XEP-0517: Jingle Synchronized Real-Time Text

Edward Tie
mailto:info@tiedragon.com

2026-06-30
Version 0.1.0

Status	Type	Short Name
Experimental	Standards Track	jingle-rtt-sync

This specification defines a Jingle application extension for negotiating real-time text as part of the same conversational session as audio and video.

Legal

Copyright

Permissions

Warranty

Liability

Conformance

Contents

1 Introduction

Real-time text is already defined for XMPP by In-Band Real Time Text (XEP-0301) [1]. Jingle is already used to negotiate real-time audio and video sessions, most commonly using Jingle RTP Sessions (XEP-0167) [2] and Jingle ICE-UDP Transport Method (XEP-0176) [3]. However, when a client establishes a Jingle audio-video call and sends real-time text as ordinary XMPP messages outside the Jingle session, the user experience can look like one conversation while the protocol state is split into two unrelated paths.
This specification defines a way to negotiate real-time text as a Jingle content in the same session as audio and video. The text content can be human typed RTT, captions, ASR output, interpreter text, translation text or transcript text. The goal is Total Conversation: audio, video and text presented as one conversational unit.
The motivating implementation problem is simple: a call can exist, text can exist, and yet the text might not be part of the negotiated Jingle session. In that case the receiver cannot reliably treat the text as synchronized conversational media.
This document does not define RTP/T.140 support itself. Jingle RTP Sessions (XEP-0167) [4] already supports RTP media descriptions, and implementations can use RTP/T.140 as specified by RFC 4103 without the rtt-sync element. The rtt-sync element is additional metadata for implementations that need to label the text stream as conversation text, captions, transcripts, translation or interpreter text and expose its synchronization quality to the user.

2 Requirements

This specification is designed to meet the following requirements.

1. Enable a Jingle initiator to offer real-time text in the same session as audio and video.
2. Enable a responder to accept or reject real-time text independently from audio and video.
3. Define a first-class Jingle content for text, for example with content name text or rtt.
4. Allow endpoints to identify the text purpose and source.
5. Allow endpoints to indicate whether the text is synchronized to a media clock, a session clock, the call session only, or not synchronized.
6. Use the existing Jingle Grouping Framework (XEP-0338) [5] grouping framework to bind audio, video and text contents together.

[1] XEP-0301: In-Band Real Time Text <https://xmpp.org/extensions/xep-0301.html>.
[2] XEP-0167: Jingle RTP Sessions <https://xmpp.org/extensions/xep-0167.html>.
[3] XEP-0176: Jingle ICE-UDP Transport Method <https://xmpp.org/extensions/xep-0176.html>.
[4] XEP-0167: Jingle RTP Sessions <https://xmpp.org/extensions/xep-0167.html>.
[5] XEP-0338: Jingle Grouping Framework <https://xmpp.org/extensions/xep-0338.html>.

7. Allow fallback to In-Band Real Time Text (XEP-0301) [6] when synchronized Jingle text is not supported.

8. Prevent clients from silently presenting fallback RTT as synchronized text.

2.1 Implementation levels

Implementations can support different levels without falsely claiming full synchronization.

Level	Name	Minimum capability	User-visible promise
0	XEP-0301 fallback	Ordinary in-band RTT outside Jingle	Live text, not media synchronized
1	Jingle co-session text	Text is negotiated by the same Jingle session but does not share a media clock	Belongs to the call, limited synchronization
2	Session-clock text	Text has timestamps relative to a shared call or session clock	Call-synchronized text
3	Media-clock text	RTP/T.140 or equivalent media-clock timing with audio/video correlation	Strict synchronized Total Conversation

An implementation MUST NOT advertise a higher level than it can actually deliver. In particular, a WebRTC data channel that is merely opened during a call is Level 1 unless it can demonstrate a shared session clock or media clock.

3 Glossary

RTT Real-Time Text, transmitted while it is being typed or created.

Total Conversation A conversation containing simultaneous audio, video and real-time text.

Jingle content A named component inside a Jingle session, such as audio, video or text.

Conversation group A set of Jingle contents intended to be presented as one synchronized conversational unit.

[6] XEP-0301: In-Band Real Time Text <https://xmpp.org/extensions/xep-0301.html>.

4 Use Cases

4.1 Offering Total Conversation

An initiator offers audio, video and text contents in one Jingle session. The receiver accepts all three contents and presents them as a single Total Conversation.

