XEP-0147: XMPP URI Scheme Query Components

Peter Saint-Andre
mailto:stpeter@stpeter.im
xmpp:stpeter@jabber.org
https://stpeter.im/

2006-09-13
Version 1.2

Status	Type	Short Name
Active	Informational	querytypes

This document defines a registry of query components to be used in the context of XMPP IRIs/URIs and also specifies an initial submission of values to that registry.

Legal

Copyright

This XMPP Extension Protocol is copyright © 1999 – 2024 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

1 Introduction **1**

2 Terminology **1**

3 Query Actions **2**

3.1 Message Action 2

3.2 Roster Management Actions 3

3.2.1 Add Roster Item 4

3.2.2 Remove Roster Item 5

3.3 Subscription Management Actions 5

3.3.1 Subscribe 5

3.3.2 Unsubscribe 6

4 Internationalization Considerations **6**

5 Security Considerations **6**

6 IANA Considerations **6**

7 XMPP Registrar Considerations **7**

7.1 XMPP IRI/URI Querytype Registry 7

7.1.1 Registration Process 7

7.1.2 Registration 7

1 Introduction

RFC 5122 [1] defines a Uniform Resource Identifier (URI) scheme for use in forming URIs and Internationalized Resource Identifiers (IRIs) [2] from the addresses of entities that can communicate using the Extensible Messaging and Presence Protocol (see XMPP Core [3]). Such identifiers enable identification of and interaction with XMPP entities by non-native applications such as databases and web browsers.
However, RFC 5122 intentionally leaves the potential values of the query component open-ended, does not provide a list of common "actions" for queries (e.g., send message or join chatroom), and does not specify recommended "key-value" pairs to be used in the context of such actions. Therefore, this document defines a registry of such actions and key-value pairs (to be maintained by the XMPP Registrar [4]) and specifies a set of initial values for that registry.
This document is organized as follows:

- The registry is defined in the XMPP IRI/URI Querytype Registry section.
- An initial set of actions and key-value pairs is described in the Query Actions section.
- A corresponding initial submission to the registry is specified in the Initial Registration section.
- The process for submitting additional values to the registry is specified in the Registration Process section.

Note: The format of the XMPP URI scheme, including the format of the query component, is fully specified and formally defined in RFC 5122; this document does not modify the xmpp URI scheme in any way and assumes that the reader is familiar with all aspects of RFC 5122.

2 Terminology

This document inherits terminology from RFC 3986 [5], RFC 3987 [6], and RFC 5122.

[1] RFC 5122: Internationalized Resource Identifiers (IRIs) and Uniform Resource Identifiers (URIs) for the Extensible Messaging and Presence Protocol (XMPP) <http://tools.ietf.org/html/rfc5122>.
[2] On the difference between IRIs and URIs, see RFC 3987.
[3] RFC 6120: Extensible Messaging and Presence Protocol (XMPP): Core <http://tools.ietf.org/html/rfc6120>.
[4] The 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 <https://xmpp.org/registrar/>.
[5] RFC 3986: Uniform Resource Identifiers (URI): Generic Syntax <http://tools.ietf.org/html/rfc3986>.
[6] RFC 3987: Internationalized Resource Identifiers (IRIs) <http://tools.ietf.org/html/rfc3987>.

3 Query Actions

The range of actions that might be triggered by interaction with an XMPP entity by means of an XMPP IRI/URI is potentially as wide as the range of extensions to XMPP. This document does not seek to exhaustively define all such potential actions. However, the following actions might be of general interest:

1. Sending a message.
2. Adding or removing a roster item.
3. Subscribing to or unsubscribing from an entity's presence information.
4. Determining Service Discovery (XEP-0030) [7] information about an entity.
5. Joining a groupchat room (see Multi-User Chat (XEP-0045) [8]).
6. Requesting initiation of Ad-Hoc Commands (XEP-0050) [9] associated with an entity.
7. Retrieving the vcard-temp (XEP-0054) [10] data associated with an entity.
8. Subscribing to or unsubscribing from a Publish-Subscribe (XEP-0060) [11] node.
9. Registering with another entity via In-Band Registration (XEP-0077) [12].
10. Initiating a file transfer with another entity (see SI File Transfer (XEP-0096) [13]).

