PAYLOAD

Ali Begen
Roni Even
IETF 81
Quebec July 2011

Payload

- Formed since IETF 79
- Tasked with the specification and maintenance of payload formats for use with RTP.
- Please join the payload mailing list at https://www.ietf.org/mailman/listinfo/payload
  - to be able to send and receive emails to the list, we see people trying to post who did not join.

Milestones

- Jan 2011 Submit RTP Payload Format for MIDI for Proposed Standard - Done
- Feb 2011 Submit How to Write an RTP Payload Format for Informational
- Feb 2011 Submit RTP Payload Format for MPEG-4 Audio/Visual Streams for Proposed Standard
- Mar 2011 Submit RTP Payload Format for EVBR/G.718 for Proposed Standard
- Mar 2011 Submit RTP Payload Format for Enhanced Variable Rate Narrowband-Wideband Codec (EVRC-NW) for Proposed Standard
- Mar 2011 Submit RTP Payload Format for Bluetooth's SBC audio codec for Proposed Standard
- Apr 2011 Submit RTP Payload Format for MPEG2-TS preamble for Proposed Standard
- Apr 2011 Submit RTP Payload Format for DV (IEC 61834) Video for Proposed Standard
- Apr 2011 Submit RTP Payload Format for the iSAC codec for Proposed Standard
- Apr 2011 Submit RTP profile for the carriage of SMPTE 336M data for Proposed Standard
- Jun 2011 Submit RTP Payload Format for MVC Video for Proposed Standard
- Aug 2011 Submit RTP Payload Format for VP8 Video for Proposed Standard

Status of WG drafts

- RFC published since IETF80
  - RFC 6295 draft-ietf-payload-rfc4695-bis (MIDI)
- RTP Payloads
  - draft-ietf-payload-rfc3016bis-01 (MPEG4) – In IESG review – waiting for revised draft.
  - draft-ietf-avt-rtp-evrc-nw-03 – WGLC after IETF81
  - draft-ietf-payload-rtp-klv-01 – was in WGLC did not get enough review - need reviewers
  - draft-ietf-payload-rtp-mvc-00 -
  - draft-ietf-payload-rtp-sbc-00 - need reviewers
  - draft-ietf-avt-payload-g718-00 – ready for WGLC?
  - draft-ietf-payload-rfc3189bis-01 (DV video) – will go to publication.
  - draft-ietf-payload-vp8-01 - need reviews

We need people to review documents!

Other documents

– draft-spittka-payload-rtp-opus-00 - had comments in the mailing list. Need a new revision before adopting as WG document.

XRBLOCK

- Formed since IETF 79
- Handle RTCP Extended Report (XR) Blocks
- 6 current individual drafts – most of them addressing video quality.
- 10 avt drafts (now expired) were basis for milestones – mostly audio related.

Status

- Revive expired drafts based on monitoring architecture.
- Need to create milestones for the video quality Xrblocks.

We need people to review and edit documents!

draft-ramalho-payload-g7110-00

27 July 2011

Michael A. Ramalho, Ed.
Paul Jones
Noboru Harada
Muthu Arul Mozhi Perumal
Miao Lei

What this Draft is About

- Payload format specification for ITU-T G.711.0
  - Media type registrations, security considerations, etc.
- RTP payload and signaling considerations for:
  - "G.711.0 Compression Segments"
  - Potential RTP specification of multiple G.711 channels within one G.711.0 payload.
- Storage Mode Formats for G.711.0
  - Two formats proposed
  - Definition of a "G.711 erasure frame"

Essential G.711.0

- G.711.0 is a data compression algorithm especially designed for A-law or Mu-law G.711 VoIP payloads (i.e., not a generic compression).
- Lossless => Lossless for <u>ANY</u> payload (including random data in DS0s).
- Stateless => Compression not dependent on previous frames.
  - No error-propagation at decoder possible due to lost prior packets.
- "Self-describing" => G.711 regenerated <u>WITHOUT</u> access to signaling.
- Two Dominant Use Cases:
  - <u>End-to-End:</u> G.711.0 negotiated as "if it were a codec"
    - Nearly identical to G.711 RTP specification (exception is dynamic PT)
  - <u>In The Middle:</u> Can be employed multiple times within an end-to-end G.711 session.
    - Without endpoint or call agent knowledge
    - With endpoint or call agent knowledge
    - With no degradation of voice quality relative to G.711
- Most open issues for "In The Middle" case (next slide).