Listing 1: Total Conversation session overview

```
Jingle session sid = abc123
  content audio -> RTP audio
  content video -> RTP video or signing
  content text  -> RTP T.140 or WebRTC datachannel T.140
```

4.2 Adding text during a call

A participant starts an audio-video call and later adds captions, ASR or typed text by sending a Jingle content-add action for the text content.

4.3 Fallback to XEP-0301

If the peer does not support this specification, a client can fall back to In-Band Real Time Text (XEP-0301) [7]. The fallback MUST be visible to the user when synchronized text is required.

5 Protocol Overview

A Total Conversation call SHOULD contain three Jingle contents:

Listing 2: Jingle contents for Total Conversation

```
<content name='audio'> ... </content>
<content name='video'> ... </content>
<content name='text'>  ... </content>
```

The text content is not an ordinary XMPP message stream. It is part of the Jingle session and is described by this extension.
The binding key is the Jingle sid plus the content name. When audio, video and text contents need to be presented as one conversation, the sender SHOULD group those contents using Jingle Grouping Framework (XEP-0338) [8]. A client MUST NOT infer synchronization only from the peer JID, because a user can have multiple simultaneous sessions, devices or fallback chat streams with the same peer.

[7] XEP-0301: In-Band Real Time Text <https://xmpp.org/extensions/xep-0301.html>.
[8] XEP-0338: Jingle Grouping Framework <https://xmpp.org/extensions/xep-0338.html>.

6 Discovery

An entity supporting this specification MUST advertise the following feature:

Listing 3: Primary discovery feature
```
<feature var='urn:xmpp:jingle:apps:rtt-sync:0'/>
```

If the entity supports RTP/T.140, it SHOULD advertise:

Listing 4: RTP/T.140 discovery feature
```
<feature var='urn:xmpp:jingle:apps:rtt-sync:rtp-t140:0'/>
```

If the entity supports WebRTC datachannel T.140, it SHOULD advertise:

Listing 5: Datachannel/T.140 discovery feature
```
<feature var='urn:xmpp:jingle:apps:rtt-sync:dc-t140:0'/>
```

If the entity supports fallback to In-Band Real Time Text (XEP-0301) [9], it SHOULD also advertise the normal XEP-0301 feature.

7 Application Format

This specification defines an rtt-sync element qualified by the urn:xmpp:jingle:apps:rtt-sync:0 namespace.

Attribute	Required	Values	Meaning
role	yes	conversation, caption, transcript, translation, interpreter	Purpose of the text stream
source	no	human, asr, captioner, interpreter, translation, system	Origin of the text
sync-mode	yes	media-clock, session-clock, co-session, none	Synchronization model
max-skew	no	milliseconds	Maximum target presentation difference
finality	no	partial, final, mixed	Whether text can change

[9] XEP-0301: In-Band Real Time Text <https://xmpp.org/extensions/xep-0301.html>.

Listing 6: RTT synchronization element

```
<rtt-sync xmlns='urn:xmpp:jingle:apps:rtt-sync:0'
          role='caption'
          source='asr'
          sync-mode='media-clock'
          max-skew='500'
          finality='partial'/>
```

Language negotiation is intentionally not defined as an rtt-sync attribute. A future Jingle mapping for RFC 8373 could carry language information for text and other media contents consistently.

8 RTP/T.140 Profile

The RTP/T.140 profile is the preferred profile when strict synchronization with audio and video is required. The initiator offers a Jingle RTP content with media='text' and payload types for t140 and optionally red. Implementations that only need RTP/T.140 interoperability MAY use Jingle RTP Sessions (XEP-0167) [10] without the rtt-sync metadata element.

Listing 7: Session initiation with text media

```
<iq from='romeo@example.org/desktop'
    to='juliet@example.org/mobile'
    id='j1'
    type='set'>
  <jingle xmlns='urn:xmpp:jingle:1'
          action='session-initiate'
          initiator='romeo@example.org/desktop'
          sid='abc123'>
    <group xmlns='urn:xmpp:jingle:apps:grouping:0' semantics='LS'>
      <content name='audio'/>
      <content name='video'/>
      <content name='text'/>
    </group>
    <content creator='initiator' name='audio'>
      <description xmlns='urn:xmpp:jingle:apps:rtp:1' media='audio'>
        <payload-type id='111' name='opus' clockrate='48000' channels=
           '2'/>
      </description>
      <transport xmlns='urn:xmpp:jingle:transports:ice-udp:1'/>
    </content>
    <content creator='initiator' name='video'>
```

