



# XMPP

## XEP-0439: Quick Response

Tim Henkes

<mailto:me@syndace.dev>

2020-05-05

Version 0.1.0

Status	Type	Short Name
Deferred	Standards Track	NOT_YET_ASSIGNED

Quickly respond to automated messages.

# Legal

## Copyright

This XMPP Extension Protocol is copyright © 1999 – 2020 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 <https://xmpp.org/about/xsf/ipr-policy>) or obtained by writing to XMPP Standards Foundation, P.O. Box 787, Parker, CO 80134 USA).

# Contents

<b>1</b>	<b>Introduction</b>	<b>1</b>
<b>2</b>	<b>Requirements</b>	<b>1</b>
<b>3</b>	<b>Use Cases</b>	<b>1</b>
<b>4</b>	<b>Elements</b>	<b>1</b>
4.1	Response . . . . .	1
4.2	Action . . . . .	2
4.3	Action Selection . . . . .	2
<b>5</b>	<b>Protocol</b>	<b>2</b>
5.1	Sending a Set of Possible Responses . . . . .	2
5.2	Selecting a Response . . . . .	3
5.3	Sending a Set of Available Actions . . . . .	3
5.4	Selecting an Action . . . . .	3
<b>6</b>	<b>Business Rules</b>	<b>4</b>
<b>7</b>	<b>Accessibility Considerations</b>	<b>4</b>
<b>8</b>	<b>Internationalization Considerations</b>	<b>4</b>
<b>9</b>	<b>Security Considerations</b>	<b>5</b>
<b>10</b>	<b>IANA Considerations</b>	<b>5</b>
<b>11</b>	<b>XMPP Registrar Considerations</b>	<b>5</b>
11.1	Protocol Namespaces . . . . .	5
11.2	Protocol Versioning . . . . .	5
<b>12</b>	<b>XML Schema</b>	<b>5</b>

## 1 Introduction

Interactions with bots often require sending one of multiple predefined (plaintext) messages. This specification offers a way for XMPP entities to list possible responses to a message, so that entities that receive such a list can offer convenient UI to quickly respond with one of them. Additionally, this specification provides a way for entities to provide generic actions in similar fashion to quick responses.

## 2 Requirements

- Offer a simple way to list possible responses to a message.
- Offer a solution that doesn't rely on the receiving device to support Quick Response.

## 3 Use Cases

A chat bot wants to provide a list of possible responses to a message it sends.

A chat bot wants to provide quick access to certain actions for convenience.

## 4 Elements

### 4.1 Response

Each possible response is represented by a `<response>` element in the `urn:xmpp:tmp:quick-response` namespace.

```
<response xmlns="urn:xmpp:tmp:quick-response" xml:lang="en" value="yes" label="Yes!" />
```

**value** The value is the internationalized textual payload to put into the `<body>` of the message stanza that is sent when this response is selected.

**label** The label is an optional internationalized textual label for this response. Clients that offer UI for quick selection of one of the possible responses MAY refer to this response by label instead of value. Topic for discussion: are labels required or should UIs just show the value? Are labels maybe even harmful because they could show something totally different than the value?

**xml:lang** The `xml:lang` set on this element MUST mirror the `xml:lang` of the `<body>` included in the message stanza next to the `<response>` element. Refer to the Internationalization Considerations for details. This includes not setting an `xml:lang` at all if not present on the `<body>`.

## 4.2 Action

Each available action is represented by an `<action>` element in the `urn:xmpp:tmp:quick-response` namespace.

```
<action xmlns="urn:xmpp:tmp:quick-response" xml:lang="en" id="merge"
  label="Merge_Now!" />
```

**id** A string identifying the action. When selected, this `id` is sent in an `<action-selected>` element as part of a message stanza without any `<body>` elements.

**label** Internationalized textual label for this action. The `xml:lang` attribute and the language of the label should mirror those of the `<body>` element included in this `<message>`.

**xml:lang** The `xml:lang` set on this element **MUST** mirror the `xml:lang` of the `<body>` included in the message stanza next to the `<action>` element. Refer to the Internationalization Considerations for details. This includes not setting an `xml:lang` at all if not present on the `<body>`.

