# Real-time text media handling in multi-party conferences

draft-hellstrom-mmusic-multi-party-rtt-00

Presentation in IETF AVTCORE
22 November 2019

# Real-time text (RTT) - Basic facts

- **Transmit while you create text** - no special user action for sending
- Provides **tight interaction** - no waiting for completed messages.
- Required for accessibility reasons but useful for all.
- **Presentation level standard ITU-T T.140** based on UTF-8 and ISO 6429
- **Transport** for SIP by **RTP** packetization RFC 4103. Robustness by redundancy.
- Transport for **WebRTC** in progress in draft-ietf-mmusic-t140-usage-data-channel. Robustness by "reliable" data channel.
- Specified for **NG emergency services, 3GPP IMS, SIP VoIP, relay services.**

# Multi-party RTT

- **Need for multi-party RTT** support is mentioned in many specifications.
- **No approved SDO-document says how.**
- Current user terminals only support two-party view
- Capability negotiation needed for type of multi-party RTT implementation
- Centralized conference server mixing RTT needed when support is negotiated
- Basic requirements: Identify source of text. Separate text from sources. Do not delay text presentation.
- Possibly limited functionality fallback mixing method for cases when terminal has no multi-party support.
- Let user or user app decide on presentation style (see following examples)

# A one-column three-party real-time chat-style view

```
                 _______________________________________________
                |                                               |^|
                |                                               | |
                |[Alice] I am coming on Thursday, my            | |
                |      performance is not until Friday morning.| |
                |                                               | |
                |[Bob] And I on Wednesday evening.              | |
                |                                               | |
                |[Alice] Can we meet on Thursday evening?       | |
                |                                               | |
                |[Eve] Yes, definitely. How about 7pm.          | |
                |     at the entrance of the restaurant         | |
                |     Le Lion Blanc?                            | |
                |[Eve] we can have dinner and then take a walk  | |
                |                                               | |
                | <Eve-typing> But I need to be back to         | |
                |    the hotel by 11 because I need             | |
                |                                               |-|
                | <Bob-typing> I wou                            |-|
                |_______________________________________________|v|
Alice, typing:  | of course, I underst                              |
                |_________________________________________________|
```

# A multi-panel view example

```
 _____________________________________________________________________
|        Bob         |        Eve            |        Alice          |
|____________________|_______________________|_______________________|
|                    |                       |I will arrive by TGV.  |
|My flight is to Orly|                       |Convenient to the main |
|                    |Hi all, can we plan    |station.               |
|                    |for the seminar?       |                       |
|Eve, will you do    |                       |                       |
|your presentation on|                       |                       |
|Friday?             |Yes, Friday at 10.     |                       |
|Fine, wo            |                       |We need to meet befo   |
|____________________________________________________________________|
```

# Requirements

- Applicable to 3GPP IMS, SIP VoIP, NG Emergency services, relay services
- No longer transmission interval than 500 ms when there is text to send
- Indicate possible text loss by standardized mark in stream
- Display so that text is readable and source identified
- Support erasure and correction
- Support for confidentiality, privacy, integrity and reliable source identification
- Implementable in near-future
- Fallback method for current two-party terminals if found feasible.

# draft-hellstrom-mmusic-multi-party-rtt contents

- Methods for coordination of RTP RTT streams - pros and cons
- Methods for coordination of WebRTC RTT streams - pros and cons
- Method for mixing for conference-unaware devices - minimize limitations
- Methods for multi-party RTT capability negotiation - pros and cons
- Identification of source of text
- Presentation aspects for multi-party aware devices
- Presentation aspects for multi-party unaware devices
- Robustness and indication of loss
- Performance
- Security
- IANA considerations

# Goals

- Discuss the different alternative methods for stream mixing and source identification. Agree on preferred methods.
- Discuss the different alternative methods for multi-party capability negotiation. Agree on a preferred method.
- Discuss the method for conference-unaware devices. Tune for least shortcomings..
- The draft touches topics of many working groups. avtcore, mmusic, sipcore. Arrange discussions in a practical way. E.g. have most discussions in mmusic.

# Example discussion topic: coordination of RTP RTT

4.1. RTP Translator sending one RTT stream per participant with own SSRC
Theoretically ideal, but no support from SDP and installed parc.

4.2. RTP Mixer indicating participants in CSRC
Some problems in source of redundancy and marking of loss.

4.3. RTP Mixer indicating participants by a T.140 control code in the stream
Good interop options but may require stream manipulation by mixer.

4.4. Mesh of RTP endpoints. No central mixing.
Theoretically interesting, but not the favoured model by big implementations

4.5. Multiple RTP sessions, one for each participant
Good for separating sources, but high network resource usage

4.6. Mixing for conference-unaware user agents
Required fallback for today's UAs. Optimized solution needed..

# Example discussion topic: Capability negotiation

7.1. Implicit RTT multi-party capability indication (by conference-awareness)
   Use RFC 4575 conference negotiation. Requires that conference model.

7.2. RTT multi-party capability declared by new SIP media-tags
   Works for SIP. Inexact if more than one text media session is declared.

7.3. SDP media attribute for RTT multi-party capability indication
   Works anywhere were SDP is used.

# Considerations

- IETF is the right place: Need for both NENA and 3GPP and others to have a common specification to refer to. IETF is a good common place for all.
- Form: Best Current Practice (BCP) is proposed.
- Urgency: Implementations required early 2021

.

# Looking forward to rapid progress of multi-party RTT.

Thanks,

Gunnar Hellström

Omnitor

gunnar.hellstrom@omnitor.se