For each such action, the XMPP Registrar maintains a RECOMMENDED "querytype" (this can be thought of as an action name or "verb"; see RFC 5122 for syntax and semantics) as well as an OPTIONAL list of keys to be used in key-value pairs if appropriate.
The querytypes and key-value pairs related to RFC 3920 and RFC 3921 are defined herein; the querytypes and key-value pairs related to protocols defined in the XMPP Standards Foundation's XEP series are defined in the relevant XMPP Extension Protocol specifications.

3.1 Message Action

It may desirable for an XMPP IRI/URI to trigger a specialized interface for sending an XMPP message stanza. The RECOMMENDED querytype for this action is "message". If no key-value pair is provided, interacting with an XMPP IRI/URI that contains a querytype of "message" SHOULD trigger an interface that enables the user to input the text of an XMPP message and other relevant parameters (e.g., a message subject or XHTML-IM (XEP-0071) [14] markup).

[7] XEP-0030: Service Discovery <https://xmpp.org/extensions/xep-0030.html>.
[8] XEP-0045: Multi-User Chat <https://xmpp.org/extensions/xep-0045.html>.
[9] XEP-0050: Ad-Hoc Commands <https://xmpp.org/extensions/xep-0050.html>.
[10] XEP-0054: vcard-temp <https://xmpp.org/extensions/xep-0054.html>.
[11] XEP-0060: Publish-Subscribe <https://xmpp.org/extensions/xep-0060.html>.
[12] XEP-0077: In-Band Registration <https://xmpp.org/extensions/xep-0077.html>.
[13] XEP-0096: SI File Transfer <https://xmpp.org/extensions/xep-0096.html>.
[14] XEP-0071: XHTML-IM <https://xmpp.org/extensions/xep-0071.html>.

Listing 1: Basic Message Action

```
xmpp:romeo@montague.net?message
```

A query component whose querytype is "message" MAY specify various key-value pairs. The following three keys are associated with the child elements of the <message/> stanza specified in the XML schema for the 'jabber:client' namespace as defined in XMPP IM [15]:

1. subject
2. body
3. thread

In addition, the following three keys are associated with the attributes of the <message/> stanza specified in the XML schema for the 'jabber:client' namespace (the 'to' attribute is unnecessary, since it is provided by the XMPP address included in the IRI/URI):

1. from
2. id
3. type

Other keys MAY be registered with the XMPP Registrar but are not specified herein.

Listing 2: Message Action with Keys: IRI/URI

```
xmpp:romeo@montague.net?message;subject=Test%20Message;body=Here%27s
    %20a%20test%20message
```

Listing 3: Message Action with Keys: Resulting Stanza

```
<message to='romeo@montague.net'>
  <subject>Test Message</subject>
  <body>Here&apos;s a test message.</body>
</message>
```

3.2 Roster Management Actions

The 'jabber:iq:roster' namespace provides a mechanism for managing an XMPP roster (also called a "contact list"). This namespace is defined in RFC 3921. The following actions are defined.

[15] RFC 6121: Extensible Messaging and Presence Protocol (XMPP): Instant Messaging and Presence <http://tools.ietf.org/html/rfc6121>.

3.2.1 Add Roster Item

The registered querytype for adding items to the roster or editing items in the roster is "roster" (effectively there is no difference between adding and editing). An XMPP IRI/URI containing a "roster" querytype MAY also include at most one "name" key whose value maps to the 'name' attribute of the <item/> element within the 'jabber:iq:roster' namespace, and MAY contain one "group" key whose value maps to the character data of the <group/> child element of <item/>.

