              RTP Payload Format and File Storage Format for the
    Adaptive Multi-Rate (AMR) and Adaptive Multi-Rate Wideband (AMR-WB)
                              Audio Codecs

Status of This Memo

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

Copyright Notice

Abstract

   This document specifies a Real-time Transport Protocol (RTP) payload
   format to be used for Adaptive Multi-Rate (AMR) and Adaptive Multi-
   Rate Wideband (AMR-WB) encoded speech signals. The payload format is
   designed to be able to interoperate with existing AMR and AMR-WB
   transport formats on non-IP networks. In addition, a file format is
   specified for transport of AMR and AMR-WB speech data in storage mode
   applications such as email. Two separate media type registrations
   are included, one for AMR and one for AMR-WB, specifying use of both
   the RTP payload format and the storage format. This document
   obsoletes RFC 3267.

Table of Contents

1. Introduction

   This document obsoletes RFC 3267 and extends that specification with
   offer/answer rules. See Section 10 for the changes made to this
   format in relation to RFC 3267.

   This document specifies the payload format for packetization of AMR
   and AMR-WB encoded speech signals into the Real-time Transport
   Protocol (RTP) [8]. The payload format supports transmission of
   multiple channels, multiple frames per payload, the use of fast codec
   mode adaptation, robustness against packet loss and bit errors, and
   interoperation with existing AMR and AMR-WB transport formats on
   non-IP networks, as described in Section 3.

   The payload format itself is specified in Section 4. A related file
   format is specified in Section 5 for transport of AMR and AMR-WB
   speech data in storage mode applications such as email. In Section
   8, two separate media type registrations are provided, one for AMR
   and one for AMR-WB.

   Even though this RTP payload format definition supports the transport
   of both AMR and AMR-WB speech, it is important to remember that AMR
   and AMR-WB are two different codecs and they are always handled as
   different payload types in RTP.

2. Conventions and Acronyms

   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 [5].

   The following acronyms are used in this document:

       3GPP   - the Third Generation Partnership Project
       AMR   - Adaptive Multi-Rate (Codec)
       AMR-WB - Adaptive Multi-Rate Wideband (Codec)
       CMR   - Codec Mode Request
       CN   - Comfort Noise
       DTX   - Discontinuous Transmission
       ETSI   - European Telecommunications Standards Institute
       FEC   - Forward Error Correction
       SCR   - Source Controlled Rate Operation
       SID   - Silence Indicator (the frames containing only CN
                   parameters)
       VAD   - Voice Activity Detection
       UED   - Unequal Error Detection
       UEP   - Unequal Error Protection

The term "frame-block" is used in this document to describe the
time-synchronized set of speech frames in a multi-channel AMR or
AMR-WB session. In particular, in an N-channel session, a frame-
block will contain N speech frames, one from each of the channels,
and all N speech frames represents exactly the same time period.

The byte order used in this document is network byte order, i.e., the
most significant byte first. The bit order is also the most
significant bit first. This is presented in all figures as having
the most significant bit leftmost on a line and with the lowest
number. Some bit fields may wrap over multiple lines in which cases
the bits on the first line are more significant than the bits on the
next line.

3. Background on AMR/AMR-WB and Design Principles

AMR and AMR-WB were originally designed for circuit-switched mobile
radio systems. Due to their flexibility and robustness, they are
also suitable for other real-time speech communication services over
packet-switched networks such as the Internet.

Because of the flexibility of these codecs, the behavior in a
particular application is controlled by several parameters that
select options or specify the acceptable values for a variable.
These options and variables are described in general terms at
appropriate points in the text of this specification as parameters to
be established through out-of-band means. In Section 8, all of the
parameters are specified in the form of media subtype registrations
for the AMR and AMR-WB encodings. 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
8.3 provides a mapping of the parameters into the Session Description
Protocol (SDP) [11] for those applications that use SDP.

3.1. The Adaptive Multi-Rate (AMR) Speech Codec

The AMR codec was originally developed and standardized by the
European Telecommunications Standards Institute (ETSI) for GSM
cellular systems. It is now chosen by the Third Generation
Partnership Project (3GPP) as the mandatory codec for third
generation (3G) cellular systems [1].

The AMR codec is a multi-mode codec that supports eight narrow band
speech encoding modes with bit rates between 4.75 and 12.2 kbps. The
sampling frequency used in AMR is 8000 Hz and the speech encoding is
performed on 20 ms speech frames. Therefore, each encoded AMR speech
frame represents 160 samples of the original speech.

Among the eight AMR encoding modes, three are already separately
adopted as standards of their own. Particularly, the 6.7 kbps mode
is adopted as PDC-EFR [18], the 7.4 kbps mode as IS-641 codec in TDMA
[17], and the 12.2 kbps mode as GSM-EFR [16].

3.2. The Adaptive Multi-Rate Wideband (AMR-WB) Speech Codec

The Adaptive Multi-Rate Wideband (AMR-WB) speech codec [3] was
originally developed by 3GPP to be used in GSM and 3G cellular
systems.

Similar to AMR, the AMR-WB codec is also a multi-mode speech codec.
AMR-WB supports nine wide band speech coding modes with respective
bit rates ranging from 6.6 to 23.85 kbps. The sampling frequency
used in AMR-WB is 16000 Hz and the speech processing is performed on
20 ms frames. This means that each AMR-WB encoded frame represents
320 speech samples.

3.3. Multi-Rate Encoding and Mode Adaptation

The multi-rate encoding (i.e., multi-mode) capability of AMR and
AMR-WB is designed for preserving high speech quality under a wide
range of transmission conditions.

With AMR or AMR-WB, mobile radio systems are able to use available
bandwidth as effectively as possible. For example, in GSM it is
possible to dynamically adjust the speech encoding rate during a
session so as to continuously adapt to the varying transmission
conditions by dividing the fixed overall bandwidth between speech
data and error protective coding. This enables the best possible
trade-off between speech compression rate and error tolerance. To
perform mode adaptation, the decoder (speech receiver) needs to
signal the encoder (speech sender) the new mode it prefers. This
mode change signal is called Codec Mode Request or CMR.

Since in most sessions speech is sent in both directions between the
two ends, the mode requests from the decoder at one end to the
encoder at the other end are piggy-backed over the speech frames in
the reverse direction. In other words, there is no out-of-band
signaling needed for sending CMRs.

Every AMR or AMR-WB codec implementation is required to support all
the respective speech coding modes defined by the codec and must be
able to handle mode switching to any of the modes at any time.
However, some transport systems may impose limitations in the number
of modes supported and how often the mode can change due to bandwidth

limitations or other constraints. For this reason, the decoder is
allowed to indicate its acceptance of a particular mode or a subset
of the defined modes for the session using out-of-band means.

For example, the GSM radio link can only use a subset of at most four
different modes in a given session. This subset can be any
combination of the eight AMR modes for an AMR session or any
combination of the nine AMR-WB modes for an AMR-WB session.

Moreover, for better interoperability with GSM through a gateway, the
decoder is allowed to use out-of-band means to set the minimum number
of frames between two mode changes and to limit the mode change among
neighboring modes only.

Section 8 specifies a set of media type parameters that may be used
to signal these mode adaptation controls at session setup.

3.4. Voice Activity Detection and Discontinuous Transmission

Both codecs support voice activity detection (VAD) and generation of
comfort noise (CN) parameters during silence periods. Hence, the
codecs have the option to reduce 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 or AMR-WB frames containing CN
parameters are called Silence Indicator (SID) frames. See more
details about VAD and DTX functionality in [9] and [10].

3.5. Support for Multi-Channel Session

Both the RTP payload format and the storage format defined in this
document support multi-channel audio content (e.g., a stereophonic
speech session).

Although AMR and AMR-WB codecs themselves do not support encoding of
multi-channel audio content into a single bit stream, they can be
used to separately encode and decode each of the individual channels.

To transport (or store) the separately encoded multi-channel content,
the speech frames for all channels that are framed and encoded for
the same 20 ms periods are logically collected in a frame-block.

At the session setup, out-of-band signaling must be used to indicate
the number of channels in the session, and the order of the speech
frames from different channels in each frame-block. When using SDP
for signaling, the number of channels is specified in the rtpmap
attribute and the order of channels carried in each frame-block is

implied by the number of channels as specified in Section 4.1 in
[12].

3.6. Unequal Bit-Error Detection and Protection

The speech bits encoded in each AMR or AMR-WB frame have different
perceptual sensitivity to bit errors. This property has been
exploited in cellular systems to achieve better voice quality by
using unequal error protection and detection (UEP and UED)
mechanisms.

The UEP/UED mechanisms focus the protection and detection of
corrupted bits to the perceptually most sensitive bits in an AMR or
AMR-WB frame. In particular, speech bits in an AMR or AMR-WB frame
are divided into class A, B, and C, where bits in class A are the
most sensitive and bits in class C the least sensitive (see Table 1
below for AMR and [4] for AMR-WB). An AMR or AMR-WB frame is only
declared damaged if there are bit errors found in the most sensitive
bits, i.e., the class A bits. On the other hand, it is acceptable to
have some bit errors in the other bits, i.e., class B and C bits.

Index	Mode	Class A bits	Total speech bits
0	AMR 4.75	42	95
1	AMR 5.15	49	103
2	AMR 5.9	55	118
3	AMR 6.7	58	134
4	AMR 7.4	61	148
5	AMR 7.95	75	159
6	AMR 10.2	65	204
7	AMR 12.2	81	244
8	AMR SID	39	39

Table 1. The number of class A bits for the AMR codec

Moreover, a damaged frame is still useful for error concealment at
the decoder since some of the less sensitive bits can still be used.
This approach can improve the speech quality compared to discarding
the damaged frame.

3.6.1. Applying UEP and UED in an IP Network

To take full advantage of the bit-error robustness of the AMR and
AMR-WB codec, the RTP payload format is designed to facilitate
UEP/UED in an IP network. It should be noted however that the
utilization of UEP and UED discussed below is OPTIONAL.

UEP/UED in an IP network can be achieved by detecting bit errors in
class A bits and tolerating bit errors in class B/C bits of the AMR
or AMR-WB frame(s) in each RTP payload.

Link-layer protocols exist that do not discard packets containing bit
errors, e.g., SLIP and some wireless links. With the Internet
traffic pattern shifting towards a more multimedia-centric one, more
link layers of such nature may emerge in the future. With transport
layer support for partial checksums (for example, those supported by
UDP-Lite [19]), bit error tolerant AMR and AMR-WB traffic could
achieve better performance over these types of links. The
relationship between UDP-Lite's partial checksum at the transport
layer and the checksum coverage provided by the link-layer frame is
described in UDP-Lite specification [19].

There are at least two basic approaches for carrying AMR and AMR-WB
traffic over bit error tolerant IP networks:

a) Utilizing a partial checksum to cover the IP, transport protocol
   (e.g., UDP-Lite), RTP and payload headers, and the most important
   speech bits of the payload. The IP, UDP and RTP headers need to
   be protected, and it is recommended that at least all class A bits
   are covered by the checksum.

b) Utilizing a partial checksum to only cover the IP, transport
   protocol, RTP and payload headers, but an AMR or AMR-WB frame CRC
   to cover the class A bits of each speech frame in the RTP payload.

In either approach, at least part of the class B/C bits are left
without error-check and thus bit error tolerance is achieved.

   Note, it is still important that the network designer pays
   attention to the class B and C residual bit error rate. Though
   less sensitive to errors than class A bits, class B and C bits are
   not insignificant, and undetected errors in these bits cause
   degradation in speech quality. An example of residual error rates
   considered acceptable for AMR in the Universal Mobile
   Telecommunications System (UMTS) can be found in [24] and for
   AMR-WB in [25].

The application interface to the UEP/UED transport protocol (e.g.,
UDP-Lite) may not provide any control over the link error rate,
especially in a gateway scenario. Therefore, it is incumbent upon
the designer of a node with a link interface of this type to choose a
residual bit error rate that is low enough to support applications
such as AMR encoding when transmitting packets of a UEP/UED transport
protocol.

Approach 1 is bit efficient, flexible and simple, but comes with two
disadvantages, namely, a) bit errors in protected speech bits will
cause the payload to be discarded, and b) when transporting multiple
AMR or AMR-WB frames in a RTP payload, there is the possibility that
a single bit error in protected bits will cause all the frames to be
discarded.

These disadvantages can be avoided, if needed, with some overhead in
the form of a frame-wise CRC (Approach 2). In problem a), the CRC
makes it possible to detect bit errors in class A bits and use the
frame for error concealment, which gives a small improvement in
speech quality. For b), when transporting multiple frames in a
payload, the CRCs remove the possibility that a single bit error in a
class A bit will cause all the frames to be discarded. Avoiding that
improves the speech quality when transporting multiple AMR or AMR-WB
frames over links subject to bit errors.

The choice between the above two approaches must be made based on the
available bandwidth, and the desired tolerance to bit errors.
Neither solution is appropriate for all cases. Section 8 defines
parameters that may be used at session setup to choose between these
approaches.

3.7. Robustness against Packet Loss

The payload format supports several means, including forward error
correction (FEC) and frame interleaving, to increase robustness
against packet loss.

