   ISF (5 bits): Indicates the Internal Sampling Frequency employed for
      all frames in this payload. The index value corresponds to
      internal sampling frequency as specified in Table 24 in [1]. This
      field SHALL be set to 0 for payloads containing frames with Frame
      Type values 0-13.

   TFI (2 bits): Transport Frame Index, from 0 (first) to 3 (last),
      indicating the position of the first transport frame of this
      payload in the AMR-WB+ super-frame structure. For payloads with
      frames of only Frame Type values 0-9, this field SHALL be set to 0
      by the sender. The TFI value for a frame of type 0-9 SHALL be
      ignored by the receiver. Note that the frame type is coded in the
      table of contents (as discussed later); hence, the mentioned
      dependencies of the frame type can be applied easily by
      interpreting only values carried in the payload header. It is not
      necessary to interpret the audio bit stream itself.

L (1 bit): Long displacement field flag for payloads in interleaved
   mode. If set to 0, four-bit displacement fields are used to
   indicate interleaving offset; if set to 1, displacement fields of
   eight bits are used (see Section 4.3.2.2). For payloads in the
   basic mode, this bit SHALL be set to 0 and SHALL be ignored by the
   receiver.

Note that frames employing different ISF values require encapsulation
in separate packets. Thus, special considerations apply when
generating interleaved packets and an ISF change is executed. In
particular, frames that, according to the previously used
interleaving pattern, would be aggregated into a single packet have
to be separated into different packets, so that the aforementioned
condition (all frames in a packet share the ISF) remains true. A
naive implementation that splits the frames with different ISF into
different packets can result in up to twice the number of RTP
packets, when compared to an optimal interleaved solution.
Alteration of the interleaving before and after the ISF change may
reduce the need for extra RTP packets.

4.3.2. The Payload Table of Contents

The table of contents (ToC) consists of a list of entries, each entry
corresponds to a group of audio frames carried in the payload, as
depicted below.

```
+----------------+----------------+- ... -+----------------+
|  ToC entry #1  |  ToC entry #2  |         ToC entry #N  |
+----------------+----------------+- ... -+----------------+
```

When multiple groups of frames are present in a payload, the ToC
entries SHALL be placed in the packet in order of increasing RTP
timestamp value (modulo 2^32) of the first transport frame the TOC
entry represents.

4.3.2.1. ToC Entry in the Basic Mode

A ToC entry of a payload in the basic mode has the following format:

```
 0                   1
 0 1 2 3 4 5 6 7 8 9 0 1 2 3 4 5
+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
|F| Frame Type  |    #frames    |
+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
```

F (1 bit): If set to 1, indicates that this ToC entry is followed by
   another ToC entry; if set to 0, indicates that this ToC entry is
   the last one in the ToC.

Frame Type (FT) (7 bits): Indicates the audio codec frame type used
   for the group of frames referenced by this ToC entry. FT
   designates the combination of AMR-WB+ core and stereo rate, one of
   the special AMR-WB+ frame types, the AMR-WB rate, or comfort
   noise, as specified by Table 25 in [1].

#frames (8 bits): Indicates the number of frames in the group
   referenced by this ToC entry. ToC entries with this field equal
   to 0 (which would indicate zero frames) SHALL NOT be used, and
   received packets with such a TOC entry SHALL be discarded.

4.3.2.2. ToC Entry in the Interleaved Mode

   Two different ToC entry formats are defined in interleaved mode.
   They differ in the length of the displacement field, 4 bits or 8
   bits. The L-bit in the payload header differentiates between the two
   modes.

   If L=0, a ToC entry has the following format:

```
 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
+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
|F| Frame Type  |    #frames    | DIS1 |  ...  | DISi |  ...  |
+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
|  ...  |  ...  | DISn |  Padd |
+-+-+-+-+-+-+-+-+-+-+-+-+-+
```

   F (1 bit): See definition in 4.3.2.1.

   Frame Type (FT) (7 bits): See definition in 4.3.2.1.

   #frames (8 bits): See definition in 4.3.2.1.

   DIS1...DISn (4 bits): A list of n (n=#frames) displacement fields
      indicating the displacement of the i:th (i=1..n) audio frame
      relative to the preceding audio frame in the payload, in units of
      frames. The four-bit unsigned integer displacement values may be
      between 0 and 15, indicating the number of audio frames in
      decoding order between the (i-1):th and the i:th frame in the
      payload. Note that for the first ToC entry of the payload, the
      value of DIS1 is meaningless. It SHALL be set to zero by a sender
      and SHALL be ignored by a receiver. This frame's location in the
      decoding order is uniquely defined by the RTP timestamp and TFI in
      the payload header. Note also that for subsequent ToC entries,
      DIS1 indicates the number of frames between the last frame of the
      previous group and the first frame of this group.

   Padd (4 bits): To ensure octet alignment, four padding bits SHALL be
      included at the end of the ToC entry in case there is odd number
      of frames in the group referenced by this entry. These bits SHALL
      be set to zero and SHALL be ignored by the receiver. If a group
      containing an even number of frames is referenced by this ToC
      entry, these padding bits SHALL NOT be included in the payload.

   If L=1, a ToC entry has the following format:

```
 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
+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
|F| Frame Type  |    #frames    |     DIS1      |     ...       |
+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
|     ...       |     DISn      |
+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
```

   F (1 bit): See definition in 4.3.2.1.

   Frame Type (FT) (7 bits): See definition in 4.3.2.1.