Listing 4: Roster Action: IRI/URI

```
xmpp:romeo@montague.net?roster
```

Listing 5: Roster Action: Resulting Stanza

```
<iq type='set'>
  <query xmlns='jabber:iq:roster'>
    <item jid='romeo@montague.net'/>
  </query>
</iq>
```

Listing 6: Roster Action With Name: IRI/URI

```
xmpp:romeo@montague.net?roster;name=Romeo%20Montague
```

Listing 7: Roster Action With Name: Resulting Stanza

```
<iq type='set'>
  <query xmlns='jabber:iq:roster'>
    <item jid='romeo@montague.net' name='Romeo␣Montague'/>
  </query>
</iq>
```

Listing 8: Roster Action With Name and Group: IRI/URI

```
xmpp:romeo@montague.net?roster;name=Romeo%20Montague;group=Friends
```

Listing 9: Roster Action With Name and Group: Resulting Stanza

```
<iq type='set'>
  <query xmlns='jabber:iq:roster'>
    <item jid='romeo@montague.net' name='Romeo␣Montague'>
      <group>Friends</group>
    </item>
  </query>
</iq>
```

Note: Methods (if any) for including more than one group are yet to be determined.

3.2.2 Remove Roster Item

The 'jabber:iq:roster' namespace includes a mechanism for removing an XMPP roster item. The registered querytype for removing an item from the roster is "remove".

Listing 10: Roster Removal Action: IRI/URI

```
xmpp:romeo@montague.net?remove
```

Listing 11: Roster Removal Action: Resulting Stanza

```
<iq type='set'>
  <query xmlns='jabber:iq:roster'>
    <item jid='romeo@montague.net' subscription='remove'/>
  </query>
</iq>
```

3.3 Subscription Management Actions

Closely coupled with roster management is presence subscription management. In XMPP, subscription management is handled via special values of the <presence/> stanza, as described in RFC 3921. The following actions are defined

3.3.1 Subscribe

When an entity subscribes to another entity's presence by means of an XMPP IRI/URI, the invoked XMPP application SHOULD first send a roster add stanza as shown below (this is consistent with the recommendations in RFC 3921).

Listing 12: Subscribe Action: IRI/URI

```
xmpp:romeo@montague.net?subscribe
```

Listing 13: Subscribe Action: Resulting Stanzas

```
<iq type='set'>
  <query xmlns='jabber:iq:roster'>
    <item jid='romeo@montague.net'/>
  </query>
</iq>

<presence to='romeo@montague.net' type='subscribe'/>
```

3.3.2 Unsubscribe

XMPP includes a mechanism for unsubscribing from an entity. The registered querytype for doing so is "unsubscribe".

Listing 14: Unsubscribe Action: IRI/URI

```
xmpp:romeo@montague.net?unsubscribe
```

Listing 15: Unsubscribe Action: Resulting Stanza

```
<presence to='romeo@montague.net' type='unsubscribe'/>
```

4 Internationalization Considerations

Internationalization considerations for XMPP IRIs/URIs are specified in RFC 5122; the reader is referred to that document for a complete discussion of the relevant issues.
Localization of application-specific data presented to a human user (e.g., as encapsulated in key-value pairs) is the responsibility of the "using protocol".

5 Security Considerations

Security considerations for XMPP IRIs/URIs are specified in RFC 5122.
Completion of some of the actions defined herein will cause changes to an entity's account, subscriptions to information, registration with services, communication with other entities, completion of ad-hoc commands, and the like. Naturally, such changes, information, services, and communications are potentially undesirable (e.g., joining a chatroom whose discussion topic is not of interest to, or even patently offensive to, the joining user). The invoked application SHOULD appropriately warn a human user regarding the potential consequences of the action about to be completed.

6 IANA Considerations

This document requires no interaction with the Internet Assigned Numbers Authority (IANA) [16]. If in the future the IANA should wish to maintain a registry of XMPP URI/IRI query components, the XMPP Registrar will cooperate with efforts to migrate the registry from the XMPP Registrar to the IANA.