## 4.3 Action Selection

A selected action is represented by an `<action-selected>` element in the `urn:xmpp:tmp:quick-response` namespace.

```
<action-selected xmlns="urn:xmpp:tmp:quick-response" id="merge" />
```

**id** The `id` of the selected action, as defined in the selected `<action>`.

# 5 Protocol

## 5.1 Sending a Set of Possible Responses

A message with possible responses is sent by including one or more `<response>` elements with distinct values.

```
<message from="rootbot@example.com">
  <body xml:lang="en">Execute 'rm -rf /'? (yes/no)</body>
  <response xmlns="urn:xmpp:tmp:quick-response" xml:lang="en" value="
    yes" label="Sure!" />
  <response xmlns="urn:xmpp:tmp:quick-response" xml:lang="en" value="
    no" label="Uuuuuuh..." />
</message>
```

A single message MUST NOT contain multiple <response> elements with the same values for the value or the label attributes.

Clients that receive a message containing possible responses MAY offer UI to quickly and conveniently select one of the responses. Clients MUST NOT limit the allowed responses to only these responses: the sending entity could accept responses that are not explicitly listed, for example free text responses in addition to a few fixed possibilities.

## 5.2 Selecting a Response

When the user selects a response, their client sends a plaintext message body containing the value as <body> text, also copying the xml:lang of the <response> to the <body>.

```
<message to="rootbot@example.com">
  <body xml:lang="en">no</body>
</message>
```

The sender of the original message, in this example rootbot@example.com, checks incoming messages for a <body> that only contains the value of a <response> and matches in xml:lang to see if a response was selected. In this example, the <body> matches the value of the English translation for the response "No".

## 5.3 Sending a Set of Available Actions

A message with available actions is sent by including one or more <action> elements with distinct ids.

```
<message to="gitbot@example.com">
  <body>New merge request opened by ExampleUser: https://git.example.
    com/example/mrs/3/</body>
  <action xmlns="urn:xmpp:tmp:quick-response" id="merge-32643" label="
    Merge_Now" />
</message>
```

A single message MUST NOT contain multiple <action> elements with the same values for the id or label attributes.

Clients that receive a message containing available actions SHOULD offer UI to select one of the actions.

## 5.4 Selecting an Action

When the user selects an action, their client sends a message containing an <action-selected> element which identifies the selected action. The message does not contain a <body>.

```
<message to="gitbot@example.com">
  <action-selected xmlns="urn:xmpp:tmp:quick-response" id="merge-32643" />
</message>
```

## 6 Business Rules

All message bodies SHOULD always list the (internationalized) possible responses too, so that users of clients that don't support Quick Response can still know what the possible responses are.

Actions SHOULD only be a quicker way to access a feature that could also be accessed using information in the message body. For example, a bot that notifies about a new merge request includes in its notification message body a link to the web interface where manual merging is possible. An action could offer a more convenient way to merge, without taking the route via the web interface. In summary, users of clients that don't support Quick Response SHOULD still have a way to manually trigger the action.

Clients MUST only provide quick responses for the most recently received message that contains text content.

Clients SHOULD provide actions not only for the most recently received message that contains actions, but also for previous messages with actions. Sending clients MUST keep in mind that they have to choose/generate ids for each <action> accordingly, if they need to differentiate between messages.

## 7 Accessibility Considerations

The elements introduced in this specification carry clear semantics that allow clients to implement UI flexibly for their target user group and hardware platform capabilities.

## 8 Internationalization Considerations

While it is generally possible to include multiple <body> elements with different xml:langs in a single message stanza, this is intentionally not supported by this specification. Message stanzas that also contain elements in the urn:xmpp:tmp:quick-response namespace MUST NOT contain more than one <body> element.

## 9 Security Considerations

This specification only adds quicker/more convenient access to features that are accessible anyway.

## 10 IANA Considerations

This document requires no interaction with the Internet Assigned Numbers Authority (IANA).

## 11 XMPP Registrar Considerations

### 11.1 Protocol Namespaces

This specification defines the following XMPP namespaces:

- urn:xmpp:tmp:quick-response

### 11.2 Protocol Versioning

If the protocol defined in this specification undergoes a revision that is not fully backwards-compatible with an older version, the XMPP Registrar shall increment the protocol version number found at the end of the XML namespaces defined herein, as described in Section 4 of XEP-0053.

## 12 XML Schema

TODO
------