   #frames (8 bits): See definition in 4.3.2.1.

   DIS1...DISn (8 bits): A list of n (n=#frames) displacement fields
      indicating the displacement of the i:th (i=1..n) audio frame
      relative to the preceding audio frame in the payload, in units of
      frames. The eight-bit unsigned integer displacement values may be
      between 0 and 255, indicating the number of audio frames in
      decoding order between the (i-1):th and the i:th frame in the
      payload. Note that for the first ToC entry of the payload, the
      value of DIS1 is meaningless. It SHALL be set to zero by a sender
      and SHALL be ignored by a receiver. This frame's location in the
      decoding order is uniquely defined by the RTP timestamp and TFI in
      the payload header. Note also that for subsequent ToC entries,
      DIS1 indicates the displacement between the last frame of the
      previous group and the first frame of this group.

4.3.2.3. RTP Timestamp Derivation

   The RTP Timestamp value for a frame SHALL be the timestamp value of
   the first audio sample encoded in the frame. The timestamp value for
   a frame is derived differently depending on the payload mode, basic
   or interleaved. In both cases, the first frame in a compound packet
   has an RTP timestamp equal to the one received in the RTP header. In
   the basic mode, the RTP time for any subsequent frame is derived in
   two steps. First, the sum of the frame durations (see Table 1) of
   all the preceding frames in the payload is calculated. Then, this
   sum is added to the RTP header timestamp value. For example, let's

assume that the RTP Header timestamp value is 12345, the payload
carries four frames, and the frame duration is 16 ms (ISF = 32 kHz)
corresponding to 1152 timestamp ticks. Then the RTP timestamp of the
fourth frame in the payload is 12345 + 3 * 1152 = 15801.

In interleaved mode, the RTP timestamp for each frame in the payload
is derived from the RTP header timestamp and the sum of the time
offsets of all preceding frames in this payload. The frame
timestamps are computed based on displacement fields and the frame
duration derived from the ISF value. Note that the displacement in
time between frame i-1 and frame i is (DISi + 1) * frame duration
because the duration of the (i-1):th must also be taken into account.
The timestamp of the first frame of the first group of frames (TS(1))
(i.e., the first frame of the payload) is the RTP header timestamp.
For subsequent frames in the group, the timestamp is computed by

    TS(i) = TS(i-1) + (DISi + 1) * frame duration,   2 < i < n

For subsequent groups of frames, the timestamp of the first frame is
computed by

    TS(1) = TSprev + (DIS1 + 1) * frame duration,

where TSprev denotes the timestamp of the last frame in the previous
group. The timestamps of the subsequent frames in the group are
computed in the same way as for the first group.

The following example derives the RTP timestamps for the frames in an
interleaved mode payload having the following header and ToC
information:

RTP header timestamp: 12345
ISF = 32 kHz
Frame 1 displacement field: DIS1 = 0
Frame 2 displacement field: DIS2 = 6
Frame 3 displacement field: DIS3 = 4
Frame 4 displacement field: DIS4 = 7

Assuming an ISF of 32 kHz, which implies a frame duration of 16 ms,
one frame lasts 1152 ticks. The timestamp of the first frame in the
payload is the RTP timestamp, i.e., TS(1) = RTP TS. Note that the
displacement field value for this frame must be ignored. For the
second frame in the payload, the timestamp can be calculated as TS(2)
= TS(1) + (DIS2 + 1) * 1152 = 20409. For the third frame, the
timestamp is TS(3) = TS(2) + (DIS3 + 1) * 1152 = 26169. Finally, for
the fourth frame of the payload, we have TS(4) = TS(3) + (DIS4 + 1) *
1152 = 35385.

4.3.2.4. Frame Type Considerations

   The value of Frame Type (FT) is defined in Table 25 in [1]. FT=14
   (AUDIO_LOST) is used to denote frames that are lost. A NO_DATA
   (FT=15) frame could result from two situations: First, that no data
   has been produced by the audio encoder; and second, that no data is
   transmitted in the current payload. An example for the latter would
   be that the frame in question has been or will be sent in an earlier
   or later packet. The duration for these non-included frames is
   dependent on the internal sampling frequency indicated by the ISF
   field.

   For frame types with index 0-13, the ISF field SHALL be set 0. The
   frame duration for these frame types is fixed to 20 ms in time, i.e.,
   1440 ticks in 72 kHz. For payloads containing only frames of type
   0-9, the TFI field SHALL be set to 0 and SHALL be ignored by the
   receiver. In a payload combining frames of type 0-9 and 10-13, the
   TFI values need to be set to match the transport frames of type
   10-13. Thus, frames of type 0-9 will also have a derived TFI, which
   is ignored.

4.3.2.5. Other TOC Considerations

   If a ToC entry with an undefined FT value is received, the whole
   packet SHALL be discarded. This is to avoid the loss of data
   synchronization in the depacketization process, which can result in a
   severe degradation in audio quality.

   Packets containing only NO_DATA frames SHOULD NOT be transmitted.
   Also, NO_DATA frames at the end of a frame sequence to be carried in
   a payload SHOULD NOT be included in the transmitted packet. The
   AMR-WB+ SCR/DTX is identical with AMR-WB SCR/DTX described in [5] and
   can only be used in combination with the AMR-WB frame types (0-8).