3.7.1. Use of Forward Error Correction (FEC)

The simple scheme of repetition of previously sent data is one way of
achieving FEC. Another possible scheme which is more bandwidth
efficient is to use payload-external FEC, e.g., RFC 2733 [23], which
generates extra packets containing repair data. The whole payload
can also be sorted in sensitivity order to support external FEC
schemes using UEP. There is also a work in progress on a generic
version of such a scheme [22] that can be applied to AMR or AMR-WB
payload transport.

With AMR or AMR-WB, it is possible to use the multi-rate capability
of the codec to send redundant copies of a frame using either the
same mode or another mode, e.g., one with lower bandwidth. We
describe such a scheme next.

This involves the simple retransmission of previously transmitted
frame-blocks together with the current frame-block(s). This is done
by using a sliding window to group the speech frame-blocks to send in
each payload. Figure 1 below shows us 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) ---->
```
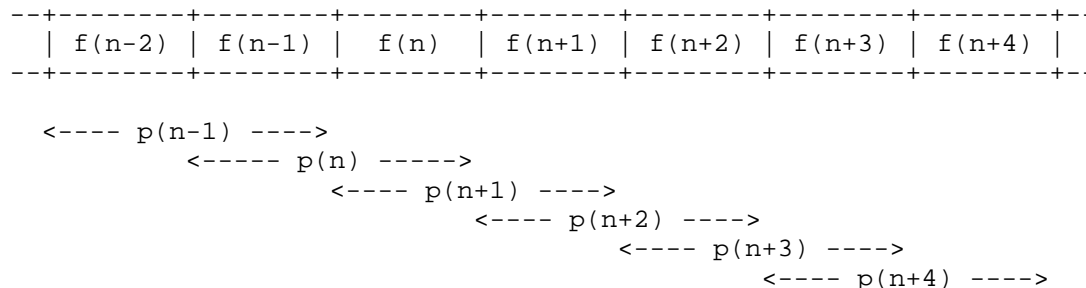
Figure 1: An example of redundant transmission

In this example each frame-block is retransmitted one time in the
following RTP payload packet. Here, f(n-2)..f(n+4) denotes a
sequence of speech frame-blocks, and p(n-1)..p(n+4) a sequence of
payload packets.

The use of this approach does not require signaling at the session
setup. However, a parameter for providing a maximum delay in
transmitting any redundant frame is defined in Section 8. In other
words, the speech sender can choose to use this scheme without
consulting the receiver. This is because a packet containing
redundant frames will not look different from a packet with only new
frames. The receiver may receive multiple copies or versions
(encoded with different modes) of a frame for a certain timestamp if
no packet is lost. If multiple versions of the same speech frame are
received, it is recommended that the mode with the highest rate be
used by the speech decoder.

This redundancy scheme provides the same functionality as the one
described in RFC 2198, "RTP Payload for Redundant Audio Data" [27].
In most cases the mechanism in this payload format is more efficient
and simpler than requiring both endpoints to support RFC 2198 in
addition. There are two situations in which use of RFC 2198 is
indicated: if the spread in time required between the primary and
redundant encodings is larger than the duration of 5 frames, the
bandwidth overhead of RFC 2198 will be lower; or, if a non-AMR codec
is desired for the redundant encoding, the AMR payload format won't
be able to carry it.

The sender is responsible for selecting an appropriate amount of
redundancy based on feedback about the channel, e.g., in RTCP

receiver reports. A sender should not base selection of FEC on the
CMR, as this parameter most probably was set based on non-IP
information, e.g., radio link performance measures. The sender is
also responsible for avoiding congestion, which may be exacerbated by
redundancy (see Section 6 for more details).

3.7.2. Use of Frame Interleaving

To decrease protocol overhead, the payload design allows several
speech frame-blocks to be encapsulated into a single RTP packet. One
of the drawbacks of such an approach is that packet loss can cause
loss of several consecutive speech frame-blocks, which usually causes
clearly audible distortion in the reconstructed speech. Interleaving
of frame-blocks can improve the speech quality in such cases by
distributing the consecutive losses into a series of single frame-
block losses. However, interleaving and bundling several frame-
blocks per payload will also increase end-to-end delay and is
therefore not appropriate for all types of applications. Streaming
applications will most likely be able to exploit interleaving to
improve speech quality in lossy transmission conditions.

This payload design supports the use of frame interleaving as an
option. For the encoder (speech sender) to use frame interleaving in
its outbound RTP packets for a given session, the decoder (speech
receiver) needs to indicate its support via out-of-band means (see
Section 8).

3.8. Bandwidth-Efficient or Octet-Aligned Mode

For a given session, the payload format can be either bandwidth
efficient or octet aligned, depending on the mode of operation that
is established for the session via out-of-band means.

In the octet-aligned format, all the fields in a payload, including
payload header, table of contents entries, and speech frames
themselves, are individually aligned to octet boundaries to make
implementations efficient. In the bandwidth-efficient format, only
the full payload is octet aligned, so fewer padding bits are added.

   Note, octet alignment of a field or payload means that the last
   octet is padded with zeroes in the least significant bits to fill
   the octet. Also note that this padding is separate from padding
   indicated by the P bit in the RTP header.

Between the two operation modes, only the octet-aligned mode has the
capability to use the robust sorting, interleaving, and frame CRC to
make the speech transport more robust to packet loss and bit errors.

3.9. AMR or AMR-WB Speech over IP Scenarios

   The primary scenario for this payload format is IP end-to-end between
   two terminals, as shown in Figure 2. This payload format is expected
   to be useful for both conversational and streaming services.

```
          +----------+                        +----------+
          |          |   IP/UDP/RTP/AMR or    |          |
          | TERMINAL |<---------------------->| TERMINAL |
          |          |     IP/UDP/RTP/AMR-WB  |          |
          +----------+                        +----------+
```
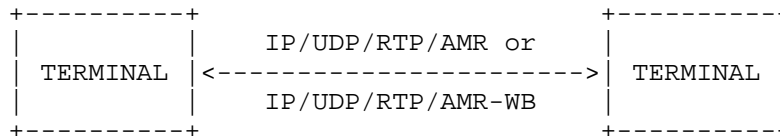
              Figure 2: IP terminal to IP terminal scenario

   A conversational service puts requirements on the payload format.
   Low delay is one very important factor, i.e., few speech frame-blocks
   per payload packet. Low overhead is also required when the payload
   format traverses low bandwidth links, especially as the frequency of
   packets will be high. For low bandwidth links, it is also an
   advantage to support UED, which allows a link provider to reduce
   delay and packet loss, or to reduce the utilization of link
   resources.

   A streaming service has less strict real-time requirements and
   therefore can use a larger number of frame-blocks per packet than a
   conversational service. This reduces the overhead from IP, UDP, and
   RTP headers. However, including several frame-blocks per packet
   makes the transmission more vulnerable to packet loss, so
   interleaving may be used to reduce the effect that packet loss will
   have on speech quality. A streaming server handling a large number
   of clients also needs a payload format that requires as few resources
   as possible when doing packetization. The octet-aligned and
   interleaving modes require the least amount of resources, while CRC,
   robust sorting, and bandwidth-efficient modes have higher demands.

   Another scenario is when AMR or AMR-WB encoded speech is transmitted
   from a non-IP system (e.g., a GSM or 3GPP UMTS network) to an
   IP/UDP/RTP VoIP terminal, and/or vice versa, as depicted in Figure 3.

```
       AMR or AMR-WB
       over
       I.366.{2,3} or +------+                            +----------+
       3G Iu or       |      |  IP/UDP/RTP/AMR or         |          |
       <------------->|  GW  |<-------------------------->| TERMINAL |
       GSM Abis       |      |  IP/UDP/RTP/AMR-WB         |          |
       etc.           +------+                            +----------+
                          |
        GSM/              |              IP network
        3GPP UMTS network |
```
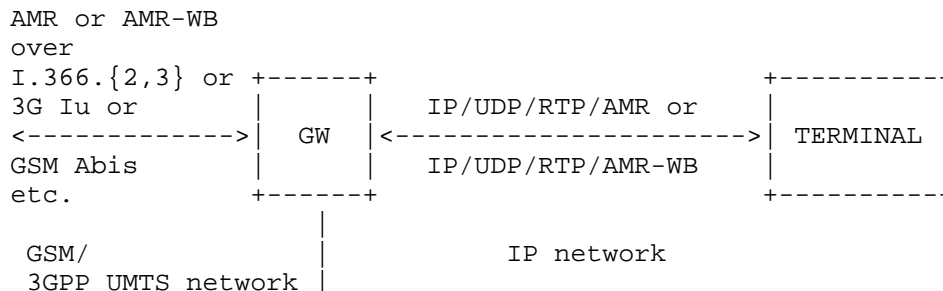
                 Figure 3: GW to VoIP terminal scenario

In such a case, it is likely that the AMR or AMR-WB frame is
packetized in a different way in the non-IP network and will need to
be re-packetized into RTP at the gateway. Also, speech frames from
the non-IP network may come with some UEP/UED information (e.g., a
frame quality indicator) that will need to be preserved and forwarded
on to the decoder along with the speech bits. This is specified in
Section 4.3.2.

AMR's capability to do fast mode switching is exploited in some non-
IP networks to optimize speech quality. To preserve this
functionality in scenarios including a gateway to an IP network, a
codec mode request (CMR) field is needed. The gateway will be
responsible for forwarding the CMR between the non-IP and IP parts in
both directions. The IP terminal should follow the CMR forwarded by
the gateway to optimize speech quality going to the non-IP decoder.
The mode control algorithm in the gateway must accommodate the delay
imposed by the IP network on the IP terminal's response to CMR.

The IP terminal should not set the CMR (see Section 4.3.1), but the
gateway can set the CMR value on frames going toward the encoder in
the non-IP part to optimize speech quality from that encoder to the
gateway. The gateway can alternatively set a lower CMR value, if
desired, as one means to control congestion on the IP network.

A third likely scenario is that IP/UDP/RTP is used as transport
between two non-IP systems, i.e., IP is originated and terminated in
gateways on both sides of the IP transport, as illustrated in Figure
4 below.

```
  AMR or AMR-WB                                        AMR or AMR-WB
  over                                                 over
  I.366.{2,3} or +------+                    +------+ I.366.{2,3} or
  3G Iu or       |      | IP/UDP/RTP/AMR or  |      | 3G Iu or
  <------------->|  GW  |<------------------>|  GW  |<------------->
  GSM Abis       |      | IP/UDP/RTP/AMR-WB  |      | GSM Abis
  etc.           +------+                    +------+ etc.
                    |                           |
   GSM/             |         IP network        |  GSM/
   3GPP UMTS network |                          |  3GPP UMTS network
```
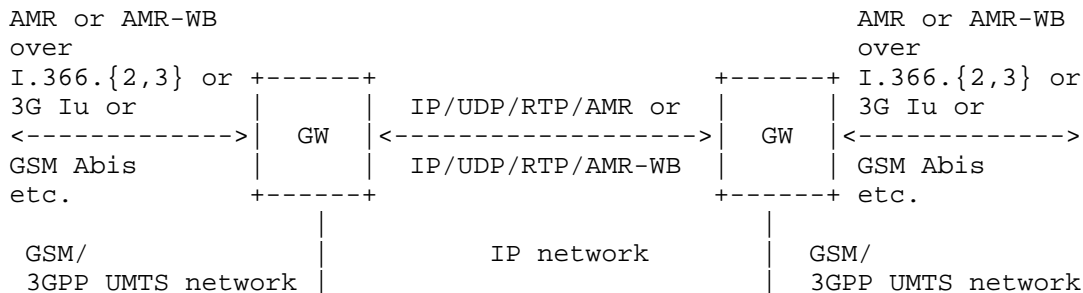
                    Figure 4: GW to GW scenario

   This scenario requires the same mechanisms for preserving UED/UEP and
   CMR information as in the single gateway scenario. In addition, the
   CMR value may be set in packets received by the gateways on the IP
   network side. The gateway should forward to the non-IP side a CMR
   value that is the minimum of three values:

      - the CMR value it receives on the IP side;

      - the CMR value it calculates based on its reception quality on
         the non-IP side; and

      - a CMR value it may choose for congestion control of
         transmission on the IP side.

   The details of the control algorithm are left to the implementation.

4. AMR and AMR-WB RTP Payload Formats

   The AMR and AMR-WB payload formats have identical structure, so they
   are specified together. The only differences are in the types of
   codec frames contained in the payload. The payload format consists
   of the RTP header, payload header, and payload data.

4.1. RTP Header Usage

   The format of the RTP header is specified in [8]. 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-block in the packet. The
   timestamp clock frequency is the same as the sampling frequency, so
   the timestamp unit is in samples.

The duration of one speech frame-block is 20 ms for both AMR and
AMR-WB. For AMR, the sampling frequency is 8 kHz, corresponding to
160 encoded speech samples per frame from each channel. For AMR-WB,
the sampling frequency is 16 kHz, corresponding to 320 samples per
frame from each channel. Thus, the timestamp is increased by 160 for
AMR and 320 for AMR-WB for each consecutive frame-block.

A packet may contain multiple frame-blocks of encoded speech or
comfort noise parameters. If interleaving is employed, the frame-
blocks encapsulated into a payload are picked according to the
interleaving rules as defined in Section 4.4.1. Otherwise, each
packet covers a period of one or more contiguous 20 ms frame-block
intervals. In case the data from all the channels for a particular
frame-block in the period is missing (for example, at a gateway from
some other transport format), it is possible to indicate that no data
is present for that frame-block rather than breaking a multi-frame-
block packet into two, as explained in Section 4.3.2.

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 speech frame multiple times, in exact
duplicates, in different AMR rate modes, or with data present in one
packet and not present in another. If multiple versions of the same
speech frame are received, it is RECOMMENDED that the mode with the
highest rate be used by the speech decoder. A given frame MUST NOT
be encoded as speech in one packet and comfort noise parameters in
another.

The payload length is always made an integral number of octets by
padding with zero bits if necessary. If additional padding is
required to bring the payload length to a larger multiple of octets
or for some other purpose, then the P bit in the RTP in the header
may be set and padding appended as specified in [8].

The RTP header marker bit (M) SHALL be set to 1 if the first frame-
block carried in the packet contains a speech frame which is the
first in a talkspurt. For all other packets the marker bit SHALL be
set to zero (M=0).

The assignment of an RTP payload type for this new packet format is
outside the scope of this document, and will not be specified here.
It is expected that the RTP profile under which this payload format
is being used will assign a payload type for this encoding or specify
that the payload type is to be bound dynamically.

4.2. Payload Structure

   The complete payload consists of a payload header, a payload table of
   contents, and speech data representing one or more speech frame-
   blocks. The following diagram shows the general payload format
   layout:

```
+----------------+-------------------+----------------
| payload header | table of contents | speech data ...
+----------------+-------------------+----------------
```
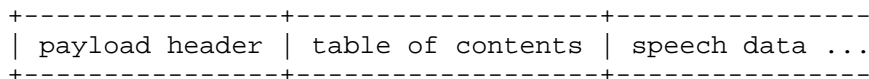
   Payloads containing more than one speech frame-block are called
   compound payloads.

   The following sections describe the variations taken by the payload
   format depending on whether the AMR session is set up to use the
   bandwidth-efficient mode or octet-aligned mode and any of the
   OPTIONAL functions for robust sorting, interleaving, and frame CRCs.
   Implementations SHOULD support both bandwidth-efficient and octet-
   aligned operation to increase interoperability.

4.3. Bandwidth-Efficient Mode

4.3.1. The Payload Header

   In bandwidth-efficient mode, the payload header simply consists of a
   4-bit codec mode request:

```
 0 1 2 3
+-+-+-+-+
|  CMR  |
+-+-+-+-+
```
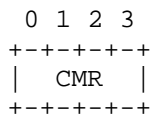
   CMR (4 bits): Indicates a codec mode request sent to the speech
      encoder at the site of the receiver of this payload. The value of
      the CMR field is set to the frame type index of the corresponding
      speech mode being requested. The frame type index may be 0-7 for
      AMR, as defined in Table 1a in [2], or 0-8 for AMR-WB, as defined
      in Table 1a in [4]. CMR value 15 indicates that no mode request
      is present, and other values are for future use.

   The codec mode request received in the CMR field is valid until the
   next codec mode request is received, i.e., a newly received CMR value
   corresponding to a speech mode, or NO_DATA overrides the previously
   received CMR value corresponding to a speech mode or NO_DATA.
   Therefore, if a terminal continuously wishes to receive frames in the

same mode X, it needs to set CMR=X for all its outbound payloads, and
if a terminal has no preference in which mode to receive, it SHOULD
set CMR=15 in all its outbound payloads.

If receiving a payload with a CMR value that is not a speech mode or
NO_DATA, the CMR MUST be ignored by the receiver.

In a multi-channel session, the codec mode request SHOULD be
interpreted by the receiver of the payload as the desired encoding
mode for all the channels in the session.

An IP end-point SHOULD NOT set the codec mode request based on packet
losses or other congestion indications, for several reasons:

   - The other end of the IP path may be a gateway to a non-IP
      network (such as a radio link) that needs to set the CMR field
      to optimize performance on that network.

   - Congestion on the IP network is managed by the IP sender, in
      this case, at the other end of the IP path. Feedback about
      congestion SHOULD be provided to that IP sender through RTCP or
      other means, and then the sender can choose to avoid congestion
      using the most appropriate mechanism. That may include
      adjusting the codec mode, but also includes adjusting the level
      of redundancy or number of frames per packet.

The encoder SHOULD follow a received codec mode request, but MAY
change to a lower-numbered mode if it so chooses, for example, to
control congestion.

The CMR field MUST be set to 15 for packets sent to a multicast
group. The encoder in the speech sender SHOULD ignore codec mode
requests when sending speech to a multicast session but MAY use RTCP
feedback information as a hint that a codec mode change is needed.

The codec mode selection MAY be restricted by a session parameter to
a subset of the available modes. If so, the requested mode MUST be
among the signalled subset (see Section 8). If the received CMR
value is outside the signalled subset of modes, it MUST be ignored.

4.3.2. The Payload Table of Contents

The table of contents (ToC) consists of a list of ToC entries, each
representing a speech frame.

In bandwidth-efficient mode, a ToC entry takes the following format:

```
 0 1 2 3 4 5
+-+-+-+-+-+-+
|F|  FT   |Q|
+-+-+-+-+-+-+
```

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

FT (4 bits): Frame type index, indicating either the AMR or AMR-WB
   speech coding mode or comfort noise (SID) mode of the
   corresponding frame carried in this payload.

The value of FT is defined in Table 1a in [2] for AMR and in Table 1a
in [4] for AMR-WB. FT=14 (SPEECH_LOST, only available for AMR-WB)
and FT=15 (NO_DATA) are used to indicate frames that are either lost
or not being transmitted in this payload, respectively.

NO_DATA (FT=15) frame could mean either that no data for that frame
has been produced by the speech encoder or that no data for that
frame is transmitted in the current payload (i.e., valid data for
that frame could be sent in either an earlier or later packet).

If receiving a ToC entry with a FT value in the range 9-14 for AMR or
10-13 for AMR-WB, the whole packet SHOULD be discarded. This is to
avoid the loss of data synchronization in the depacketization
process, which can result in a huge degradation in speech quality.

Note that packets containing only NO_DATA frames SHOULD NOT be
transmitted in any payload format configuration, except in the case
of interleaving. Also, frame-blocks containing only NO_DATA frames
at the end of a packet SHOULD NOT be transmitted in any payload
format configuration, except in the case of interleaving. The AMR
SCR/DTX is described in [6] and AMR-WB SCR/DTX in [7].

The extra comfort noise frame types specified in table 1a in [2]
(i.e., GSM-EFR CN, IS-641 CN, and PDC-EFR CN) MUST NOT be used in
this payload format because the standardized AMR codec is only
required to implement the general AMR SID frame type and not those
that are native to the incorporated encodings.

Q (1 bit): Frame quality indicator. If set to 0, indicates the
   corresponding frame is severely damaged, and the receiver should
   set the RX_TYPE (see [6]) to either SPEECH_BAD or SID_BAD
   depending on the frame type (FT).

The frame quality indicator is included for interoperability with the
ATM payload format described in ITU-T I.366.2, the UMTS Iu interface
[20], as well as other transport formats. The frame quality
indicator enables damaged frames to be forwarded to the speech
decoder for error concealment. This can improve the speech quality
more than dropping the damaged frames. See Section 4.4.2.1 for more
details.

For multi-channel sessions, the ToC entries of all frames from a
frame-block are placed in the ToC in consecutive order as defined in
Section 4.1 in [12]. When multiple frame-blocks are present in a
packet in bandwidth-efficient mode, they will be placed in the packet
in order of their creation time.

Therefore, with N channels and K speech frame-blocks in a packet,
there MUST be N*K entries in the ToC, and the first N entries will be
from the first frame-block, the second N entries will be from the
second frame-block, and so on.

The following figure shows an example of a ToC of three entries in a
single-channel session using bandwidth-efficient mode.

```
 0                   1
 0 1 2 3 4 5 6 7 8 9 0 1 2 3 4 5 6 7
+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
|1|  FT   |Q|1|  FT   |Q|0|  FT   |Q|
+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
```
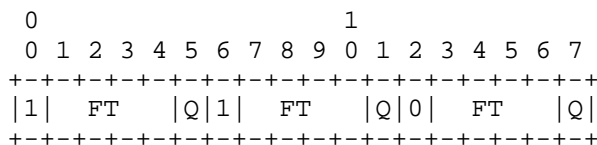
Below is an example of how the ToC entries will appear in the ToC of
a packet carrying three consecutive frame-blocks in a session with
two channels (L and R).

```
+----+----+----+----+----+----+
| 1L | 1R | 2L | 2R | 3L | 3R |
+----+----+----+----+----+----+
|<------->|<------->|<------->|
  Frame-    Frame-    Frame-
  Block 1   Block 2   Block 3
```
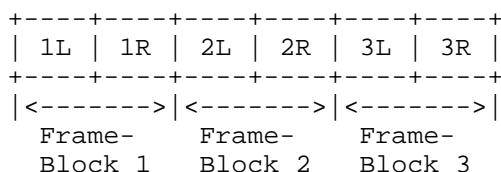
4.3.3. Speech Data

Speech data of a payload contains zero or more speech frames or
comfort noise frames, as described in the ToC of the payload.

   Note, for ToC entries with FT=14 or 15, there will be no
   corresponding speech frame present in the speech data.

Each speech frame represents 20 ms of speech encoded with the mode
indicated in the FT field of the corresponding ToC entry. The length
of the speech frame is implicitly defined by the mode indicated in
the FT field. The order and numbering notation of the bits are as
specified for Interface Format 1 (IF1) in [2] for AMR and [4] for
AMR-WB. As specified there, the bits of speech frames have been
rearranged in order of decreasing sensitivity, while the bits of
comfort noise frames are in the order produced by the encoder. The
resulting bit sequence for a frame of length K bits is denoted d(0),
d(1), ..., d(K-1).

4.3.4. Algorithm for Forming the Payload

   The complete RTP payload in bandwidth-efficient mode is formed by
   packing bits from the payload header, table of contents, and speech
   frames in order (as defined by their corresponding ToC entries in the
   ToC list), and to bring the payload to octet alignment, 0 to 7
   padding bits. Padding bits MUST be set to zero and MUST be ignored
   on reception. They are packed contiguously into octets beginning
   with the most significant bits of the fields and the octets.

   To be precise, the four-bit payload header is packed into the first
   octet of the payload with bit 0 of the payload header in the most
   significant bit of the octet. The four most significant bits
   (numbered 0-3) of the first ToC entry are packed into the least
   significant bits of the octet, ending with bit 3 in the least
   significant bit. Packing continues in the second octet with bit 4 of
   the first ToC entry in the most significant bit of the octet. If
   more than one frame is contained in the payload, then packing
   continues with the second and successive ToC entries. Bit 0 of the
   first data frame follows immediately after the last ToC bit,
   proceeding through all the bits of the frame in numerical order.
   Bits from any successive frames follow contiguously in numerical
   order for each frame and in consecutive order of the frames.

   If speech data is missing for one or more speech frame within the
   sequence, because of, for example, DTX, a ToC entry with FT set to
   NO_DATA SHALL be included in the ToC for each of the missing frames,
   but no data bits are included in the payload for the missing frame
   (see Section 4.3.5.2 for an example).

4.3.5. Payload Examples

4.3.5.1. Single-Channel Payload Carrying a Single Frame

   The following diagram shows a bandwidth-efficient AMR payload from a
   single-channel session carrying a single speech frame-block.

In the payload, no specific mode is requested (CMR=15), the speech
frame is not damaged at the IP origin (Q=1), and the coding mode is
AMR 7.4 kbps (FT=4). The encoded speech bits, d(0) to d(147), are
arranged in descending sensitivity order according to [2]. Finally,
two padding bits (P) are added to the end as padding to make the
payload octet aligned.

```
 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
+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
| CMR=15|0| FT=4  |1|d(0)                                       |
+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
|                                                               |
+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
|                                                               |
+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
|                                                               |
+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
|                                                      d(147)|P|P|
+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
```

4.3.5.2. Single-Channel Payload Carrying Multiple Frames

The following diagram shows a single-channel, bandwidth-efficient
compound AMR-WB payload that contains four frames, of which one has
no speech data. The first frame is a speech frame at 6.6 kbps mode
(FT=0) that is composed of speech bits d(0) to d(131). The second
frame is an AMR-WB SID frame (FT=9), consisting of bits g(0) to
g(39). The third frame is a NO_DATA frame and does not carry any
speech information, it is represented in the payload by its ToC
entry. The fourth frame in the payload is a speech frame at 8.85
kbps mode (FT=1), it consists of speech bits h(0) to h(176).

As shown below, the payload carries a mode request for the encoder on
the receiver's side to change its future coding mode to AMR-WB 8.85
kbps (CMR=1). None of the frames are damaged at IP origin (Q=1).
The encoded speech and SID bits, d(0) to d(131), g(0) to g(39), and
h(0) to h(176), are arranged in the payload in descending sensitivity
order according to [4]. (Note, no speech bits are present for the
third frame.) Finally, seven zero bits are padded to the end to
make the payload octet aligned.