G.711.0 Compression Segment

```
                                       *********************
                                       *                   *
|------------|    |--------------------| *  |---------------| *
|     A:     |    |         B:         | *  |  C: G.711.0   | *
|  SENDING   |--->|    ZERO OR MORE    |---->|  Compression  | *
|   G.711    |    |   ROUTING AND/OR   | *  |   PT=[0|8]    | *
|  ENDPOINT  |    |   SWITCHING HOPS   | *  |  CHANGED TO   | *
|            |    | AND/OR MIDDLEBOXES | *  |    PT = Q     | *
|____________|    |____________________| *  |_______________| *
                                       *            |          *
                                      *             |          *
                            ********* *             |          *
                            *                       |          *
                            *                       V          *
                            *      |--------------------|      *
                            *      |         D:         |      *
                  G.711.0   *      |    ZERO OR MORE    |      *
               COMPRESSION  *      |   ROUTING AND/OR   |      *
                  SEGMENT   *      |   SWITCHING HOPS   |      *
                 (payload   *      | AND/OR MIDDLEBOXES |      *
                   only)    *      |____________________|      *
                            *                       |          *
                            *                       |          *
                            ***********             |          *
                                      *             |          *
                                       *            V          *
|------------|    |--------------------| *  |---------------| *
|     G:     |    |         F:         | *  |  E: G.711.0   | *
| RECEIVING  |<---|    ZERO OR MORE    |<----| DECOMPRESSION | *
|   G.711    |    |   ROUTING AND/OR   | *  |    PT = Q     | *
|  ENDPOINT  |    |   SWITCHING HOPS   | *  | CHANGED BACK  | *
|            |    | AND/OR MIDDLEBOXES | *  |  TO PT=[0|8]  | *
|____________|    |____________________| *  |_______________| *
                                       *                   *
                                       *********************
```

- PT = 0; G.711 Mu-law (PCMU)
- PT = 8; G.711 A-law (PCMA)
- May have multiple compression segments on end-to-end connection.
- No Box D => Compression over single link.
- Middleboxes present challenges.
- Typically SBCs must be informed to let traffic other than what it expects to pass (in this case to let PT = Q pass).
- One solution is to put “hint” in the G.711 SDP, such as:

m=audio RTP/AVP 0
a=rtpmap: 0 PCMU
a=fmtp:0 G7110 = Q < the G.711 SDP hint

- Other clever solutions?

Known Open Issues

- Compression Segment Issues with Middleboxes.
  - Provide hint in G711 SDP? Robustness? Can it hurt?
  - Other suggestions/methods?
- Should we specify the multiplexing of *multiple G.711 "channels"* within *one G.711.0 RTP session*?
  - Beneficial for many service provider and enterprise uses
  - Could define via a channels parameter for G.711.0
    - Need to specify a delimiter for the channels within a G.711.0 payload (G.711.0 delimiters available from ITU-T work)
- Definition of storage mode for long recordings

Thank You

(Backup slides follow)

Design Requirements in ITU-T G.711.0 “Terms of Reference”

- Support both G.711 A-law and Mu-law.
- Lossless for ANY payload (including random data in DS0s).
- Accommodates G.711 payload sizes typically used in VoIP.
- Stateless: Compression not dependent on previous frames.
  - No error-propagation at decoder possible due to lost prior packets.
- Algorithmic delay equal to the time represented by G.711 input.
  - No “look-ahead” or per-channel state.
- Self describing G.711.0 output frame.
  - Decoder is NOT dependent on access to signaling.*
  - Encoder is NOT dependent on access to signaling.*
- Bounded expansion for “uncompressible G.711 input frames”.
- Low complexity (<1 WMOPS, 10k memory, 3.6k basic operations).