   When multiple groups of frames are present, their ToC entries SHALL
   be placed in the ToC in order of increasing RTP timestamp value
   (modulo 2^32) of the first transport frame the TOC entry represents,
   independent of the payload mode. In basic mode, the frames SHALL be
   consecutive in time, while in interleaved mode the frames MAY not
   only be non-consecutive in time but MAY even have varying inter-frame
   distances.

4.3.2.6. ToC Examples

   The following example illustrates a ToC for three audio frames in
   basic mode. Note that in this case all audio frames are encoded
   using the same frame type, i.e., there is only one ToC entry.

```
 0                     1
 0 1 2 3 4 5 6 7 8 9 0 1 2 3 4 5
+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
|0| Frame Type1 |  #frames = 3  |
+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
```

The next example depicts a ToC of three entries in basic mode. Note
that in this case the payload also carries three frames, but three
ToC entries are needed because the frames of the payload are encoded
using different frame types.

```
 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| Frame Type1 |  #frames = 1  |1| Frame Type2 |  #frames = 1  |
+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
|0| Frame Type3 |  #frames = 1  |
+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
```

The following example illustrates a ToC with two entries in
interleaved mode using four-bit displacement fields. The payload
includes two groups of frames, the first one including a single
frame, and the other one consisting of two frames.

```
 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| Frame Type1 |  #frames = 1  | DIS1  |  padd |0| Frame Type2 |
+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
|  #frames = 2  | DIS1  | DIS2  |
+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
```

4.3.3. Audio Data

   Audio data of a payload consists of zero or more audio frames, as
   described in the ToC of the payload.

   ToC entries with FT=14 or 15 represent frame types with a length of
   0. Hence, no data SHALL be placed in the audio data section to
   represent frames of this type.

   As already discussed, each audio frame of an extension frame type
   represents an AMR-WB+ transport frame corresponding to the encoding
   of 512 samples of audio, sampled with the internal sampling frequency
   specified by the ISF indicator. As an exception, frame types with
   index 10-13 are only capable of using a single internal sampling
   frequency (25600 Hz). The encoding rates (combination of core bit-
   rate and stereo bit-rate) are indicated in the frame type field of

the corresponding ToC entry. The octet length of the audio frame is
implicitly defined by the frame type field and is given in Tables 21
and 25 of [1]. The order and numbering notation of the bits are as
specified in [1]. For the AMR-WB+ extension frame types and comfort
noise frames, the bits are in the order produced by the encoder. The
last octet of each audio frame MUST be padded with zeroes at the end
if not all bits in the octet are used. In other words, each audio
frame MUST be octet-aligned.

4.3.4. Methods for Forming the Payload

   The payload begins with the payload header, followed by the table of
   contents, which consists of a list of ToC entries.

   The audio data follows the table of contents. All the octets
   comprising an audio frame SHALL be appended to the payload as a unit.
   The audio frames are packetized in timestamp order within each group
   of frames (per ToC entry). The groups of frames are packetized in
   the same order as their corresponding ToC entries. Note that there
   are no data octets in a group having a ToC entry with FT=14 or FT=15.

4.3.5. Payload Examples

4.3.5.1. Example 1: Basic Mode Payload Carrying Multiple Frames Encoded
          Using the Same Frame Type

   Figure 4 depicts a payload that carries three AMR-WB+ frames encoded
   using 14 kbit/s frame type (FT=26) with a frame length of 280 bits
   (35 bytes). The internal sampling frequency in this example is 25.6
   kHz (ISF = 8). The TFI for the first frame is 2, indicating that the
   first transport frame in this payload is the third in a super-frame.
   Since this payload is in the basic mode, the subsequent frames of the
   payload are consecutive frames in decoding order, i.e., the fourth
   transport frame of the current super-frame and the first transport
   frame of the next super-frame. Note that because the frames are all
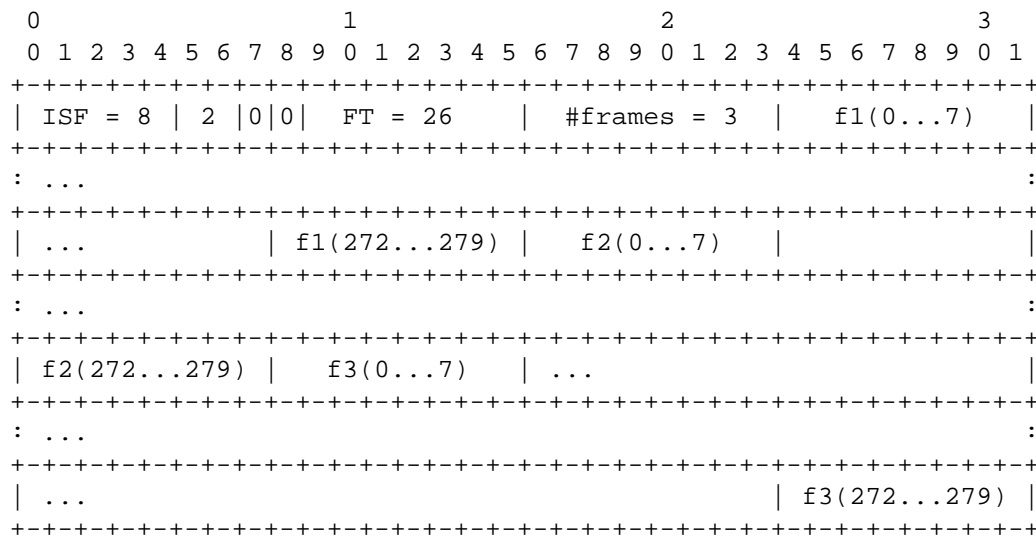   encoded using the same frame type, only one ToC entry is required.

```
     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
    +-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
    | ISF = 8 | 2 |0|0|  FT = 26     | #frames = 3  |   f1(0...7)   |
    +-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
    : ...                                                          :
    +-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
    | ...           | f1(272...279) |   f2(0...7)   |              |
    +-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
    : ...                                                          :
    +-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
    | f2(272...279) |   f3(0...7)   | ...                          |
    +-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
    : ...                                                          :
    +-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
    | ...                                          | f3(272...279) |
    +-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
```