```
 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
+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
| CMR=1 |1| FT=0  |1|1| FT=9  |1|1| FT=15 |1|0| FT=1  |1|d(0)   |
+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
|                                                               |
+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
|                                                               |
+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
|                                                               |
+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
|                                                        d(131)|
+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
|g(0)                                                           |
+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
|         g(39)|h(0)                                            |
+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
|                                                               |
+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
|                                                               |
+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
|                                                               |
+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
|                                        h(176)|P|P|P|P|P|P|P|
+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
```

4.3.5.3. Multi-Channel Payload Carrying Multiple Frames

   The following diagram shows a two-channel payload carrying 3 frame-
   blocks, i.e., the payload will contain 6 speech frames.

   In the payload, all speech frames contain the same mode 7.4 kbps
   (FT=4) and are not damaged at IP origin. The CMR is set to 15, i.e.,
   no specific mode is requested. The two channels are defined as left
   (L) and right (R) in that order. The encoded speech bits is
   designated dXY(0).. dXY(K-1), where X = block number, Y = channel,
   and K is the number of speech bits for that mode. Exemplifying this,
   for frame-block 1 of the left channel, the encoded bits are
   designated as d1L(0) to d1L(147).

```
  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
 +-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
 | CMR=15|1|1L FT=4|1|1|1R FT=4|1|1|2L FT=4|1|1|2R FT=4|1|1|3L FT|
 +-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
 |4|1|0|3R FT=4|1|d1L(0)                                         |
 +-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
 |                                                               |
 +-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
 |                                                               |
 +-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
 |                                                               |
 +-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
 |                                                  d1L(147)|d1R(0) |
 +-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
 : ...                                                           :
 +-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
 |                      d1R(147)|d2L(0)                          |
 +-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
 : ...                                                           :
 +-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
 |d2L(147|d2R(0)                                                 |
 +-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
 : ...                                                           :
 +-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
 |                                           d2R(147)|d3L(0)     |
 +-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
 : ...                                                           :
 +-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
 |              d3L(147)|d3R(0)                                  |
 +-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
 : ...                                                           :
 +-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
 |                                                      d3R(147)|
 +-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
```

4.4. Octet-Aligned Mode

4.4.1. The Payload Header

   In octet-aligned mode, the payload header consists of a 4-bit CMR, 4
   reserved bits, and optionally, an 8-bit interleaving header, as shown
   below:

```
    0                   1
    0 1 2 3 4 5 6 7 8 9 0 1 2 3 4 5
   +-+-+-+-+-+-+-+-+- - - - - - - -
   |  CMR  |R|R|R|R|  ILL  |  ILP  |
   +-+-+-+-+-+-+-+-+- - - - - - - -
```

   CMR (4 bits): same as defined in Section 4.3.1.

   R: is a reserved bit that MUST be set to zero. All R bits MUST be
      ignored by the receiver.

   ILL (4 bits, unsigned integer): This is an OPTIONAL field that is
      present only if interleaving is signalled out-of-band for the
      session. ILL=L indicates to the receiver that the interleaving
      length is L+1, in number of frame-blocks.

   ILP (4 bits, unsigned integer): This is an OPTIONAL field that is
      present only if interleaving is signalled. ILP MUST take a value
      between 0 and ILL, inclusive, indicating the interleaving index
      for frame-blocks in this payload in the interleaving group. If
      the value of ILP is found greater than ILL, the payload SHOULD be
      discarded.

   ILL and ILP fields MUST be present in each packet in a session if
   interleaving is signalled for the session. Interleaving MUST be
   performed on a frame-block basis (i.e., NOT on a frame basis) in a
   multi-channel session.

   The following example illustrates the arrangement of speech frame-
   blocks in an interleaving group during an interleaving session. Here
   we assume ILL=L for the interleaving group that starts at speech
   frame-block n. We also assume that the first payload packet of the
   interleaving group is s, and the number of speech frame-blocks
   carried in each payload is N. Then we will have:

   Payload s (the first packet of this interleaving group):
      ILL=L, ILP=0,
      Carry frame-blocks: n, n+(L+1), n+2*(L+1), ..., n+(N-1)*(L+1)

   Payload s+1 (the second packet of this interleaving group):
      ILL=L, ILP=1,
      frame-blocks: n+1, n+1+(L+1), n+1+2*(L+1), ..., n+1+(N-1)*(L+1)
      ...

   Payload s+L (the last packet of this interleaving group):
      ILL=L, ILP=L,
      frame-blocks: n+L, n+L+(L+1), n+L+2*(L+1), ..., n+L+(N-1)*(L+1)

   The next interleaving group will start at frame-block n+N*(L+1).

   There will be no interleaving effect unless the number of frame-
   blocks per packet (N) is at least 2. Moreover, the number of frame-
   blocks per payload (N) and the value of ILL MUST NOT be changed
   inside an interleaving group. In other words, all payloads in an
   interleaving group MUST have the same ILL and MUST contain the same
   number of speech frame-blocks.

   The sender of the payload MUST only apply interleaving if the
   receiver has signalled its use through out-of-band means. Since
   interleaving will increase buffering requirements at the receiver,
   the receiver uses media type parameter "interleaving=I" to set the
   maximum number of frame-blocks allowed in an interleaving group to I.

   When performing interleaving, the sender MUST use a proper number of
   frame-blocks per payload (N) and ILL so that the resulting size of an
   interleaving group is less or equal to I, that is, N*(L+1)<=I.

4.4.2. The Payload Table of Contents and Frame CRCs

   The table of contents (ToC) in octet-aligned mode consists of a list
   of ToC entries where each entry corresponds to a speech frame carried
   in the payload and, optionally, a list of speech frame CRCs. That
   is, the ToC is as follows:

   +---------------------+
   | list of ToC entries |
   +---------------------+
   | list of frame CRCs | (optional)
    - - - - - - - - - - -

      Note, for ToC entries with FT=14 or 15, there will be no
      corresponding speech frame or frame CRC present in the payload.

The list of ToC entries is organized in the same way as described for
bandwidth-efficient mode in 4.3.2, with the following exception:
when interleaving is used, the frame-blocks in the ToC will almost
never be placed consecutively in time. Instead, the presence and
order of the frame-blocks in a packet will follow the pattern
described in 4.4.1.

The following example shows the ToC of three consecutive packets,
each carrying three frame-blocks, in an interleaved two-channel
session. Here, the two channels are left (L) and right (R) with L
coming before R, and the interleaving length is 3 (i.e., ILL=2).
This results in the interleaving group size of 9 frame-blocks.

```
Packet #1
---------

ILL=2, ILP=0:
+----+----+----+----+----+----+
| 1L | 1R | 4L | 4R | 7L | 7R |
+----+----+----+----+----+----+
|<------->|<------->|<------->|
  Frame-    Frame-    Frame-
  Block 1   Block 4   Block 7

Packet #2
---------

ILL=2, ILP=1:
+----+----+----+----+----+----+
| 2L | 2R | 5L | 5R | 8L | 8R |
+----+----+----+----+----+----+
|<------->|<------->|<------->|
  Frame-    Frame-    Frame-
  Block 2   Block 5   Block 8

Packet #3
---------

ILL=2, ILP=2:
+----+----+----+----+----+----+
| 3L | 3R | 6L | 6R | 9L | 9R |
+----+----+----+----+----+----+
|<------->|<------->|<------->|
  Frame-    Frame-    Frame-
  Block 3   Block 6   Block 9
```

A ToC entry takes the following format in octet-aligned mode:

```
 0 1 2 3 4 5 6 7
+-+-+-+-+-+-+-+-+
|F|  FT   |Q|P|P|
+-+-+-+-+-+-+-+-+
```

F (1 bit): see definition in Section 4.3.2.

FT (4 bits, unsigned integer): see definition in Section 4.3.2.

Q (1 bit): see definition in Section 4.3.2.

P bits: padding bits, MUST be set to zero, and MUST be ignored on
        reception.

The list of CRCs is OPTIONAL. It only exists if the use of CRC is
signalled out-of-band for the session. When present, each CRC in the
list is 8 bits long and corresponds to a speech frame (NOT a frame-
block) carried in the payload. Calculation and use of the CRC is
specified in the next section.

4.4.2.1. Use of Frame CRC for UED over IP

The general concept of UED/UEP over IP is discussed in Section 3.6.
This section provides more details on how to use the frame CRC in the
octet-aligned payload header together with a partial transport layer
checksum to achieve UED.

To achieve UED, one SHOULD use a transport layer checksum (for
example, the one defined in UDP-Lite [19]) to protect the IP,
transport protocol (e.g., UDP-Lite), and RTP headers, as well as the
payload header and the table of contents in the payload. The frame
CRC, when used, MUST be calculated only over all class A bits in the
AMR or AMR-WB frame. Class B and C bits in the AMR or AMR-WB frame
MUST NOT be included in the CRC calculation and SHOULD NOT be covered
by the transport checksum.

   Note, the number of class A bits for various coding modes in AMR
   codec is specified as informative in [2] and is therefore copied
   into Table 1 in Section 3.6 to make it normative for this payload
   format. The number of class A bits for various coding modes in
   AMR-WB codec is specified as normative in Table 2 in [4], and the
   SID frame (FT=9) has 40 class A bits. These definitions of class
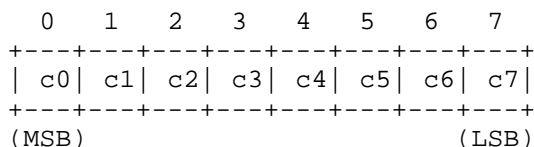   A bits MUST be used for this payload format.

If the transport layer checksum or link layer checksum detects any
errors within the protected (sensitive) part, it is assumed that the
complete packet will be discarded as defined by UDP-Lite [19].

The receiver of the payload SHOULD examine the data integrity of the
received class A bits by re-calculating the CRC over the received
class A bits and comparing the result to the value found in the
received payload header. If the two values mismatch, the receiver
SHALL consider the class A bits in the receiver frame damaged and
MUST clear the Q flag of the frame (i.e., set it to 0). This will
subsequently cause the frame to be marked as SPEECH_BAD, if the FT of
the frame is 0..7 for AMR or 0..8 for AMR-WB, or SID_BAD if the FT of
the frame is 8 for AMR or 9 for AMR-WB, before it is passed to the
speech decoder. See [6] and [7] more details.

The following example shows an octet-aligned ToC with a CRC list for
a payload containing 3 speech frames from a single-channel session
(assuming none of the FTs is equal to 14 or 15):

```
 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|  FT#1 |Q|P|P|1|  FT#2 |Q|P|P|0|  FT#3 |Q|P|P|     CRC#1     |
+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
|     CRC#2     |     CRC#3     |
+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
```
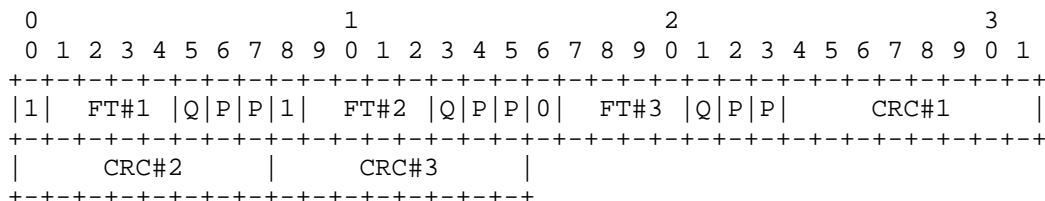
Each of the CRCs takes 8 bits