* Only needs to know A-law or mu-law. Encoder only needs the G.711 input frame.

G.711.0 Basic Operation

G.711 Input Frame
of X bytes

(where X MUST be 40, 80
160, 240 or 320 bytes)

G.711 encode →

← G.711.0 decode

G.711.0 Output Frame
containing 1 to X+1 bytes

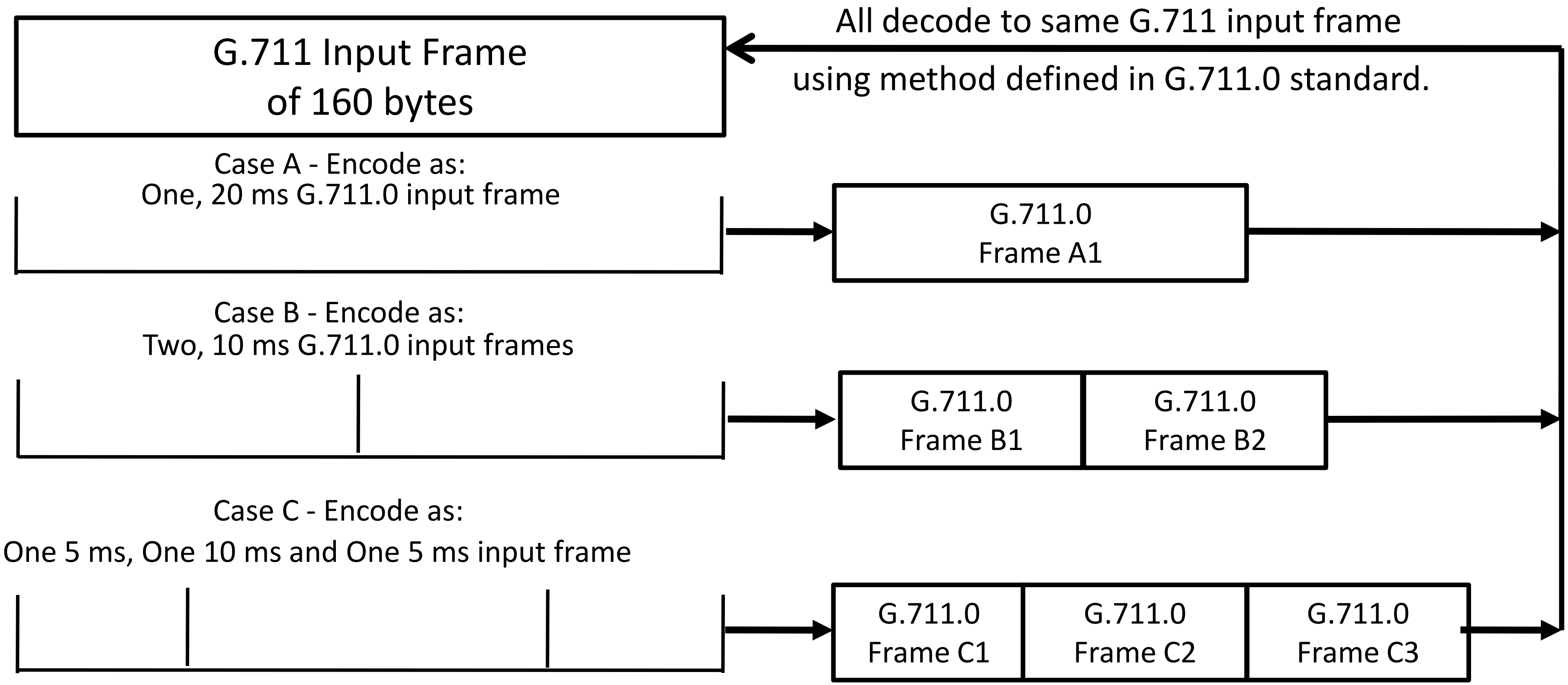
(value dependent on
G.711.0 ability to compress)

<u>At 8k sampling:</u>
40 samples = 5 ms
80 samples = 10 ms
160 samples = 20 ms
240 samples = 30 ms
320 samples = 40 ms

