



XMPP

XEP-0308: Last Message Correction

Kevin Smith

<mailto:kevin@kismith.co.uk>

<xmpp:kevin@doomsong.co.uk>

2011-11-10

Version 0.1

Status	Type	Short Name
Experimental	Standards Track	NOT_YET_ASSIGNED

This specification defines a method for marking a message as a correction of the last sent message.

Legal

Copyright

This XMPP Extension Protocol is copyright © 1999 - 2011 by the [XMPP Standards Foundation](#) (XSF).

Permissions

Permission is hereby granted, free of charge, to any person obtaining a copy of this specification (the "Specification"), to make use of the Specification without restriction, including without limitation the rights to implement the Specification in a software program, deploy the Specification in a network service, and copy, modify, merge, publish, translate, distribute, sublicense, or sell copies of the Specification, and to permit persons to whom the Specification is furnished to do so, subject to the condition that the foregoing copyright notice and this permission notice shall be included in all copies or substantial portions of the Specification. Unless separate permission is granted, modified works that are redistributed shall not contain misleading information regarding the authors, title, number, or publisher of the Specification, and shall not claim endorsement of the modified works by the authors, any organization or project to which the authors belong, or the XMPP Standards Foundation.

Warranty

NOTE WELL: This Specification is provided on an "AS IS" BASIS, WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, express or implied, including, without limitation, any warranties or conditions of TITLE, NON-INFRINGEMENT, MERCHANTABILITY, or FITNESS FOR A PARTICULAR PURPOSE.

Liability

In no event and under no legal theory, whether in tort (including negligence), contract, or otherwise, unless required by applicable law (such as deliberate and grossly negligent acts) or agreed to in writing, shall the XMPP Standards Foundation or any author of this Specification be liable for damages, including any direct, indirect, special, incidental, or consequential damages of any character arising from, out of, or in connection with the Specification or the implementation, deployment, or other use of the Specification (including but not limited to damages for loss of goodwill, work stoppage, computer failure or malfunction, or any and all other commercial damages or losses), even if the XMPP Standards Foundation or such author has been advised of the possibility of such damages.

Conformance

This XMPP Extension Protocol has been contributed in full conformance with the XSF's Intellectual Property Rights Policy (a copy of which can be found at <http://xmpp.org/about-xmpp/xsf/xsf-ipr-policy/> or obtained by writing to XMPP Standards Foundation, 1899 Wynkoop Street, Suite 600, Denver, CO 80202 USA).

Contents

1	Introduction	1
2	Discovering support	1
3	Use Case	1
4	Business Rules	2
5	Security Considerations	2
6	IANA Considerations	2
7	XMPP Registrar Considerations	3
	7.1 Protocol Namespaces	3
	7.2 Service Discovery Identities	3
8	XML Schema	3

1 Introduction

When sending a message, people often introduce typing errors and send a follow-up message to correct them. This specification allows the sending client to flag the second message as correcting the first.

2 Discovering support

If a server implements message correction, it MUST specify the 'urn:xmpp:message-correct:0' feature in its service discovery information features as specified in [Service Discovery](#)¹ and the Entity Capabilities profile specified in [Entity Capabilities](#)².

Listing 1: Client requests information about a chat partner's client

```
<iq type='get'
  from='romeo@montague.net/orchard'
  id='info1'>
  <query xmlns='http://jabber.org/protocol/disco#info' />
</iq>
```

Listing 2: Partner's client advertises support for correction

```
<iq type='get'
  to='romeo@montague.net/home'
  from='montague.net'
  id='info1'>
  <query xmlns='http://jabber.org/protocol/disco#info'>
  ...
  <feature var='urn:xmpp:message-correct:0' />
  ...
  </query>
</iq>
```

3 Use Case

When a user indicates to the client that he wants to correct the most recently sent message to a contact, the client will resend the corrected message with a new id, and with the replace payload referring to the previous message by id.

Listing 3: User sends a message with a mistake in

¹XEP-0030: Service Discovery <<http://xmpp.org/extensions/xep-0030.html>>.

²XEP-0115: Entity Capabilities <<http://xmpp.org/extensions/xep-0115.html>>.

```
<message to='juliet@capulet.net/balcony' id='bad1'>  
  <body>But soft, what light through yonder airlock breaks?</body>  
</message>
```

Listing 4: User corrects the message and sends

```
<message to='juliet@capulet.net/balcony' id='good1'>  
  <body>But soft, what light through yonder window breaks?</body>  
  <replace id='bad1' xmlns='urn:xmpp:message-correct:0' />  
</message>
```

4 Business Rules

The 'id' attribute is included on the replace to prevent situations where messages being routed to a different resource than the intended cause incorrect replacements.

A receiving client can choose to replace the previous message in whatever display is used for messages, or in any stored history, or can choose to display the correction in another way.

A client SHOULD alert the user that the displayed message has been edited since it was originally sent.

Clients MUST send ids on messages if they allow the user to correct messages.

To deal with multiple payloads, the sender MUST re-send the entire stanza with only the id and the corrections changed. It is expected that the receiver will then treat the new stanza as complete replacement, but such logic is ultimately the responsibility of the client.

The Sender MUST NOT include a correction for a message with non-messaging payloads. For example, a sender MUST NOT include a correction for a roster item exchange request.

5 Security Considerations

The replacement message could have an entirely different meaning from the original message, so clients will need to warn users that the displayed message has been edited.

6 IANA Considerations

None.

7 XMPP Registrar Considerations

7.1 Protocol Namespaces

Upon advancement of this specification, the [XMPP Registrar](#)³ shall issue permanent namespaces in accordance with the process defined in Section 4 of [XMPP Registrar Function](#)⁴. The following namespaces are requested, and are thought to be unique per the XMPP Registrar's requirements:

- urn:xmpp:message-correct:0

7.2 Service Discovery Identities

```
<var>
  <name>urn:xmpp:message-correct:0</name>
  <desc>Support for message correction</desc>
  <doc>THIS PROTOXEP</doc>
</var>
```

8 XML Schema

```
<?xml version='1.0' encoding='UTF-8'?>
<xs:schema
  xmlns:xs='http://www.w3.org/2001/XMLSchema'
  targetNamespace='urn:xmpp:message-correct:0'
  xmlns='urn:xmpp:message-correct:0'
  elementFormDefault='qualified'>
  <xs:element name='replace'>
    <xs:complexType>
      <xs:simpleContent>
        <xs:extension base='xs:string'>
          <xs:attribute name='id' type='xs:string' use='required' />
        </xs:extension>
      </xs:simpleContent>
    </xs:complexType>
  </xs:element>
</xs:schema>
```

³The XMPP Registrar maintains a list of reserved protocol namespaces as well as registries of parameters used in the context of XMPP extension protocols approved by the XMPP Standards Foundation. For further information, see <http://xmpp.org/registrar/>.

⁴XEP-0053: XMPP Registrar Function <http://xmpp.org/extensions/xep-0053.html>.