```
  0   1   2   3   4   5   6   7
+---+---+---+---+---+---+---+---+
| c0| c1| c2| c3| c4| c5| c6| c7|
+---+---+---+---+---+---+---+---+
(MSB)                     (LSB)
```

and is calculated by the cyclic generator polynomial,

   $C(x) = 1 + x^2 + x^3 + x^4 + x^8$

where ^ is the exponentiation operator.

In binary form, the polynomial appears as follows: 101110001
(MSB..LSB).

The actual calculation of the CRC is made as follows: First, an
8-bit CRC register is reset to zero: 00000000. For each bit over
which the CRC shall be calculated, an XOR operation is made between
the rightmost (LSB) bit of the CRC register and the bit. The CRC

register is then right-shifted one step (each bit's significance is
reduced by one), inputting a "0" as the leftmost bit (MSB). If the
result of the XOR operation mentioned above is a "1", then "10111000"
is bit-wise XOR-ed into the CRC register. This operation is repeated
for each bit that the CRC should cover. In this case, the first bit
would be d(0) for the speech frame for which the CRC should cover.
When the last bit (e.g., d(54) for AMR 5.9 according to Table 1 in
Section 3.6) has been used in this CRC calculation, the contents in
CRC register should simply be copied to the corresponding field in
the list of CRCs.

Fast calculation of the CRC on a general-purpose CPU is possible
using a table-driven algorithm.

4.4.3. Speech Data

In octet-aligned mode, speech data is carried in a similar way to
that in the bandwidth-efficient mode as discussed in Section 4.3.3,
with the following exceptions:

   - The last octet of each speech frame MUST be padded with zero
      bits at the end if all bits in the octet are not used. The
      padding bits MUST be ignored on reception. In other words,
      each speech frame MUST be octet-aligned.

   - When multiple speech frames are present in the speech data
      (i.e., compound payload), the speech frames are arranged either
      one whole frame after another as usual, or with the octets of
      all frames interleaved together at the octet level, depending
      on the media type parameters negotiated for the payload type.
      Since the bits within each frame are ordered with the most
      error-sensitive bits first, interleaving the octets collects
      those sensitive bits from all frames to be nearer the beginning
      of the packet. This is called "robust sorting order" which
      allows the application of UED (such as UDP-Lite [19]) or UEP
      (such as the ULP [22]) mechanisms to the payload data. The
      details of assembling the payload are given in the next
      section.

The use of robust sorting order for a payload type MUST be agreed via
out-of-band means. Section 8 specifies a media type parameter for
this purpose.

Note, robust sorting order MUST only be performed on the frame level
and thus is independent of interleaving, which is at the frame-block
level, as described in Section 4.4.1. In other words, robust sorting
can be applied to either non-interleaved or interleaved payload
types.

4.4.4. Methods for Forming the Payload

   Two different packetization methods, namely, normal order and robust
   sorting order, exist for forming a payload in octet-aligned mode. In
   both cases, the payload header and table of contents are packed into
   the payload the same way; the difference is in the packing of the
   speech frames.

   The payload begins with the payload header of one octet, or two
   octets if frame interleaving is selected. The payload header is
   followed by the table of contents consisting of a list of one-octet
   ToC entries. If frame CRCs are to be included, they follow the table
   of contents with one 8-bit CRC filling each octet. Note that if a
   given frame has a ToC entry with FT=14 or 15, there will be no CRC
   present.

   The speech data follows the table of contents, or the CRCs if
   present. For packetization in the normal order, all of the octets
   comprising a speech frame are appended to the payload as a unit. The
   speech frames are packed in the same order as their corresponding ToC
   entries are arranged in the ToC list, with the exception that if a
   given frame has a ToC entry with FT=14 or 15, there will be no data
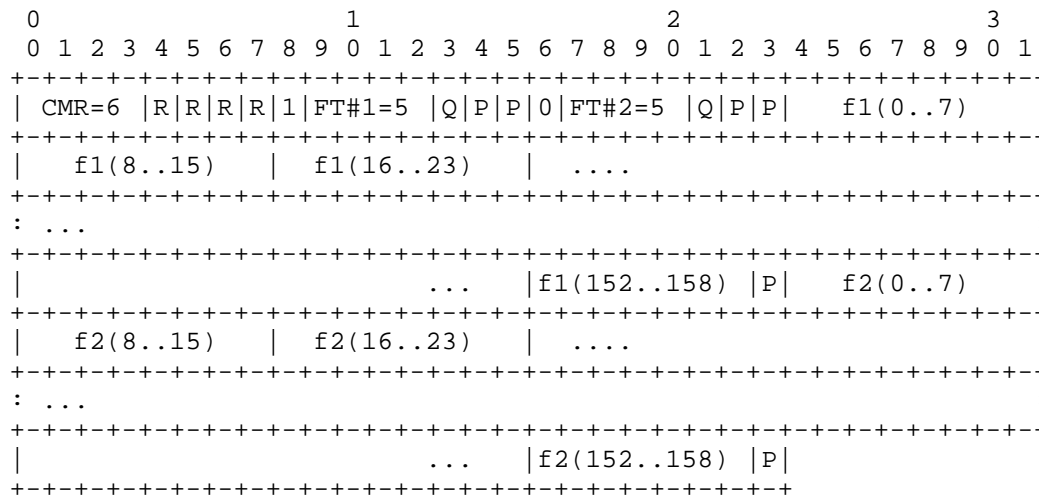   octets present for that frame.

   For packetization in robust sorting order, the octets of all speech
   frames are interleaved together at the octet level. That is, the
   data portion of the payload begins with the first octet of the first
   frame, followed by the first octet of the second frame, then the
   first octet of the third frame, and so on. After the first octet of
   the last frame has been appended, the cycle repeats with the second
   octet of each frame. The process continues for as many octets as are
   present in the longest frame. If the frames are not all the same
   octet length, a shorter frame is skipped once all octets in it have
   been appended. The order of the frames in the cycle will be
   sequential if frame interleaving is not in use, or according to the
   interleave pattern specified in the payload header if frame
   interleaving is in use. Note that if a given frame has a ToC entry
   with FT=14 or 15, there will be no data octets present for that
   frame, so it is skipped in the robust sorting cycle.

   The UED and/or UEP is RECOMMENDED to cover at least the RTP header,
   payload header, table of contents, and class A bits of a sorted
   payload. Exactly how many octets need to be covered depends on the
   network and application. If CRCs are used together with robust
   sorting, only the RTP header, the payload header, and the ToC SHOULD
   be covered by UED/UEP. The means for communicating the number of
   octets to be covered to other layers performing UED/UEP is beyond the
   scope of this specification.

4.4.5. Payload Examples

4.4.5.1. Basic Single-Channel Payload Carrying Multiple Frames

   The following diagram shows an octet aligned payload from a single
   channel payload type that carries two AMR frames of 7.95 kbps coding
   mode (FT=5). In the payload, a codec mode request is sent (CMR=6),
   requesting the encoder at the receiver's side to use AMR 10.2 kbps
   coding mode. No frame CRC, interleaving, or robust sorting is in
   use.

```
 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
+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
| CMR=6 |R|R|R|R|1|FT#1=5 |Q|P|P|0|FT#2=5 |Q|P|P|   f1(0..7)    |
+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
|   f1(8..15)   |   f1(16..23)  |   ....                        |
+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
: ...                                                           :
+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
|                           ...   |f1(152..158) |P|   f2(0..7)  |
+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
|   f2(8..15)   |   f2(16..23)  |   ....                        |
+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
: ...                                                           :
+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
|                           ...   |f2(152..158) |P|
+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
```

   Note, in the above example, the last octet in both speech frames is
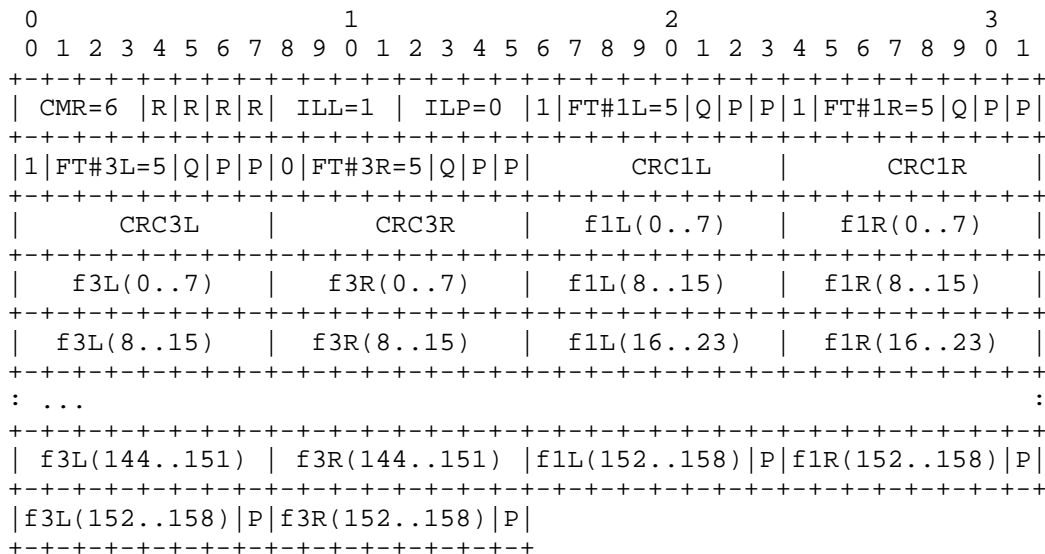   padded with one zero bit to make it octet-aligned.

4.4.5.2. Two-Channel Payload with CRC, Interleaving, and Robust Sorting

   This example shows an octet aligned payload from a two-channel
   payload type. Two frame-blocks, each containing two speech frames of
   7.95 kbps coding mode (FT=5), are carried in this payload.

   The two channels are left (L) and right (R) with L coming before R.
   In the payload, a codec mode request is also sent (CMR=6), requesting
   the encoder at the receiver's side to use AMR 10.2 kbps coding mode.

   Moreover, frame CRC, robust sorting, and frame-block interleaving are
   all enabled for the payload type. The interleaving length is 2
   (ILL=1), and this payload is the first one in an interleaving group
   (ILP=0).

The first two frames in the payload are the L and R channel speech
frames of frame-block #1, consisting of bits f1L(0..158) and
f1R(0..158), respectively. The next two frames are the L and R
channel frames of frame-block #3, consisting of bits f3L(0..158) and
f3R(0..158), respectively, due to interleaving. For each of the four
speech frames, a CRC is calculated as CRC1L(0..7), CRC1R(0..7),
CRC3L(0..7), and CRC3R(0..7), respectively. Finally, the payload is
robust sorted.

```
 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
+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
| CMR=6 |R|R|R|R| ILL=1 | ILP=0 |1|FT#1L=5|Q|P|P|1|FT#1R=5|Q|P|P|
+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
|1|FT#3L=5|Q|P|P|0|FT#3R=5|Q|P|P|     CRC1L      |     CRC1R     |
+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
|     CRC3L      |     CRC3R      |   f1L(0..7)   |   f1R(0..7)   |
+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
|   f3L(0..7)   |   f3R(0..7)    |   f1L(8..15)  |   f1R(8..15)  |
+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
|   f3L(8..15)  |   f3R(8..15)   |   f1L(16..23) |   f1R(16..23) |
+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
: ...                                                           :
+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
| f3L(144..151) | f3R(144..151) |f1L(152..158)|P|f1R(152..158)|P|
+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
|f3L(152..158)|P|f3R(152..158)|P|
+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
```

Note, in the above example, the last octet in all four speech frames
is padded with one zero bit to make it octet-aligned.

4.5. Implementation Considerations

An application implementing this payload format MUST understand all
the payload parameters in the out-of-band signaling used. For
example, if an application uses SDP, all the SDP and media type
parameters in this document MUST be understood. This requirement
ensures that an implementation always can decide if it is capable or
not of communicating.

No operating mode of the payload format is mandatory to implement.
The requirements of the application using the payload format should
be used to determine what to implement. To achieve basic
interoperability, an implementation SHOULD at least implement both
bandwidth-efficient and octet-aligned modes for a single audio

channel. The other operating modes: interleaving, robust sorting,
and frame-wise CRC (in both single and multi-channel) are OPTIONAL to
implement.

The mode-change-period, mode-change-capability, and mode-change-
neighbor parameters are intended for signaling with GSM endpoints.
When interoperability with GSM is desired, encoders SHOULD only
perform codec mode changes to neighboring modes and in integer
multiples of 40 ms (two frame-blocks), but decoders SHOULD accept
codec mode changes at any time, i.e., for every frame-block. The
encoder may arbitrarily select the initial phase (odd or even frame-
block) where codec mode changes are performed, but then SHOULD stick
to that phase as far as possible. However, in rare cases, handovers
or other events (e.g., call forwarding) may change this phase and may
also cause mode changes to non-neighboring modes. The decoder SHALL
therefore be prepared to accept changes also in the other phase and
to other modes. Section 8 specifies the usage of the parameters
mode-change-period and mode-change-capability to indicate the desired
behavior in applications.

See 3GPP TS 26.103 [28] for preferred AMR and AMR-WB configurations
for operation in GSM and 3GPP UMTS networks. In gateway scenarios,
encoders can be requested through the "mode-set" parameter to use a
limited mode-set that is supported by the link beyond the gateway.
Further, to avoid congestion on that link, the encoder SHOULD limit
the initial codec mode for a session to a lower mode, until at least
one frame-block is received with rate control information.

4.5.1. Decoding Validation

When processing a received payload packet, if the receiver finds that
the calculated payload length, based on the information for the
payload type and the values found in the payload header fields, does
not match the size of the received packet, the receiver SHOULD
discard the packet. This is because decoding a packet that has
errors in its length field could severely degrade the speech quality.

5. AMR and AMR-WB Storage Format

    The storage format is used for storing AMR or AMR-WB speech frames in
    a file or as an email attachment. Multiple channel content is
    supported.

    In general, an AMR or AMR-WB file has the following structure:

    +------------------+
    | Header |
    +------------------+
    | Speech frame 1 |
    +------------------+
    : ...   :
    +------------------+
    | Speech frame n |
    +------------------+

    Note, to preserve interoperability with already deployed
    implementations, single-channel content uses a file header format
    different from that of multi-channel content.

    There also exists another storage format for AMR and AMR-WB that is
    suitable for applications with more advanced demands on the storage
    format, like random access or synchronization with video. This
    format is the 3GPP-specified ISO-based multimedia file format 3GP
    [31]. Its media type is specified by RFC 3839 [32].

5.1. Single-Channel Header

    A single-channel AMR or AMR-WB file header contains only a magic
    number. Different magic numbers are defined to distinguish AMR from
    AMR-WB.

    The magic number for single-channel AMR files MUST consist of ASCII
    character string:

        "#!AMR\n"
        (or 0x2321414d520a in hexadecimal).

    The magic number for single-channel AMR-WB files MUST consist of
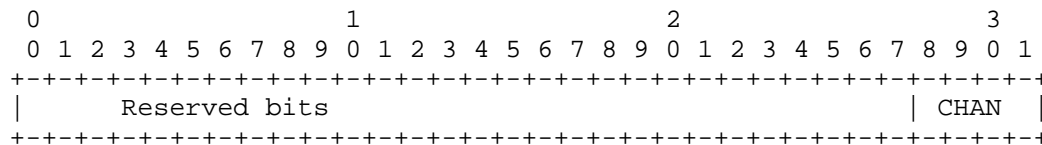   ASCII character string:

        "#!AMR-WB\n"
        (or 0x2321414d522d57420a in hexadecimal).

Note, the "\n" is an important part of the magic numbers and MUST be
included in the comparison, since, otherwise, the single-channel
magic numbers above will become indistinguishable from those of the
multi-channel files defined in the next section.

5.2. Multi-Channel Header

The multi-channel header consists of a magic number followed by a
32-bit channel description field, giving the multi-channel header the
following structure:

```
+-----------------+
| magic number    |
+-----------------+
| chan-desc field |
+-----------------+
```
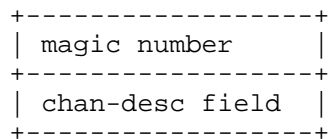
The magic number for multi-channel AMR files MUST consist of the
ASCII character string:

    "#!AMR_MC1.0\n"
    (or 0x2321414d525F4D43312E300a in hexadecimal).

The magic number for multi-channel AMR-WB files MUST consist of the
ASCII character string:

    "#!AMR-WB_MC1.0\n"
    (or 0x2321414d522d57425F4D43312E300a in hexadecimal).

The version number in the magic numbers refers to the version of the
file format.

The 32 bit channel description field is defined as:

```
 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
+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
|       Reserved bits                                   | CHAN  |
+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
```

Reserved bits: MUST be set to 0 when written, and a reader MUST
              ignore them.

CHAN (4 bits, unsigned integer): Indicates the number of audio
channels contained in this storage file. The valid values and the
order of the channels within a frame-block are specified in Section
4.1 in [12].

5.3. Speech Frames

   After the file header, speech frame-blocks consecutive in time are
   stored in the file. Each frame-block contains a number of octet-
   aligned speech frames equal to the number of channels, and stored in
   increasing order, starting with channel 1.

   Each stored speech frame starts with a one-octet frame header with
   the following format:

```
 0 1 2 3 4 5 6 7
+-+-+-+-+-+-+-+-+
|P|  FT   |Q|P|P|
+-+-+-+-+-+-+-+-+
```

   The FT field and the Q bit are defined in the same way as in Section
   4.3.2. The P bits are padding and MUST be set to 0, and MUST be
   ignored.

   Following this one octet header come the speech bits as defined in
   4.4.3. The last octet of each frame is padded with zeroes, if
   needed, to achieve octet alignment.

   The following example shows an AMR frame in 5.9 kbps coding mode
   (with 118 speech bits) in the storage 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
+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
|P| FT=2  |Q|P|P|                                               |
+-+-+-+-+-+-+-+-+                                               +
|                                                               |
+          Speech bits for frame-block n, channel k            +
|                                                               |
+                                                       +-+-+
|                                                       |P|P|
+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
```

   Non-received speech frames or frame-blocks between SID updates during
   non-speech periods MUST be stored as NO_DATA frames (frame type 15,
   as defined in [2] and [4]). Frames or frame-blocks lost in
   transmission MUST be stored as NO_DATA frames or SPEECH_LOST (frame
   type 14, only available for AMR-WB) in complete frame-blocks to keep
   synchronization with the original media.

   Comfort noise frames of other types than AMR SID (FT=8) (i.e., frame
   type 9, 10, and 11 for AMR) SHALL NOT be used in the AMR file format.

6. Congestion Control

   The general congestion control considerations for transporting RTP
   data apply to AMR or AMR-WB speech over RTP as well. However, the
   multi-rate capability of AMR and AMR-WB speech coding may provide an
   advantage over other payload formats for controlling congestion since
   the bandwidth demand can be adjusted by selecting a different coding
   mode.

   Another parameter that may impact the bandwidth demand for AMR and
   AMR-WB is the number of frame-blocks that are encapsulated in each
   RTP payload. Packing more frame-blocks in each RTP payload can
   reduce the number of packets sent and hence the overhead from
   IP/UDP/RTP headers, at the expense of increased delay.

   If forward error correction (FEC) is used to combat packet loss, the
   amount of redundancy added by FEC will need to be regulated so that
   the use of FEC itself does not cause a congestion problem.

   It is RECOMMENDED that AMR or AMR-WB applications using this payload
   format employ congestion control. The actual mechanism for
   congestion control is not specified but should be suitable for real-
   time flows, possibly "TCP Friendly Rate Control" [21].

7. Security Considerations

   RTP packets using the payload format defined in this specification
   are subject to the general security considerations discussed in [8]
   and in any used profile, like AVP [12] or SAVP [26].

   As this format transports encoded speech, the main security issues
   include confidentiality, authentication, and integrity of the speech
   itself. The payload format itself does not have any built-in
   security mechanisms. External mechanisms, such as SRTP [26], need to
   be used for this functionality. Note that the appropriate mechanism
   to provide security to RTP and the payloads following this memo may
   vary. It is dependent on the application, the transport, and the
   signaling protocol employed. Therefore, a single mechanism is not
   sufficient, although if suitable the usage of SRTP [26] is
   RECOMMENDED. Other known mechanisms that may be used are IPsec [33]
   and TLS [34] (RTP over TCP), but other alternatives may also exist.

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

7.1. Confidentiality

   To achieve confidentiality of the encoded AMR or AMR-WB speech, all
   speech data bits will need to be encrypted. There is less of a need
   to encrypt the payload header or the table of contents due to a) that
   they only carry information about the requested speech mode, frame
   type, and frame quality, and b) that this information could be useful
   to some third party, e.g., quality monitoring.

   The packetization and unpacketization of the AMR and AMR-WB payload
   is done only at the endpoints. Therefore encryption should be
   performed after packet encapsulation, and decryption should be
   performed before packet decapsulation.

   Encryption may affect interleaving. Specifically, a change of keys
   should occur at the boundary between interleaving groups. If it is
   not done at that boundary on both endpoints, the speech quality will
   be degraded during the complete interleaving group for any receiver.

   The encryption mechanism may impact the robustness of the error
   correcting mechanism. This is discussed in Section 9.5 of SRTP [26].
   From this, UED/UEP based on robust sorting may be difficult to apply
   when the payload data is encrypted.

7.2. Authentication and Integrity

   To authenticate the sender and to protect the integrity of the RTP
   packets in transit, an external mechanism has to be used. As stated
   before, it is RECOMMENDED that SRTP [26] be used for common
   interoperability. Note that the use of UED/UEP may be difficult to
   combine with some integrity protection mechanisms because any bit
   errors will cause the integrity check to fail.

   Data tampering by a man-in-the-middle attacker could result in
   erroneous depacketization/decoding that could lower the speech
   quality or produce unintelligible communications. Tampering with the
   CMR field may result in a different speech quality than desired.

8. Payload Format Parameters

   This section defines the parameters that may be used to select
   optional features of the AMR and AMR-WB payload formats. The
   parameters are defined here as part of the media type registrations
   for the AMR and AMR-WB speech codecs. The registrations are done
   following RFC 4855 [15] and the media registration rules [14].

A mapping of the parameters into the Session Description Protocol
(SDP) [11] is also provided for those applications that use SDP.
Equivalent parameters could be defined elsewhere for use with control
protocols that do not use media types or SDP.

Two separate media type registrations are made, one for AMR and one
for AMR-WB, because they are distinct encodings that must be
distinguished by their own media type.

Data formats are specified for both real-time transport in RTP and
for storage type applications such as email attachments.

8.1. AMR Media Type Registration

The media type for the Adaptive Multi-Rate (AMR) codec is allocated
from the IETF tree since AMR is a widely used speech codec in general
VoIP and messaging applications. This media type registration covers
both real-time transfer via RTP and non-real-time transfers via
stored files.

Note, any unspecified parameter MUST be ignored by the receiver.

Media Type name: audio

Media subtype name: AMR

Required parameters: none

Optional parameters:

   These parameters apply to RTP transfer only.

   octet-align: Permissible values are 0 and 1. If 1, octet-aligned
           operation SHALL be used. If 0 or if not present,
           bandwidth-efficient operation is employed.

   mode-set: Restricts the active codec mode set to a subset of all
           modes, for example, to be able to support transport
           channels such as GSM networks in gateway use cases.
           Possible values are a comma separated list of modes from
           the set: 0,...,7 (see Table 1a [2]). The SID frame type
           8 and NO_DATA (frame type 15) are never included in the
           mode set, but can always be used. If mode-set is
           specified, it MUST be abided, and frames encoded with
           modes outside of the subset MUST NOT be sent in any RTP
           payload or used in codec mode requests. If not present,
           all codec modes are allowed for the payload type.

mode-change-period: Specifies a number of frame-blocks, N (1 or
          2), that is the frame-block period at which codec mode
          changes are allowed for the sender. The initial phase of
          the interval is arbitrary, but changes must be separated
          by a period of N frame-blocks, i.e., a value of 2
          allows the sender to change mode every second frame-
          block. The value of N SHALL be either 1 or 2. If this
          parameter is not present, mode changes are allowed at
          any time during the session, i.e., N=1.

mode-change-capability: Specifies if the client is capable to
          transmit with a restricted mode change period. The
          parameter may take value of 1 or 2. A value of 1
          indicates that the client is not capable of restricting
          the mode change period to 2, and that the codec mode may
          be changed at any point. A value of 2 indicates that the
          client has the capability to restrict the mode change
          period to 2, and thus that the client can correctly
          interoperate with a receiver requiring a mode-change-
          period=2. If this parameter is not present, the mode-
          change restriction capability is not supported, i.e.
          mode-change-capability=1. To be able to interoperate
          fully with gateways to circuit switched networks (for
          example, GSM networks), transmissions with restricted
          mode changes (mode-change-capability=2) are required.
          Thus, clients RECOMMENDED to have the capability to
          support transmission according to
          mode-change-capability=2.

mode-change-neighbor: Permissible values are 0 and 1. If 1, the
          sender SHOULD only perform mode changes to the
          neighboring modes in the active codec mode set.

          Neighboring modes are the ones closest in bit rate to
          the current mode, either the next higher or next lower
          rate. If 0 or if not present, change between any two
          modes in the active codec mode set is allowed.

maxptime: The maximum amount of media which can be encapsulated
          in a payload packet, expressed as time in milliseconds.
          The time is calculated as the sum of the time that the
          media present in the packet represents. The time SHOULD
          be an integer multiple of the frame size. If this
          parameter is not present, the sender MAY encapsulate any
          number of speech frames into one RTP packet.

crc: Permissible values are 0 and 1. If 1, frame CRCs SHALL be
          included in the payload. If 0 or not present, CRCs
          SHALL NOT be used. If crc=1, this also implies
          automatically that octet-aligned operation SHALL be used
          for the session.

robust-sorting: Permissible values are 0 and 1. If 1, the
          payload SHALL employ robust payload sorting. If 0 or if
          not present, simple payload sorting SHALL be used. If
          robust-sorting=1, this also implies automatically that
          octet-aligned operation SHALL be used for the session.

interleaving: Indicates that frame-block level interleaving SHALL
          be used for the session, and its value defines the
          maximum number of frame-blocks allowed in an
          interleaving group (see Section 4.4.1). If this
          parameter is not present, interleaving SHALL NOT be
          used. The presence of this parameter also implies
          automatically that octet-aligned operation SHALL be
          used.

ptime: see RFC 4566 [11].

channels: The number of audio channels. The possible values
          (1-6) and their respective channel order is specified in
          Section 4.1 in [12]. If omitted, it has the default
          value of 1.

max-red: The maximum duration in milliseconds that elapses between
          the primary (first) transmission of a frame and any
          redundant transmission that the sender will use. This
          parameter allows a receiver to have a bounded delay when
          redundancy is used. Allowed values are between 0 (no
          redundancy will be used) and 65535. If the parameter is
          omitted, no limitation on the use of redundancy is
          present.

Encoding considerations:
    The Audio data is binary data, and must be encoded for non-
    binary transport; the Base64 encoding is suitable for email.
    When used in RTP context the data is framed as defined in [14].

Security considerations:
    See Section 7 of RFC 4867.

Public specification:
    RFC 4867
    3GPP TS 26.090, 26.092, 26.093, 26.101

Applications that use this media type:
     This media type is used in numerous applications needing
     transport or storage of encoded voice. Some examples include;
     Voice over IP, streaming media, voice messaging, and voice
     recording on digital cameras.