        Figure 4: An example of a basic mode payload carrying three frames
                  of the same frame type

4.3.5.2.   Example 2: Basic Mode Payload Carrying Multiple Frames Encoded
           Using Different Frame Types

   Figure 5 depicts a payload that carries three AMR-WB+ frames; the
   first frame is encoded using 18.4 kbit/s frame type (FT=33) with a
   frame length of 368 bits (46 bytes), and the two subsequent frames
   are encoded using 20 kbit/s frame type (FT=35) having frame length of
   400 bits (50 bytes). The internal sampling frequency in this example
   is 32 kHz (ISF = 10), implying the overall bit-rates of 23 kbit/s for
   the first frame of the payload, and 25 kbit/s for the subsequent
   frames. The TFI for the first frame is 3, indicating that the first
   transport frame in this payload is the fourth in a super-frame.
   Since this is a payload in the basic mode, the subsequent frames of
   the payload are consecutive frames in decoding order, i.e., the first
   and second transport frames of the current super-frame. Note that
   since the payload carries two different frame types, there are two
   ToC entries.

```
 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
+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
| ISF=10 | 3 |0|1|  FT = 33    | #frames = 1  |0|  FT = 35     |
+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
|  #frames = 2  |   f1(0...7)  | ...                           |
+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
: ...                                                          :
+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
| ...                           | f1(360...367) |   f2(0...7)   |
+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
: ...                                                          :
+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
| f2(392...399) |   f3(0...7)   | ...                          |
+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
: ...                                                          :
+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
| ...                           | f3(392...399) |
+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
```

        Figure 5: An example of a basic mode payload carrying three frames
                employing two different frame types

4.3.5.3. Example 3: Payload in Interleaved Mode

   The example in Figure 6 depicts a payload in interleaved mode,
   carrying four frames encoded using 32 kbit/s frame type (FT=47) with
   frame length of 640 bits (80 bytes). The internal sampling frequency
   is 38.4 kHz (ISF = 13), implying a bit-rate of 48 kbit/s for all
   frames in the payload. The TFI for the first frame is 0; hence, it
   is the first transport frame of a super-frame. The displacement
   fields for the subsequent frames are DIS2=18, DIS3=15, and DIS4=10,
   which indicates that the subsequent frames have the TFIs of 3, 3, and
   2, respectively. The long displacement field flag L in the payload
   header is set to 1, which results in the use of eight bits for the
   displacement fields in the ToC entry. Note that since all frames of
   this payload are encoded using the same frame type, there is need
   only for a single ToC entry. Furthermore, the displacement field for
   the first frame (corresponding to the first ToC entry with DIS1=0)
   must be ignored, since its timestamp and TFI are defined by the RTP
   timestamp and the TFI found in the payload header.

   The RTP timestamp values of the frames in this example are:

   Frame1: TS1 = RTP Timestamp
   Frame2: TS2 = TS1 + 19 * 960
   Frame3: TS3 = TS2 + 16 * 960
   Frame4: TS4 = TS3 + 11 * 960

```
 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
+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
|  ISF=13  | 0 |1|0|   FT = 47    |  #frames = 4  |  DIS1 = 0    |
+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
|   DIS2 = 18   |   DIS3 = 15    |   DIS4 = 10   |   f1(0...7)   |
+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
: ...                                                           :
+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
| ...                           | f1(632...639) |   f2(0...7)   |
+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
: ...                                                           :
+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
| ...                           | f2(632...639) |   f3(0...7)   |
+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
: ...                                                           :
+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
| ...                           | f3(632...639) |   f4(0...7)   |
+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
: ...                                                           :
+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
| ...                           | f4(632...639) |
+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
```

    Figure 6: An example of an interleaved mode payload carrying four
              frames at the same frame type

4.4. Interleaving Considerations

   The use of interleaving requires further considerations. As
   presented in the example in Section 3.6.2, a given interleaving
   pattern requires a certain amount of the deinterleaving buffer. This
   buffer space, expressed in a number of transport frame slots, is
   indicated by the "interleaving" media type parameter. The number of
   frame slots needed can be converted into actual memory requirements
   by considering the 80 bytes per frame used by the largest combination
   of AMR-WB+'s core and stereo rates.

   The information about the frame buffer size is not always sufficient
   to determine when it is appropriate to start consuming frames from
   the interleaving buffer. There are two cases in which additional
   information is needed: first, when switching of the ISF occurs, and
   second, when the interleaving pattern changes. The "int-delay" media
   type parameter is defined to convey this information. It allows a
   sender to indicate the minimal media time that needs to be present in
   the buffer before the decoder can start consuming frames from the
   buffer. Because the sender has full control over ISF changes and the
   interleaving pattern, it can calculate this value.

In certain cases (for example, if joining a multicast session with
interleaving mid-session), a receiver may initially receive only part
of the packets in the interleaving pattern. This initial partial
reception (in frame sequence order) of frames can yield too few
frames for acceptable quality from the audio decoding. This problem
also arises when using encryption for access control, and the
receiver does not have the previous key.