[16] The Internet Assigned Numbers Authority (IANA) is the central coordinator for the assignment of unique parameter values for Internet protocols, such as port numbers and URI schemes. For further information, see <http://www.iana.org/>.

7 XMPP Registrar Considerations

7.1 XMPP IRI/URI Querytype Registry

The XMPP Registrar maintains a registry of querytype values (see <https://xmpp.org/registrar/querytypes.html>).

7.1.1 Registration Process

In order to submit new values to this registry, the registrant shall define an XML fragment of the following form and either include it in the relevant XMPP Extension Protocol or send it to the email address <registrar@xmpp.org>:

```
<querytype>
  <name>the name of the querytype (e.g., "pubsub")</name>
  <proto>
    the namespace of associated protocol output
    (e.g., "http://jabber.org/protocol/pubsub")
  </proto>
  <desc>a natural-language description of the querytype</desc>
  <doc>the document in which the querytype is specified</doc>
  <keys>
    <key>
      <name>the name of the key (e.g., "action")</name>
      <desc>a natural-language description of the key</desc>
      <values>
        <value>
          <name>the name of a value (e.g., "subscribe")</name>
          <desc>a natural-language description of the value</desc>
        </value>
      </values>
    </key>
  </keys>
</querytype>
```

Note: Within the <querytype/> element, the <keys/> child element is OPTIONAL; within any given <key/> element, the <values/> child element is also OPTIONAL.
The registrant may register more than one querytype at a time, each contained in a separate <querytype/> element.

7.1.2 Registration

The following submission registers parameters related to the XMPP RFCs. For submissions related to XMPP extensions, refer to the relevant XMPP Extension Protocol specifications.

```
<querytype>
  <name>message</name>
  <proto>jabber:client</proto>
  <desc>enables sending of an XMPP message stanza</desc>
  <doc>XEP-0147</doc>
  <keys>
    <key>
      <name>subject</name>
      <desc>a subject for the message per the "jabber:client" schema</
          desc>
    </key>
    <key>
      <name>body</name>
      <desc>a body for the message per the "jabber:client" schema</
          desc>
    </key>
    <key>
      <name>thread</name>
      <desc>a Thread ID for the message per the "jabber:client" schema
          </desc>
    </key>
    <key>
      <name>from</name>
      <desc>a from address for the message per the "jabber:client"
          schema</desc>
    </key>
    <key>
      <name>id</name>
      <desc>an ID for the message per the "jabber:client" schema</desc
          >
    </key>
    <key>
      <name>type</name>
      <desc>the message type per the "jabber:client" schema</desc>
      <values>
        <value>
          <name>chat</name>
          <desc>a message of type "chat"</desc>
        </value>
        <value>
          <name>groupchat</name>
          <desc>a message of type "groupchat"</desc>
        </value>
        <value>
          <name>headline</name>
          <desc>a message of type "headline"</desc>
        </value>
        <value>
          <name>normal</name>
```

```
          <desc>a message of type "normal"</desc>
        </value>
      </values>
    </key>
  </keys>
</querytype>

<querytype>
  <name>remove</name>
  <proto>jabber:iq:roster</proto>
  <desc>enables removing a roster item</desc>
  <doc>XEP-0147</doc>
</querytype>

<querytype>
  <name>roster</name>
  <proto>jabber:iq:roster</proto>
  <desc>enables adding or editing a roster item</desc>
  <doc>XEP-0147</doc>
  <keys>
    <key>
      <name>group</name>
      <desc>the user-assigned group for the roster item</desc>
    </key>
    <key>
      <name>name</name>
      <desc>the user-assigned name for the roster item</desc>
    </key>
  </keys>
</querytype>

<querytype>
  <name>subscribe</name>
  <proto>jabber:client</proto>
  <desc>enables sending a presence subscription request</desc>
  <doc>XEP-0147</doc>
</querytype>

<querytype>
  <name>unsubscribe</name>
  <proto>jabber:client</proto>
  <desc>enables unsubscribing from an entity's presence</desc>
  <doc>XEP-0147</doc>
</querytype>
```