- Mapping is 1:1 in both directions
- G.711.0 is a “Self Describing” encoding:
  - Decoder – without any signaling information - knows how many G.711 source samples to produce
- Optimized for zero-mean acoustic signals, however ...
- Lossless for any G.711 input frame (including random data)

Complex G.711.0 Encoding Example:

20ms/160 bytes of G.711



- A smart encoder may choose ANY combination of sub input frame sizes to determine which compresses best (usually the largest does)
- As a result, ANY integer number of 5 ms of G.711 can be encoded and placed in a RTP payload

G711.0 Internal Design & Compression Results

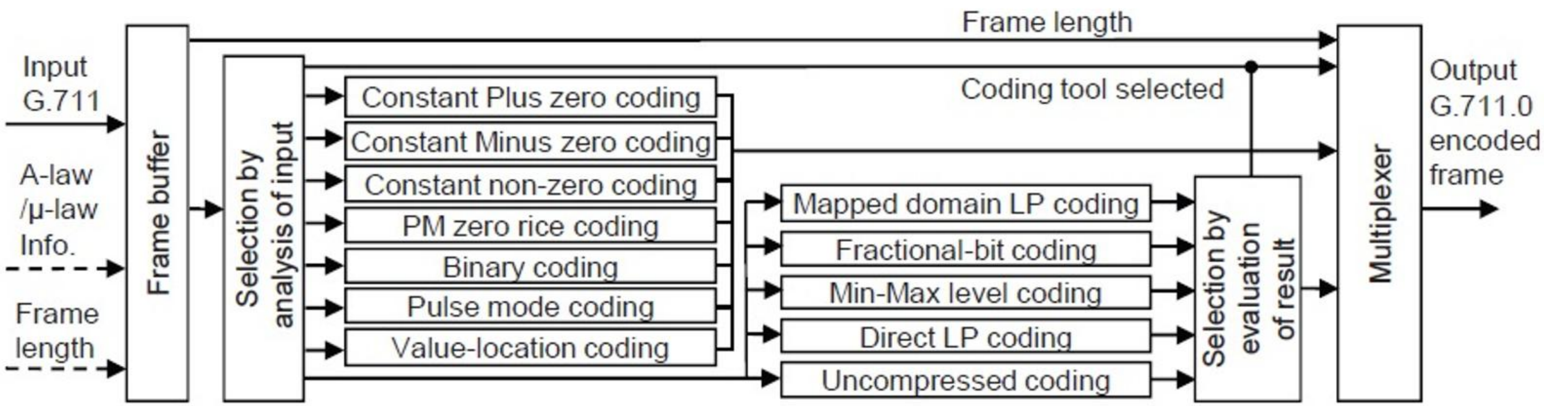


Fig. 1. High-level block diagram of the G.711.0 encoder.

Test category		Compression ratio [%]	
		A-law	µ-law
(a1): Clean speech	-16 dBoV	59.56 %	50.67 %
	-26 dBoV	69.39 %	60.62 %
	-36 dBoV	77.01 %	72.55 %
(a2): Noisy speech	SNR 15 dB	50.90 %	44.52 %
	SNR 20 dB	54.43 %	47.15 %
	SNR 25 dB	60.64 %	52.43 %
(a1) and (a2) conditions in total		57.55 %	50.24 %
(a3): Tandem conditions in total		60.08 %	54.52 %
(b): Recorded (NTT) µ-law corpus		-	50.83 %

Note: Conservative because averaged over all G.711.0 frame lengths (of 5ms, 10ms, 20ms and 30ms).
Results for 20ms are better by about 2%. A-law compresses better due to coarser quantization at low levels.

N. Harada, Y. Yamamoto, T. Moriya, Y. Hiwasaki, M. A. Ramalho, L. Netsch, Y. Stachurski, Miao Lei, H. Taddei, and Q. Fengyan,
"Emerging ITU-T Standard G.711.0 - Lossless Compression of G.711 Pulse Code Modulation"
International Conference on Acoustics Speech and Signal Processing (ICASSP), March 2010, ISBN 978-1-4244-4244-4295-9