Although the AMR-WB+ is robust and thus tolerant to a high random
frame erasure rate, it would have difficulties handling consecutive
frame losses at startup. Thus, some special implementation
considerations are described. In order to handle this type of
startup efficiently, it must be noted that decoding is only possible
to start at the beginning of a super-frame, and that holds true even
if the first transport frame is indicated as lost. Secondly,
decoding is only RECOMMENDED to start if at least 2 transport frames
are available out of the 4 belonging to that super-frame.

After receiving a number of packets, in the worst case as many
packets as the interleaving pattern covers, the previously described
effects disappear and normal decoding is resumed.

Similar issues arise when a receiver leaves a session or has lost
access to the stream. If the receiver leaves the session, this would
be a minor issue since playout is normally stopped. It is also a
minor issue for the case of lost access, since the AMR-WB+ error
concealment will fade out the audio if massive consecutive losses are
encountered.

The sender can avoid this type of problem in many sessions by
starting and ending interleaving patterns correctly when risks of
losses occur. One such example is a key-change done for access
control to encrypted streams. If only some keys are provided to
clients and there is a risk of their receiving content for which they
do not have the key, it is recommended that interleaving patterns not
overlap key changes.

4.5. Implementation Considerations

An application implementing this payload format MUST understand all
the payload parameters. Any mapping of the parameters to a signaling
protocol MUST support all parameters. So an implementation of this
payload format in an application using SDP is required to understand
all the payload parameters in their SDP-mapped form. This
requirement ensures that an implementation always can decide whether
it is capable of communicating.

Both basic and interleaved mode SHALL be implemented. The
implementation burden of both is rather small, and requiring both
ensures interoperability. As the AMR-WB+ codec contains the full
functionality of the AMR-WB codec, it is RECOMMENDED to also
implement the payload format in RFC 3267 [7] for the AMR-WB frame
types when implementing this specification. Doing so makes
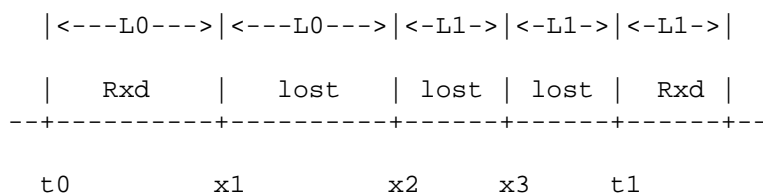interoperability with devices that only support AMR-WB more likely.

The switching of ISF, when combined with packet loss, could result in
concealment using the wrong audio frame length. This can occur if
packet losses result in lost frames directly after the point of ISF
change. The packet loss would prevent the receiver from noticing the
changed ISF and thereby conceal the lost transport frame with the
previous ISF, instead of the new one. Although always later
detectable, such an error results in frame boundary misalignment,
which can cause audio distortions and problems with synchronization,
as too many or too few audio samples were created. This problem can
be mitigated in most cases by performing ISF recovery prior to
concealment as outlined in Section 4.5.1.

4.5.1. ISF Recovery in Case of Packet Loss

In case of packet loss, it is important that the AMR-WB+ decoder
initiates a proper error concealment to replace the frames carried in
the lost packet. A loss concealment algorithm requires a codec
framing that matches the timestamps of the correctly received frames.
Hence, it is necessary to recover the timestamps of the lost frames.
Doing so is non-trivial because the codec frame length that is
associated with the ISF may have changed during the frame loss.

In the following, the recovery of the timestamp information of lost
frames is illustrated by the means of an example. Two frames with
timestamps t0 and t1 have been received properly, the first one being
the last packet before the loss, and the latter one being the first
packet after the loss period. The ISF values for these packets are
isf0 and isf1, respectively. The TFIs of these frames are tfi0 and
tfi1, respectively. The associated frame lengths (in timestamp
ticks) are given as L0 and L1, respectively. In this example three
frames with timestamps x1 - x3 have been lost. The example further
assumes that ISF changes once from isf0 to isf1 during the frame loss
period, as shown in the figure below.

Since not all information required for the full recovery of the
timestamps is generally known in the receiver, an algorithm is needed
to estimate the ISF associated with the lost frames. Also, the
number of lost frames needs to be recovered.

```
   |<---L0--->|<---L0--->|<-L1->|<-L1->|<-L1->|

   |  Rxd    |  lost    | lost | lost |  Rxd |
 --+---------+---------+------+------+------+--

   t0         x1         x2     x3     t1
```

Example Algorithm:

```
Start:                              # check for frame loss
If (t0 + L0) == t1 Then goto End    # no frame loss

Step 1:                             # check case with no ISF change
If (isf0 != isf1) Then goto Step 2  # At least one ISF change
If (isFractional(t1 - t0)/L0) Then goto Step 3
                                    # More than 1 ISF change


Return recovered timestamps as
x(n) = t0 + n*L1 and associated ISF equal to isf0,
for 0 < n < (t1 - t0)/L0
goto End

Step 2:
Loop initialization: n := 4 - tfi0 mod 4
While n <= (t1-t0)/L0
  Evaluate m := (t1 - t0 - n*L0)/L1
  If (isInteger(m) AND ((tfi0+n+m) mod 4 == tfi1)) Then goto found;
  n := n+4
  endloop
goto step 3                         # More than 1 ISF change

found:
Return recovered timestamps and ISFs as
x(i) = t0 + i*L0 and associated ISF equal to isf0, for 0 < i <= n
x(i) = t0 + n*L0 + (i-n)*L1 and associated ISF equal to isf1,
for n < i <= n+m
goto End

Step 3:
More than 1 ISF change has occurred.  Since ISF changes can be
assumed to be infrequent, such a situation occurs only if long
sequences of frames are lost.  In that case it is probably not useful
to try to recover the timestamps of the lost frames.  Rather, the
AMR-WB+ decoder should be reset, and decoding should be resumed
starting with the frame with timestamp t1.

End:
```