[10]XEP-0167: Jingle RTP Sessions <https://xmpp.org/extensions/xep-0167.html>.

```
      <description xmlns='urn:xmpp:jingle:apps:rtp:1' media='video'>
        <payload-type id='96' name='VP8' clockrate='90000'/>
      </description>
      <transport xmlns='urn:xmpp:jingle:transports:ice-udp:1'/>
    </content>
    <content creator='initiator' name='text'>
      <description xmlns='urn:xmpp:jingle:apps:rtp:1' media='text'>
        <payload-type id='98' name='t140' clockrate='1000'/>
        <payload-type id='100' name='red' clockrate='1000'>
          <parameter name='fmtp' value='98/98/98'/>
        </payload-type>
        <rtt-sync xmlns='urn:xmpp:jingle:apps:rtt-sync:0'
                  role='conversation'
                  source='human'
                  sync-mode='media-clock'
                  max-skew='500'
                  finality='mixed'/>
      </description>
      <transport xmlns='urn:xmpp:jingle:transports:ice-udp:1'/>
    </content>
  </jingle>
</iq>
```

When sync-mode='media-clock' is negotiated, endpoints SHOULD use the same RTCP CNAME for audio, video and text RTP streams belonging to the same endpoint. Receivers SHOULD use RTP/RTCP timing to align text with audio or video where possible. If timing information is unavailable, the receiver MAY fall back to session arrival time and SHOULD indicate reduced synchronization quality.

9 WebRTC Datachannel/T.140 Profile

The datachannel profile supports browser/WebRTC deployments using T.140 over a reliable, ordered data channel. This profile is useful when a WebRTC implementation naturally uses data channels for RTT. However, data channels do not automatically share the RTP media clock, so the synchronization mode MUST be declared carefully.
RFC 8865 relies on the SDP dcmap and dcsa attributes from RFC 8864. This document does not currently define a complete Jingle mapping for RFC 8864. A future mapping for RFC 8864 should be defined before this document normatively depends on RFC 8865 data channel negotiation in the same way that Jingle RTP Sessions (XEP-0167) [11] already maps RTP/T.140.

- Use sync-mode='co-session' when the text is part of the same call but not strictly media-clock synchronized.

[11] XEP-0167: Jingle RTP Sessions <https://xmpp.org/extensions/xep-0167.html>.

- Use sync-mode='session-clock' when the implementation provides a common session clock.
- Use sync-mode='media-clock' only if the implementation can provide reliable media-clock alignment.

Listing 8: Illustrative datachannel text content

```
<content creator='initiator' name='text'>
  <description xmlns='urn:xmpp:jingle:apps:rtt-sync:0'
               profile='dc-t140'>
    <datachannel subprotocol='t140'
                 reliability='reliable'
                 order='in-order'
                 label='rtt'/>
    <rtt-sync role='conversation'
              source='human'
              sync-mode='co-session'
              max-skew='700'/>
  </description>
  <transport xmlns='urn:xmpp:jingle:transports:dtls-sctp:1'/>
</content>
```

The exact Jingle mapping for WebRTC data channel negotiation should be aligned with the relevant Jingle data channel signalling specification. This document does not attempt to replace that signalling.

10 SIP and SDP Interworking

For SIP gateways and other SDP-based systems, the most useful comparison point is the SDP that the gateway would exchange with the SIP side. The following example is illustrative and intended to guide further review with implementers experienced in RTT over SIP.

Listing 9: Illustrative SDP for RTP/T.140

```
v=0
o=alice 2890844526 2890844526 IN IP4 192.0.2.10
s=Total Conversation
c=IN IP4 192.0.2.10
t=0 0
a=group:LS audio video text
m=audio 49170 RTP/AVP 111
a=mid:audio
a=rtpmap:111 opus/48000/2
m=video 51372 RTP/AVP 96
a=mid:video
a=rtpmap:96 VP8/90000
```

```
m=text 54111 RTP/AVP 98 100
a=mid:text
a=rtpmap:98 t140/1000
a=rtpmap:100 red/1000
a=fmtp:100 98/98/98
```

The a=group line maps naturally to Jingle Grouping Framework (XEP-0338) [12]. The m=text, rtpmap:t140, rtpmap:red and fmtp lines map through the existing Jingle RTP model. If common SIP deployments use additional registered SDP attributes for RTT language, source, purpose or synchronization semantics, those attributes should be mapped directly rather than replaced by XMPP-only attributes.

11 Fallback to XEP-0301

If the responder does not support urn:xmpp:jingle:apps:rtt-sync:0, the initiator MAY fall back to In-Band Real Time Text (XEP-0301) [13]. Fallback MUST be explicit in the user interface when synchronization is required.