Additional information:
     The following applies to stored-file transfer methods:

     Magic numbers:
        single-channel:
           ASCII character string "#!AMR\n"
           (or 0x2321414d520a in hexadecimal)
        multi-channel:
           ASCII character string "#!AMR_MC1.0\n"
           (or 0x2321414d525F4D43312E300a in hexadecimal)
     File extensions: amr, AMR
     Macintosh file type code: "amr " (fourth character is space)

     AMR speech frames may also be stored in the file format "3GP"
     defined in 3GPP TS 26.244 [31], which is identified using the
     media types "audio/3GPP" or "video/3GPP" as registered by RFC
     3839 [32].

Person & email address to contact for further information:
     Magnus Westerlund <magnus.westerlund@ericsson.com>
     Ari Lakaniemi <ari.lakaniemi@nokia.com>

Intended usage: COMMON.
     This media type is widely used in streaming, VoIP, and messaging
     applications on many types of devices.

Restrictions on usage:
     When this media type is used in the context of transfer over
     RTP, the RTP payload format specified in Section 4 SHALL be
     used. In all other contexts, the file format defined in Section
     5 SHALL be used.

Author:
     Magnus Westerlund <magnus.westerlund@ericsson.com>
     Ari Lakaniemi <ari.lakaniemi@nokia.com>

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

8.2. AMR-WB Media Type Registration

   The media type for the Adaptive Multi-Rate Wideband (AMR-WB) codec is
   allocated from the IETF tree since AMR-WB is a widely used speech
   codec in general VoIP and messaging applications. This media type
   registration covers both real-time transfer via RTP and non-real-
   time transfers via stored files.

   Note, any unspecified parameter MUST be ignored by the receiver.

   Media Type name: audio

   Media subtype name: AMR-WB

   Required parameters: none

   Optional parameters:

      These parameters apply to RTP transfer only.

      octet-align: Permissible values are 0 and 1. If 1, octet-aligned
              operation SHALL be used. If 0 or if not present,
              bandwidth-efficient operation is employed.

      mode-set: Restricts the active codec mode set to a subset of all
              modes, for example, to be able to support transport
              channels such as GSM networks in gateway use cases.
              Possible values are a comma-separated list of modes from
              the set: 0,...,8 (see Table 1a [4]). The SID frame type
              9, SPEECH_LOST (frame type 14), and NO_DATA (frame type
              15) are never included in the mode set, but can always
              be used. If mode-set is specified, it MUST be abided,
              and frames encoded with modes outside of the subset MUST
              NOT be sent in any RTP payload or used in codec mode
              requests. If not present, all codec modes are allowed
              for the payload type.

      mode-change-period: Specifies a number of frame-blocks, N (1 or
              2), that is the frame-block period at which codec mode
              changes are allowed for the sender. The initial phase of
              the interval is arbitrary, but changes must be separated
              by multiples of N frame-blocks, i.e., a value of 2
              allows the sender to change mode every second frame-
              block. The value of N SHALL be either 1 or 2. If this
              parameter is not present, mode changes are allowed at
              Any time during the session, i.e., N=1.

mode-change-capability: Specifies if the client is capable to
        transmit with a restricted mode change period. The
        parameter may take value of 1 or 2. A value of 1
        indicates that the client is not capable of restricting
        the mode change period to 2, and that the codec mode may
        be changed at any point. A value of 2 indicates that the
        client has the capability to restrict the mode change
        period to 2, and thus that the client can correctly
        interoperate with a receiver requiring a mode-change-
        period=2. If this parameter is not present, the mode-
        change restriction capability is not supported, i.e.
        mode-change-capability=1. To be able to interoperate
        fully with gateways to circuit switched networks (for
        example, GSM networks), transmissions with restricted
        mode changes (mode-change-capability=2) are required.
        Thus, clients are RECOMMENDED to have the capability to
        support transmission according to
        mode-change-capability=2.

mode-change-neighbor: Permissible values are 0 and 1. If 1, the
        sender SHOULD only perform mode changes to the
        neighboring modes in the active codec mode set.
        Neighboring modes are the ones closest in bit rate to
        the current mode, either the next higher or next lower
        rate. If 0 or if not present, change between any two
        modes in the active codec mode set is allowed.

maxptime: The maximum amount of media which can be encapsulated
        in a payload packet, expressed as time in milliseconds.
        The time is calculated as the sum of the time that the
        media present in the packet represents. The time SHOULD
        be an integer multiple of the frame size. If this
        parameter is not present, the sender MAY encapsulate any
        number of speech frames into one RTP packet.

crc: Permissible values are 0 and 1. If 1, frame CRCs SHALL be
        included in the payload. If 0 or not present, CRCs
        SHALL NOT be used. If crc=1, this also implies
        automatically that octet-aligned operation SHALL be used
        for the session.

robust-sorting: Permissible values are 0 and 1. If 1, the
        payload SHALL employ robust payload sorting. If 0 or if
        not present, simple payload sorting SHALL be used. If
        robust-sorting=1, this also implies automatically that
        octet-aligned operation SHALL be used for the session.

interleaving: Indicates that frame-block level interleaving SHALL
          be used for the session, and its value defines the
          maximum number of frame-blocks allowed in an
          interleaving group (see Section 4.4.1). If this
          parameter is not present, interleaving SHALL NOT be
          used. The presence of this parameter also implies
          automatically that octet-aligned operation SHALL be
          used.

   ptime: see RFC 2327 [11].

   channels: The number of audio channels. The possible values
          (1-6) and their respective channel order is specified in
          Section 4.1 in [12]. If omitted, it has the default
          value of 1.

   max-red: The maximum duration in milliseconds that elapses between
          the primary (first) transmission of a frame and any
          redundant transmission that the sender will use. This
          parameter allows a receiver to have a bounded delay when
          redundancy is used. Allowed values are between 0 (no
          redundancy will be used) and 65535. If the parameter is
          omitted, no limitation on the use of redundancy is
          present.

Encoding considerations:
     The Audio data is binary data, and must be encoded for non-
     binary transport; the Base64 encoding is suitable for email.
     When used in RTP context the data is framed as defined in [14].

Security considerations:
     See Section 7 of RFC 4867.

Public specification:
     RFC 4867
     3GPP TS 26.190, 26.192, 26.193, 26.201

Applications that use this media type:
     This media type is used in numerous applications needing
     transport or storage of encoded voice. Some examples include;
     Voice over IP, streaming media, voice messaging, and voice
     recording on digital cameras.

    Additional information:
        The following applies to stored-file transfer methods:

        Magic numbers:
          single-channel:
          ASCII character string "#!AMR-WB\n"
          (or 0x2321414d522d57420a in hexadecimal)
          multi-channel:
          ASCII character string "#!AMR-WB_MC1.0\n"
          (or 0x2321414d522d57425F4D43312E300a in hexadecimal)
        File extensions: awb, AWB
        Macintosh file type code: amrw
        Object identifier or OID: none

        AMR-WB speech frames may also be stored in the file format "3GP"
        defined in 3GPP TS 26.244 [31] and identified using the media
        type "audio/3GPP" or "video/3GPP" as registered by RFC 3839
        [32].

    Person & email address to contact for further information:
        Magnus Westerlund <magnus.westerlund@ericsson.com>
        Ari Lakaniemi <ari.lakaniemi@nokia.com>

    Intended usage: COMMON.
        This media type is widely used in streaming, VoIP, and messaging
        applications on many types of devices.

    Restrictions on usage:
        When this media type is used in the context of transfer over
        RTP, the RTP payload format specified in Section 4 SHALL be
        used. In all other contexts, the file format defined in Section
        5 SHALL be used.

    Author:
        Magnus Westerlund <magnus.westerlund@ericsson.com>
        Ari Lakaniemi <ari.lakaniemi@nokia.com>

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

8.3. 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)
   [11], which is commonly used to describe RTP sessions. When SDP is
   used to specify sessions employing the AMR or AMR-WB codec, the
   mapping is as follows:

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

- The media subtype (payload format name) goes in SDP "a=rtpmap"
  as the encoding name. The RTP clock rate in "a=rtpmap" MUST be
  8000 for AMR and 16000 for AMR-WB, and the encoding parameters
  (number of channels) MUST either be explicitly set to N or
  omitted, implying a default value of 1. The values of N that
  are allowed are specified in Section 4.1 in [12].

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

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

8.3.1. Offer-Answer Model Considerations

   The following considerations apply when using SDP Offer-Answer
   procedures to negotiate the use of AMR or AMR-WB payload in RTP:

- Each combination of the RTP payload transport format
  configuration parameters (octet-align, crc, robust-sorting,
  interleaving, and channels) is unique in its bit-pattern and
  not compatible with any other combination. When creating an
  offer in an application desiring to use the more advanced
  features (crc, robust-sorting, interleaving, or more than one
  channel), the offerer is RECOMMENDED to also offer a payload
  type containing only the octet-aligned or bandwidth-efficient
  configuration with a single channel. If multiple
  configurations are of interest to the application, they may all
  be offered; however, care should be taken not to offer too many
  payload types. An SDP answerer MUST include, in the SDP answer
  for a payload type, the following parameters unmodified from
  the SDP offer (unless it removes the payload type): "octet-
  align"; "crc"; "robust-sorting"; "interleaving"; and
  "channels". The SDP offerer and answerer MUST generate AMR or
  AMR-WB packets as described by these parameters.

- The "mode-set" parameter can be used to restrict the set of
  active AMR/AMR-WB modes used in a session. This functionality
  is primarily intended for gateways to access networks such as
  GSM or 3GPP UMTS, where the access network may be capable of
  supporting only a subset of AMR/AMR-WB modes. The 3GPP
  preferred codec configurations are defined in 3GPP TS 26.103
  [25], and it is RECOMMENDED that other networks also needing to
  restrict the mode set follow the preferred codec configurations
  defined in 3GPP for greatest interoperability.

The parameter is bi-directional, i.e., the restricted set
applies to media both to be received and sent by the declaring
entity. If a mode set was supplied in the offer, the answerer
SHALL return the mode-set unmodified or reject the payload
type. However, the answerer is free to choose a mode-set in
the answer only if no mode-set was supplied in the offer for a
unicast two-peer session. The mode-set in the answer is
binding both for offerer and answerer. Thus, an offerer
supporting all modes and subsets SHOULD NOT include the mode-
set parameter. For any other offerer it is RECOMMENDED to
include each mode-set it can support as a separate payload type
within the offer. For multicast sessions, the answerer SHALL
only participate in the session if it supports the offered
mode-set. Thus, it is RECOMMENDED that any offer for a
multicast session include only the mode-set it will require the
answerers to support, and that the mode-set be likely to be
supported by all participants.

- The parameters "mode-change-period" and "mode-change-
  capability" are intended to be used in sessions with gateways,
  for example, when interoperating with GSM networks. Both
  parameters are declarative and are combined to allow a session
  participant to determine if the payload type can be supported.
  The mode-change-period will indicate what the offerer or
  answerer requires of data it receives, while the mode-change-
  capability indicates its transmission capabilities.

  A mode-change-period=2 in the offer indicates a requirement on
  the answerer to send with a mode-change period of 2, i.e.,
  support mode-change-capability=2. If the answerer requires
  mode-change-period=2, it SHALL only include it in the answer if
  the offerer either has indicated support with mode-change-
  capability=2 or has indicated mode-change-period=2; otherwise,
  the payload type SHALL be rejected. An offerer that supports
  mode-change-capability=2 SHALL include the parameter in all
  offers to ensure the greatest possible interoperability, unless
  it includes mode-change-period=2 in the offer. The mode-
  change-capability SHOULD be included in answers. It is then
  indicating the answerer's capability to transmit with that
  mode-change-period for the provided payload format
  configuration. The information is useful in future
  re-negotiation of the payload formats.

- The parameter "mode-change-neighbor" is a recommendation to
  restrict the switching of codec modes to its neighbor and
  SHOULD be followed. It is intended to be used in gateway
  scenarios (for example, to GSM networks) where the support of

this parameter and the operations it implies improves
interoperability.

"mode-change-neighbor" is a declarative parameter. By
including the parameter, the offerer or answerer indicates that
it desires to receive streams with "mode-change-neighbor"
restrictions.

- In most cases, the parameters "maxptime" and "ptime" will not
affect interoperability; however, the setting of the parameters
can affect the performance of the application. The SDP offer-
answer handling of the "ptime" parameter is described in RFC
3264 [13]. The "maxptime" parameter MUST be handled in the
same way.

- The parameter "max-red" is a stream property parameter. For
send-only or send-recv unicast media streams, the parameter
declares the limitation on redundancy that the stream sender
will use. For recvonly streams, it indicates the desired value
for the stream sent to the receiver. The answerer MAY change
the value, but is RECOMMENDED to use the same limitation as the
offer declares. In the case of multicast, the offerer MAY
declare a limitation; this SHALL be answered using the same
value. A media sender using this payload format is RECOMMENDED
to always include the "max-red" parameter. This information is
likely to simplify the media stream handling in the receiver.
This is especially true if no redundancy will be used, in which
case "max-red" is set to 0. As this parameter was not defined
originally, some senders will not declare this parameter even
if it will limit or not send redundancy at all.