The above algorithm still does not solve the issue when the receiver
buffer depth is shallower than the loss burst. In this kind of case,
where the concealment must be done without any knowledge about future
frames, the concealment may result in loss of frame boundary
alignment. If that occurs, it may be necessary to reset and restart
the codec to perform resynchronization.

4.5.2. Decoding Validation

If the receiver finds a mismatch between the size of a received
payload and the size indicated by the ToC of the payload, the
receiver SHOULD discard the packet. This is recommended because
decoding a frame parsed from a payload based on erroneous ToC data
could severely degrade the audio quality.

5. Congestion Control

The general congestion control considerations for transporting RTP
data apply; see RTP [3] and any applicable RTP profile like AVP [9].
However, the multi-rate capability of AMR-WB+ audio coding provides a
mechanism that may help to control congestion, since the bandwidth
demand can be adjusted (within the limits of the codec) by selecting
a different coding frame type or lower internal sampling rate.

The number of frames encapsulated in each RTP payload highly
influences the overall bandwidth of the RTP stream due to header
overhead constraints. Packetizing more frames in each RTP payload
can reduce the number of packets sent and hence the header overhead,
at the expense of increased delay and reduced error robustness.

If forward error correction (FEC) is used, the amount of FEC-induced
redundancy needs to be regulated such that the use of FEC itself does
not cause a congestion problem.

6. Security Considerations

RTP packets using the payload format defined in this specification
are subject to the general security considerations discussed in RTP
[3] and any applicable profile such as AVP [9] or SAVP [10]. As this
format transports encoded audio, the main security issues include
confidentiality, integrity protection, and data origin authentication
of the audio itself. The payload format itself does not have any
built-in security mechanisms. Any suitable external mechanisms, such
as SRTP [10], MAY be used.

This payload format and the AMR-WB+ decoder do not exhibit any
significant non-uniformity in the receiver-side computational
complexity for packet processing, and thus are unlikely to pose a
denial-of-service threat due to the receipt of pathological data.

6.1. Confidentiality

In order to ensure confidentiality of the encoded audio, all audio
data bits MUST be encrypted. There is less need to encrypt the
payload header or the table of contents since they only carry
information about the frame type. This information could also be
useful to a third party, for example, for quality monitoring.

The use of interleaving in conjunction with encryption can have a
negative impact on confidentiality, for a short period of time.
Consider the following packets (in brackets) containing frame numbers
as indicated: {10, 14, 18}, {13, 17, 21}, {16, 20, 24} (a popular
continuous diagonal interleaving pattern). The originator wishes to
deny some participants the ability to hear material starting at time
16. Simply changing the key on the packet with the timestamp at or
after 16, and denying that new key to those participants, does not
achieve this; frames 17, 18, and 21 have been supplied in prior
packets under the prior key, and error concealment may make the audio
intelligible at least as far as frame 18 or 19, and possibly further.

6.2. Authentication and Integrity

To authenticate the sender of the speech, an external mechanism MUST
be used. It is RECOMMENDED that such a mechanism protects both the
complete RTP header and the payload (speech and data bits).

Data tampering by a man-in-the-middle attacker could replace audio
content and also result in erroneous depacketization/decoding that
could lower the audio quality.

7. Payload Format Parameters

This section defines the parameters that may be used to select
features of the AMR-WB+ payload format. The parameters are defined
as part of the media type registration for the AMR-WB+ audio codec.
A mapping of the parameters into the Session Description Protocol
(SDP) [6] is also provided for those applications that use SDP.
Equivalent parameters could be defined elsewhere for use with control
protocols that do not use MIME or SDP.

The data format and parameters are only specified for real-time
transport in RTP.

7.1. Media Type Registration

   The media type for the Extended Adaptive Multi-Rate Wideband
   (AMR-WB+) codec is allocated from the IETF tree, since AMR-WB+ is
   expected to be a widely used audio codec in general streaming
   applications.

   Note: Parameters not listed below MUST be ignored by the receiver.

   Media Type name: audio

   Media subtype name: AMR-WB+

   Required parameters:

   None

   Optional parameters:

   channels: The maximum number of audio channels used by the
                    audio frames. Permissible values are 1 (mono) or 2
                    (stereo). If no parameter is present, the maximum
                    number of channels is 2 (stereo). Note: When set to
                    1, implicitly the stereo frame types cannot be used.

   interleaving: Indicates that interleaved mode SHALL
                    be used for the payload. The parameter specifies
                    the number of transport frame slots required in a
                    deinterleaving buffer (including the frame that is
                    ready to be consumed). Its value is equal to one
                    plus the maximum number of frames that precede any
                    frame in transmission order and follow the frame in
                    RTP timestamp order. The value MUST be greater than
                    zero. If this parameter is not present,
                    interleaved mode SHALL NOT be used.

   int-delay: The minimal media time delay in RTP timestamp ticks
                    that is needed in the deinterleaving buffer, i.e.,
                    the difference in RTP timestamp ticks between the
                    earliest and latest audio frame present in the
                    deinterleaving buffer.

   ptime: See Section 6 in RFC 2327 [6].

   maxptime: See Section 8 in RFC 3267 [7].