Listing 10: Informing the peer about fallback

```
<message from='romeo@example.org/desktop'
         to='juliet@example.org/mobile'
         type='chat'>
  <rtt-fallback xmlns='urn:xmpp:jingle:apps:rtt-sync:0'
                sid='abc123'
                method='xep-0301'
                sync-mode='none'
                reason='peer-unsupported'/>
</message>
```

Fallback is a state transition, not just a transport choice. If a Jingle text content is rejected but audio and video are accepted, the call MAY continue without synchronized text. If fallback RTT is started for the same conversation, it SHOULD be bound to the Jingle sid and shown as fallback rather than synchronized captions.

12 Business Rules

12.1 Sender rules

1. A sender that offers synchronized RTT MUST include an rtt-sync element.

[12] XEP-0338: Jingle Grouping Framework <https://xmpp.org/extensions/xep-0338.html>.
[13] XEP-0301: In-Band Real Time Text <https://xmpp.org/extensions/xep-0301.html>.

2. A sender MUST identify whether the stream is conversation text, caption text, transcript text, interpreter text or translation text.

3. A sender SHOULD include language metadata when a standardized Jingle mapping for the relevant SDP language attributes is available.

4. A sender MUST NOT label ASR text as human captioning.

5. A sender MUST route Jingle text for the negotiated content through the negotiated Jingle transport, not through an unrelated ordinary chat message path.

12.2 Receiver rules

1. A receiver MUST treat a Jingle synchronized RTT content as part of the call, not as normal chat.

2. A receiver SHOULD use the negotiated sync-mode to determine presentation.

3. A receiver MUST bind incoming synchronized text to the Jingle sid and content name before presenting it as part of a call.

4. A receiver SHOULD detect duplicate text received through both Jingle text and XEP-0301 fallback and avoid showing it twice.

5. A receiver SHOULD expose diagnostics when RTT is present in chat but absent from the Jingle session.

13 User Interface Guidance

A user interface SHOULD distinguish at least these cases: live text, live captions, AI captions, human captions, translation and unsynchronized fallback.
During call setup, a client SHOULD expose whether synchronized text was negotiated, whether live text fallback is active or whether text is unavailable in the call.

Listing 11: Example user-visible states

```
Synchronized text: negotiated
Live text fallback: active
Text in call: unavailable
```

14 Accessibility Considerations

This specification is specifically motivated by accessibility and Total Conversation use cases. A deaf or hard-of-hearing user MUST be able to distinguish between typed text, human captions, AI or ASR captions and translated text where this information is known.

A client SHOULD visibly indicate late captions, uncertain ASR captions or unsynchronized fallback text. A client SHOULD allow users to prefer synchronized captions over lowest-latency captions, or lowest-latency captions over strict synchronization.

15 Internationalization Considerations

Text content MUST support Unicode. Language tags SHOULD use BCP 47 when they are available through an applicable media or signalling mapping. Clients SHOULD support multiple simultaneous text streams where translation or interpreter text is provided in addition to original captions.

RFC 8373 defines SDP language negotiation for real-time media. A future Jingle mapping of RFC 8373 would be useful for real-time text and for other media.

16 Security Considerations

Synchronized RTT and captions can contain highly sensitive conversation content. Implementations SHOULD use end-to-end encrypted signalling and encrypted media where available.

For RTP/T.140, implementations SHOULD use SRTP or an equivalent encrypted RTP transport, authenticate the sender of the text stream and protect against injection of false captions. Implementations SHOULD prevent downgrade attacks from synchronized RTT to unsynchronized fallback without user indication.

Clients SHOULD avoid misrepresenting AI captions as human or verified text.

17 Privacy Considerations

Real-time text can reveal text before the sender considers it final. Captions can reveal speech content to captioning, relay or ASR services. A client SHOULD obtain user consent before sending typed RTT and before sending audio to ASR or captioning services.

A client SHOULD not store partial captions or partial RTT as a final transcript unless enabled. A client SHOULD indicate when a third-party captioning, ASR, relay or interpreting service is active.

18 IANA Considerations

This document makes no direct IANA request unless future revisions define new SDP attributes or new media types. The RTP/T.140 profile uses existing text/t140 and text/red media formats.