- Any unknown parameter in an offer SHALL be removed in the
answer.

8.3.2. Usage of Declarative SDP

In declarative usage, like SDP in RTSP [29] or SAP [30], the
parameters SHALL be interpreted as follows:

- The payload format configuration parameters (octet-align, crc,
robust-sorting, interleaving, and channels) are all declarative,
and a participant MUST use the configuration(s) that is provided
for the session. More than one configuration may be provided if
necessary by declaring multiple RTP payload types; however, the
number of types should be kept small.

- Any restriction of the AMR or AMR-WB encoder mode-switching and
  mode usage through the "mode-set", and "mode-change-period" MUST
  be followed by all participants of the session. The restriction
  indicated by "mode-change-neighbor" SHOULD be followed. Please
  note that such restrictions may be necessary if gateways to other
  transport systems like GSM participate in the session. Failure to
  consider such restrictions may result in failure for a peer behind
  such a gateway to correctly receive all or parts of the session.
  Also, if different restrictions are needed by different peers in
  the same session (unless a common subset of the restrictions
  exists), some peer will not be able to participate. Note that the
  usage of mode-change-capability is meaningless when no negotiation
  exists, and can thus be excluded in any declarations.

- Any "maxptime" and "ptime" values should be selected with care to
  ensure that the session's participants can achieve reasonable
  performance.

- The usage of "max-red" puts a global upper limit on the usage of
  redundancy that needs to be followed by all that understand the
  parameter. However, due to the late addition of this parameter,
  it may be ignored by some implementations.

8.3.3. Examples

   Some example SDP session descriptions utilizing AMR and AMR-WB
   encodings follow. In these examples, long a=fmtp lines are folded to
   meet the column width constraints of this document; the backslash
   ("\") at the end of a line and the carriage return that follows it
   should be ignored.

   In an example of the usage of AMR in a possible GSM gateway-to-
   gateway scenario, the offerer is capable of supporting three
   different mode-sets and needs the mode-change-period to be 2 in
   combination with mode-change-neighbor restrictions. The other
   gateway can only support two of these mode-sets and removes the
   payload type 97 in the answer. If the offering GSM gateway only
   supports a single mode-set active at the same time, it should
   consider doing the 1 out of N selection procedures described in
   Section 10.2 of [13]:

```
Offer:

 m=audio 49120 RTP/AVP 97 98 99
 a=rtpmap:97 AMR/8000/1
 a=fmtp:97 mode-set=0,2,5,7; mode-change-period=2; \
   mode-change-capability=2; mode-change-neighbor=1
 a=rtpmap:98 AMR/8000/1
 a=fmtp:98 mode-set=0,2,3,6; mode-change-period=2; \
   mode-change-capability=2; mode-change-neighbor=1
 a=rtpmap:99 AMR/8000/1
 a=fmtp:99 mode-set=0,2,3,4; mode-change-period=2; \
   mode-change-capability=2; mode-change-neighbor=1
 a=maxptime:20

Answer:

 m=audio 49120 RTP/AVP 98 99
 a=rtpmap:98 AMR/8000/1
 a=fmtp:98 mode-set=0,2,3,6; mode-change-period=2; \<
   mode-change-capability=2; mode-change-neighbor=1
 a=rtpmap:99 AMR/8000/1
 a=fmtp:99 mode-set=0,2,3,4; mode-change-period=2; \
   mode-change-capability=2; mode-change-neighbor=1
 a=maxptime:20
```

The following example shows the usage of AMR between a non-GSM
endpoint and a GSM gateway. The non-GSM offerer requires no
restrictions of the mode-change-period or mode-change-neighbor, but
must signal its mode-change-capability in the offer and abide by
those restrictions in the answer.

```
Offer:

 m=audio 49120 RTP/AVP 97
 a=rtpmap:97 AMR/8000/1
 a=fmtp:97 mode-change-capability=2
 a=maxptime:20

Answer:

 m=audio 49120 RTP/AVP 97
 a=rtpmap:97 AMR/8000/1
 a=fmtp:97 mode-set=0,2,4,7; mode-change-period=2; \
   mode-change-capability=2; mode-change-neighbor=1
 a=maxptime:20
```

Example of usage of AMR-WB in a possible VoIP scenario where UEP may
be used (99) and a fallback declaration (98):

```
 m=audio 49120 RTP/AVP 99 98
 a=rtpmap:98 AMR-WB/16000
 a=fmtp:98 octet-align=1; mode-change-capability=2
 a=rtpmap:99 AMR-WB/16000
 a=fmtp:99 octet-align=1; crc=1; mode-change-capability=2
```

Example of usage of AMR-WB in a possible streaming scenario (two
channel stereo):

```
 m=audio 49120 RTP/AVP 99
 a=rtpmap:99 AMR-WB/16000/2
 a=fmtp:99 interleaving=30
 a=maxptime:100
```

Note that the payload format (encoding) names are commonly shown in
upper case. MIME subtypes are commonly shown in lower case. 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.

9. IANA Considerations

Two media types (audio/AMR and audio/AMR-WB) have been updated; see
Section 8.

10. Changes from RFC 3267

The differences between RFC 3267 and this document are as follows:

- Added clarification of behavior in regards to mode change period
  and mode-change neighbor that is expected from an IP client; see
  Section 4.5.

- Updated the maxptime for better clarification. The sentence that
  previously read: "The time SHOULD be a multiple of the frame
  size." now says "The time SHOULD be an integer multiple of the
  frame size." This should have no impact on interoperability.

- Updated the definition of the mode-set parameter for
  clarification.

- Restricted the values for mode-change-period to 1 or 2, which are
  the values used in circuit-switched AMR systems.

- Added a new media type parameter Mode-Change-Capability that
  defaults to 1, which is the assumed behavior of any non-updated
  implementation. This enables the offer-answer procedures to work.

- Changed mode-change-neighbor to indicate a recommended behavior
  rather than a required one.

- Added an Offer-Answer Section, see Section 8.3.1. This will have
  implications on the interoperability to implementations that have
  guessed how to perform offer/answer negotiation of the payload
  parameters.

- Clarified and aligned the unequal detection usage with the
  published UDP-Lite specification in Sections 3.6.1 and 4.4.2.1.
  This included replacing a normative statement about packet
  handling with an informative paragraph with a reference to UDP-
  Lite.

- Clarified the bit order in the CRC calculation in Section 4.4.2.1.

- Corrected the reference in Section 5.3 for the Q and FT fields.

- Changed the padding bit definition in Sections 4.4.2 and 5.3 so
  that it is clear that they shall be ignored.

- Added a clarification that comfort noise frames with frame type 9,
  10, and 11 SHALL NOT be used in the AMR file format.

- Clarified in Section 4.3.2 that the rules about not sending
  NO_DATA frames do apply for all payload format configurations with
  the exception of the interleaved mode.

- The reference list has been updated to now published RFCs: RFC
  3448, RFC 3550, RFC 3551, RFC 3711, RFC 3828, and RFC 4566. A
  reference to 3GPP TS 26.101 has also been added.

- Added notes in storage format section and media type registration
  that AMR and AMR-WB frames can also be stored in the 3GP file
  format.

- Added a media type parameter "max-red" that allows the sender to
  declare a bounded usage of redundancy. This parameter allows a
  receiver to optimize its function as it will know if redundancy
  will be used or not. If it is used, the maximum extra delay
  introduced by the sender (that is needed to be considered by the
  receiver to fully utilize the redundancy) will be known. The
  addition of this parameter should have no negative effects on
  older implementations as they are mandated to ignore unknown

parameters per RFC 3267. In addition, older implementations are
      required to operate as if the value of max-red is unknown and
      possibly infinite.

   - Updated the media type registration to comply with the new
      registration rules.

   - Moved section on decoding validation from Security Considerations
      to Implementation Considerations, where it makes more sense.

   - Clarified the application of encryption, integrity protection, and
      authentication mechanism to the payload.

11. Acknowledgements

   The authors would like to thank Petri Koskelainen, Bernhard Wimmer,
   Tim Fingscheidt, Sanjay Gupta, Stephen Casner, and Colin Perkins for
   their significant contributions made throughout the writing and
   reviewing of RFC 3267 and this replacement. The authors would also
   like to thank Richard Ejzak, Thomas Belling, and Gorry Fairhurst for
   their input on this replacement of RFC 3267.

12. References

12.1. Normative References

   [1] 3GPP TS 26.090, "Adaptive Multi-Rate (AMR) speech transcoding",
        version 4.0.0 (2001-03), 3rd Generation Partnership Project
        (3GPP).

   [2] 3GPP TS 26.101, "AMR Speech Codec Frame Structure", version
        4.1.0 (2001-06), 3rd Generation Partnership Project (3GPP).

   [3] 3GPP TS 26.190 "AMR Wideband speech codec; Transcoding
        functions", version 5.0.0 (2001-03), 3rd Generation Partnership
        Project (3GPP).

   [4] 3GPP TS 26.201 "AMR Wideband speech codec; Frame Structure",
        version 5.0.0 (2001-03), 3rd Generation Partnership Project
        (3GPP).

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

   [6] 3GPP TS 26.093, "AMR Speech Codec; Source Controlled Rate
        operation", version 4.0.0 (2000-12), 3rd Generation Partnership
        Project (3GPP).

   [7] 3GPP TS 26.193 "AMR Wideband Speech Codec; Source Controlled
        Rate operation", version 5.0.0 (2001-03), 3rd Generation
        Partnership Project (3GPP).

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

   [9] 3GPP TS 26.092, "AMR Speech Codec; Comfort noise aspects",
        version 4.0.0 (2001-03), 3rd Generation Partnership Project
        (3GPP).

   [10] 3GPP TS 26.192 "AMR Wideband speech codec; Comfort Noise
        aspects", version 5.0.0 (2001-03), 3rd Generation Partnership
        Project (3GPP).

   [11] Handley, M., Jacobson, V., and C. Perkins, "SDP: Session
        Description Protocol", RFC 4566, July 2006.

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

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

   [14] Freed, N. and J. Klensin, "Media Type Specifications and
        Registration Procedures", BCP 13, RFC 4288, December 2005.

   [15] Casner, S., "Media Type Registration of RTP Payload Formats",
        RFC 4855, February 2007.

12.2. Informative References

   [16] GSM 06.60, "Enhanced Full Rate (EFR) speech transcoding",
        version 8.0.1 (2000-11), European Telecommunications Standards
        Institute (ETSI).

   [17] ANSI/TIA/EIA-136-Rev.C, part 410 - "TDMA Cellular/PCS Radio
        Interface, Enhanced Full Rate Voice Codec (ACELP)". Formerly
        IS-641. TIA published standard, June 1 2001.

   [18] ARIB, RCR STD-27H, "Personal Digital Cellular Telecommunication
        System RCR Standard", Association of Radio Industries and
        Businesses (ARIB).

   [19] Larzon, L-A., Degermark, M., Pink, S., Jonsson, L-E., and G.
        Fairhurst, "The Lightweight User Datagram Protocol (UDP-Lite)",
        RFC 3828, July 2004.

[20] 3GPP TS 25.415 "UTRAN Iu Interface User Plane Protocols",
     version 4.2.0 (2001-09), 3rd Generation Partnership Project
     (3GPP).

[21] Handley, M., Floyd, S., Padhye, J., and J. Widmer, "TCP Friendly
     Rate Control (TFRC): Protocol Specification", RFC 3448, January
     2003.

[22] Li, A., et al., "An RTP Payload Format for Generic FEC with
     Uneven Level Protection", Work in Progress.

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

[24] 3GPP TS 26.102, "AMR speech codec interface to Iu and Uu",
     version 4.0.0 (2001-03), 3rd Generation Partnership Project
     (3GPP).

[25] 3GPP TS 26.202, "AMR Wideband speech codec; Interface to Iu and
     Uu", version 5.0.0 (2001-03), 3rd Generation Partnership Project
     (3GPP).

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

[27] 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.

[28] 3GPP TS 26.103, "Speech codec list for GSM and UMTS", version
     5.5.0 (2004-09), 3rd Generation Partnership Project (3GPP).

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

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

[31] 3GPP TS 26.244, "3GPP file format (3GP)", version 6.1.0 (2004-
     09), 3rd Generation Partnership Project (3GPP).

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

[33] Kent, S. and K. Seo, "Security Architecture for the Internet
     Protocol", RFC 4301, December 2005.

   [34] Dierks, T. and E. Rescorla, "The Transport Layer Security (TLS)
        Protocol Version 1.1", RFC 4346, April 2006.

   ETSI documents are available from <http://www.etsi.org/>.
   3GPP documents are available from <http://www.3gpp.org/>.
   TIA documents are available from <http://www.tiaonline.org/>.

Authors' Addresses

   Johan Sjoberg
   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

   Qiaobing Xie
   Motorola, Inc.
   1501 W. Shure Drive, 2-B8
   Arlington Heights, IL 60004, USA

   Phone: +1-847-632-3028
   EMail: Qiaobing.Xie@motorola.com

Full Copyright Statement

Intellectual Property

Acknowledgement