   Restriction on Usage:
                This type is only defined for transfer via RTP (STD 64).

   Encoding considerations:
              An RTP payload according to this format is binary data
              and thus may need to be appropriately encoded in non-
              binary environments. However, as long as used within
              RTP, no encoding is necessary.

   Security considerations:
              See Section 6 of RFC 4352.

   Interoperability considerations:
              To maintain interoperability with AMR-WB-capable end-
              points, in cases where negotiation is possible and the
              AMR-WB+ end-point supporting this format also supports
              RFC 3267 for AMR-WB transport, an AMR-WB+ end-point
              SHOULD declare itself also as AMR-WB capable (i.e.,
              supporting also "audio/AMR-WB" as specified in RFC
              3267).

              As the AMR-WB+ decoder is capable of performing stereo
              to mono conversions, all receivers of AMR-WB+ should be
              able to receive both stereo and mono, although the
              receiver is only capable of playout of mono signals.

   Public specification:
              RFC 4352
              3GPP TS 26.290, see reference [1] of RFC 4352

   Additional information:
              This MIME type is not applicable for file storage.
              Instead, file storage of AMR-WB+ encoded audio is
              specified within the 3GPP-defined ISO-based multimedia
              file format defined in 3GPP TS 26.244; see reference
              [14] of RFC 4352. This file format has the MIME types
              "audio/3GPP" or "video/3GPP" as defined by RFC 3839
              [15].

   Person & email address to contact for further information:
              magnus.westerlund@ericsson.com
              ari.lakaniemi@nokia.com

   Intended usage: COMMON.
              It is expected that many IP-based streaming
              applications will use this type.

   Change controller:
              IETF Audio/Video Transport working group delegated from
              the IESG.

7.2. Mapping Media Type Parameters into SDP

   The information carried in the media type specification has a
   specific mapping to fields in the Session Description Protocol (SDP)
   [6], which is commonly used to describe RTP sessions. When SDP is
   used to specify an RTP session using this RTP payload format, the
   mapping is as follows:

   - The media type ("audio") is used in SDP "m=" as the media name.

   - The media type (payload format name) is used in SDP "a=rtpmap" as
      the encoding name. The RTP clock rate in "a=rtpmap" SHALL be
      72000 for AMR-WB+, and the encoding parameter number of channels
      MUST either be explicitly set to 1 or 2, or be omitted, implying
      the default value of 2.

   - The parameters "ptime" and "maxptime" are placed in the SDP
      attributes "a=ptime" and "a=maxptime", respectively.

   - Any remaining parameters are placed in the SDP "a=fmtp" attribute
      by copying them directly from the MIME media type string as a
      semicolon-separated list of parameter=value pairs.

7.2.1. Offer-Answer Model Considerations

   To achieve good interoperability in an Offer-Answer [8] negotiation
   usage, the following considerations should be taken into account:

   For negotiable offer/answer usage the following interpretation rules
   SHALL be applied:

   - The "interleaving" parameter is symmetric, thus requiring that the
      answerer must also include it for the answer to an offered payload
      type that contains the parameter. However, the buffer space value
      is declarative in usage in unicast. For multicast usage, the same
      value in the response is required in order to accept the payload
      type. For streams declared as sendrecv or recvonly: The receiver
      will accept reception of streams using the interleaved mode of the
      payload format. The value declares the amount of buffer space the
      receiver has available for the sender to utilize. For sendonly
      streams, the parameter indicates the desired configuration and
      amount of buffer space. An answerer is RECOMMENDED to respond
      using the offered value, if capable of using it.

- The "int-delay" parameter is declarative. For streams declared as
  sendrecv or recvonly, the value indicates the maximum initial
  delay the receiver will accept in the deinterleaving buffer. For
  sendonly streams, the value is the amount of media time the sender
  desires to use. The value SHOULD be copied into any response.

- The "channels" parameter is declarative. For "sendonly" streams,
  it indicates the desired channel usage, stereo and mono, or mono
  only. For "recvonly" and "sendrecv" streams, the parameter
  indicates what the receiver accepts to use. As any receiver will
  be capable of receiving stereo frame type and perform local mixing
  within the AMR-WB+ decoder, there is normally only one reason to
  restrict to mono only: to avoid spending bit-rate on data that are
  not utilized if the front-end is only capable of mono.

- The "ptime" parameter works as indicated by the offer/answer model
  [8]; "maxptime" SHALL be used in the same way.

- To maintain interoperability with AMR-WB in cases where
  negotiation is possible, an AMR-WB+ capable end-point that also
  implements the AMR-WB payload format [7] is RECOMMENDED to declare
  itself capable of AMR-WB as it is a subset of the AMR-WB+ codec.

In declarative usage, like SDP in RTSP [16] or SAP [17], the
following interpretation of the parameters SHALL be done:

- The "interleaving" parameter, if present, configures the payload
  format in that mode, and the value indicates the number of frames
  that the deinterleaving buffer is required to support to be able
  to handle this session correctly.

- The "int-delay" parameter indicates the initial buffering delay
  required to receive this stream correctly.

- The "channels" parameter indicates if the content being
  transmitted can contain either both stereo and mono rates, or only
  mono.