19 XMPP Registrar Considerations

This specification requests registration of the following namespace:

```
urn:xmpp:jingle:apps:rtt-sync:0
```

The following service discovery features are requested:

```
urn:xmpp:jingle:apps:rtt-sync:0
urn:xmpp:jingle:apps:rtt-sync:rtp-t140:0
urn:xmpp:jingle:apps:rtt-sync:dc-t140:0
```

20 Design Considerations

This document does not replace In-Band Real Time Text (XEP-0301) [14]. XEP-0301 remains appropriate for chat-oriented real-time text and as a fallback. The distinction is that this specification binds text to a Jingle session when an implementation needs Total Conversation semantics.
RTP/T.140 is the preferred strict synchronization profile. WebRTC datachannel T.140 is useful for browser deployments, but MUST NOT be described as media-clock synchronized unless the implementation can provide the required timing relationship.
This document also does not replace Jingle Grouping Framework (XEP-0338) [15]. Grouping of audio, video and text contents SHOULD use the existing Jingle grouping framework instead of inventing RTT-specific grouping attributes.

21 Implementation Experience

An experimental browser implementation has tested the WebRTC datachannel profile at Level 1. Two browser sessions negotiated one Jingle audio-video session plus a text content using urn:xmpp:jingle:apps:rtt-sync:0, opened a reliable ordered data channel labelled rtt, exchanged live RTT updates, and delivered final text bound to the Jingle session. The client presented the call as live text synchronized with the call session.
The same implementation retained In-Band Real Time Text (XEP-0301) [16] fallback for peers that do not negotiate the Jingle text content, so ordinary live text remains available without being presented as synchronized call media.

[14] XEP-0301: In-Band Real Time Text <https://xmpp.org/extensions/xep-0301.html>.
[15] XEP-0338: Jingle Grouping Framework <https://xmpp.org/extensions/xep-0338.html>.
[16] XEP-0301: In-Band Real Time Text <https://xmpp.org/extensions/xep-0301.html>.

22 XML Schema

The following schema is an initial sketch.

```
<xs:schema
    xmlns:xs='http://www.w3.org/2001/XMLSchema'
    targetNamespace='urn:xmpp:jingle:apps:rtt-sync:0'
    xmlns='urn:xmpp:jingle:apps:rtt-sync:0'
    elementFormDefault='qualified'>

  <xs:element name='rtt-sync'>
    <xs:complexType>
      <xs:attribute name='role' use='required'>
        <xs:simpleType>
          <xs:restriction base='xs:NCName'>
            <xs:enumeration value='conversation'/>
            <xs:enumeration value='caption'/>
            <xs:enumeration value='transcript'/>
            <xs:enumeration value='translation'/>
            <xs:enumeration value='interpreter'/>
          </xs:restriction>
        </xs:simpleType>
      </xs:attribute>
      <xs:attribute name='source' use='optional'>
        <xs:simpleType>
          <xs:restriction base='xs:NCName'>
            <xs:enumeration value='human'/>
            <xs:enumeration value='asr'/>
            <xs:enumeration value='captioner'/>
            <xs:enumeration value='interpreter'/>
            <xs:enumeration value='translation'/>
            <xs:enumeration value='system'/>
          </xs:restriction>
        </xs:simpleType>
      </xs:attribute>
      <xs:attribute name='sync-mode' use='required'>
        <xs:simpleType>
          <xs:restriction base='xs:NCName'>
            <xs:enumeration value='media-clock'/>
            <xs:enumeration value='session-clock'/>
            <xs:enumeration value='co-session'/>
            <xs:enumeration value='none'/>
          </xs:restriction>
        </xs:simpleType>
      </xs:attribute>
      <xs:attribute name='max-skew' type='xs:nonNegativeInteger' use='
          optional'/>
      <xs:attribute name='finality' use='optional'>
        <xs:simpleType>
```

```
          <xs:restriction base='xs:NCName'>
            <xs:enumeration value='partial'/>
            <xs:enumeration value='final'/>
            <xs:enumeration value='mixed'/>
          </xs:restriction>
        </xs:simpleType>
      </xs:attribute>
    </xs:complexType>
  </xs:element>
</xs:schema>
```

23 Open Issues

1. Should this be a new Jingle application format or an extension to Jingle RTP Sessions (XEP-0167) [17]?
2. Should RTP/T.140 be mandatory-to-implement for strict synchronization?
3. Should an RFC 8864 Jingle mapping be defined separately before this document normatively specifies RFC 8865 datachannel usage?
4. Should an RFC 8373 Jingle mapping carry language negotiation for RTT and other media?
5. Should emergency-service profiles have stricter requirements?
6. Should multiparty RTT support be included here or deferred to a separate specification?
7. Which SIP/SDP RTT examples should be treated as implementation targets?

[17] XEP-0167: Jingle RTP Sessions <https://xmpp.org/extensions/xep-0167.html>.