- All other parameters indicate values that are being used by the
  sending entity.

7.2.2. Examples

   One example of an SDP session description utilizing AMR-WB+ mono and
   stereo encoding follows.

    m=audio 49120 RTP/AVP 99
    a=rtpmap:99 AMR-WB+/72000/2
    a=fmtp:99 interleaving=30; int-delay=86400
    a=maxptime:100

   Note that the payload format (encoding) names are commonly shown in
   uppercase. Media subtypes are commonly shown in lowercase. These
   names are case-insensitive in both places. Similarly, parameter
   names are case-insensitive both in MIME types and in the default
   mapping to the SDP a=fmtp attribute.

8. IANA Considerations

   The IANA has registered one new MIME subtype (audio/amr-wb+); see
   Section 7.

9. Contributors

   Daniel Enstrom has contributed in writing the codec introduction
   section. Stefan Bruhn has contributed by writing the ISF recovery
   algorithm.

10. Acknowledgements

   The authors would like to thank Redwan Salami and Stefan Bruhn for
   their significant contributions made throughout the writing and
   reviewing of this document. Dave Singer contributed by reviewing and
   suggesting improved language. Anisse Taleb and Ingemar Johansson
   contributed by implementing the payload format and thus helped locate
   some flaws. We would also like to acknowledge Qiaobing Xie, coauthor
   of RFC 3267, on which this document is based.

11. References

11.1. Normative References

   [1] 3GPP TS 26.290 "Audio codec processing functions; Extended
        Adaptive Multi-Rate Wideband (AMR-WB+) codec; Transcoding
        functions", version 6.3.0 (2005-06), 3rd Generation Partnership
        Project (3GPP).

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

   [3] Schulzrinne, H., Casner, S., Frederick, R., and V. Jacobson,
        "RTP: A Transport Protocol for Real-Time Applications", STD 64,
        RFC 3550, July 2003.

   [4] 3GPP TS 26.192 "AMR Wideband speech codec; Comfort Noise
        aspects", version 6.0.0 (2004-12), 3rd Generation Partnership
        Project (3GPP).

   [5] 3GPP TS 26.193 "AMR Wideband speech codec; Source Controlled
        Rate operation", version 6.0.0 (2004-12), 3rd Generation
        Partnership Project (3GPP).

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

   [7] Sjoberg, J., Westerlund, M., Lakaniemi, A., and Q. Xie, "Real-
        Time Transport Protocol (RTP) Payload Format and File Storage
        Format for the Adaptive Multi-Rate (AMR) and Adaptive Multi-Rate
        Wideband (AMR-WB) Audio Codecs", RFC 3267, June 2002.

   [8] Rosenberg, J. and H. Schulzrinne, "An Offer/Answer Model with
        Session Description Protocol (SDP)", RFC 3264, June 2002.

   [9] Schulzrinne, H. and S. Casner, "RTP Profile for Audio and Video
        Conferences with Minimal Control", STD 65, RFC 3551, July 2003.

11.2. Informative References

   [10] Baugher, M., McGrew, D., Naslund, M., Carrara, E., and K.
        Norrman, "The Secure Real-time Transport Protocol (SRTP)", RFC
        3711, March 2004.

   [11] Rosenberg, J. and H. Schulzrinne, "An RTP Payload Format for
        Generic Forward Error Correction", RFC 2733, December 1999.

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

   [13] 3GPP TS 26.233 "Packet Switched Streaming service", version
        5.7.0 (2005-03), 3rd Generation Partnership Project (3GPP).

   [14] 3GPP TS 26.244 "Transparent end-to-end packet switched streaming
        service (PSS); 3GPP file format (3GP)", version 6.4.0 (2005-09),
        3rd Generation Partnership Project (3GPP).

   [15] Castagno, R. and D. Singer, "MIME Type Registrations for 3rd
        Generation Partnership Project (3GPP) Multimedia files", RFC
        3839, July 2004.

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

   [17] Handley, M., Perkins, C., and E. Whelan, "Session Announcement
        Protocol", RFC 2974, October 2000.

   [18] 3GPP TS 26.140 "Multimedia Messaging Service (MMS); Media
        formats and codes", version 6.2.0 (2005-03), 3rd Generation
        Partnership Project (3GPP).

   [19] 3GPP TS 26.140 "Multimedia Broadcast/Multicast Service (MBMS);
        Protocols and codecs", version 6.3.0 (2005-12), 3rd Generation
        Partnership Project (3GPP).

   Any 3GPP document can be downloaded from the 3GPP webserver,
   "http://www.3gpp.org/", see specifications.

Authors' Addresses

   Johan Sjoberg
   Ericsson Research
   Ericsson AB
   SE-164 80 Stockholm
   SWEDEN

   Phone: +46 8 7190000
   EMail: Johan.Sjoberg@ericsson.com

   Magnus Westerlund
   Ericsson Research
   Ericsson AB
   SE-164 80 Stockholm
   SWEDEN

   Phone: +46 8 7190000
   EMail: Magnus.Westerlund@ericsson.com

   Ari Lakaniemi
   Nokia Research Center
   P.O. Box 407
   FIN-00045 Nokia Group
   FINLAND

   Phone: +358-71-8008000
   EMail: ari.lakaniemi@nokia.com

   Stephan Wenger
   Nokia Corporation
   P.O. Box 100
   FIN-33721 Tampere
   FINLAND

   Phone: +358-50-486-0637
   EMail: Stephan.Wenger@nokia.com