



XMPP

XEP-0048: Bookmarks

Rachel Blackman
<mailto:rcb@ceruleanstudios.com>
<xmpp:sparks@jabber.org>

Peter Millard

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

2020-08-04
Version 1.2

Status	Type	Short Name
Deprecated	Standards Track	bookmarks

This specification defines an XML data format for use by XMPP clients in storing bookmarks to multi-user chatrooms and web pages. The chatroom bookmarking function includes the ability to auto-join rooms on login.

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	Data Format	1
2.1	The conference element	1
2.2	The url element	3
3	Storage	4
3.1	Uploading Data	4
3.2	Receiving Notifications	5
3.3	Retrieving Data	6
4	Security Considerations	6
5	IANA Considerations	7
6	XMPP Registrar Considerations	7
7	XML Schema	7
8	Author Note	8

1 Introduction

For ease-of-use in a Jabber client, it is desirable to have a way to store shortcuts to various services and resources (such as conference rooms and web pages) as "bookmarks" that can be displayed in the user's client. Several Jabber clients have already agreed on and implemented a method to provide this service; that informal agreement is documented and expanded upon in this document. In particular, we introduce the `<storage/>` element (qualified by the 'storage:bookmarks' namespace) as a container for this sort of this data. While bookmarks data can be stored using any XML storage mechanism, this document recommends one method that is specific to XMPP.

2 Data Format

A storage element qualified by the 'storage:bookmarks' namespace may contain a collection of child elements, each representing a bookmark to be displayed in a client. At present, only two sub-elements are defined: `<conference/>` for bookmarking of [Multi-User Chat \(XEP-0045\)](#)¹ rooms and `<url/>` for bookmarking of web pages.

All child elements allow a 'name' attribute, which is the friendly name by which they will be displayed in the client. If an element lacks a 'name' attribute, the client SHOULD generate an appropriate substitution based on the other available data.

2.1 The conference element

A common use case is bookmarking of multi-user chat rooms. A room is bookmarked using the `<conference/>` child element. The syntax is as follows.

¹XEP-0045: Multi-User Chat <https://xmpp.org/extensions/xep-0045.html>.

Element or Attribute	Definition	Datatype	Inclusion
'autojoin' attribute	Whether the client should automatically join the conference room on login.	boolean defaulting to false In accordance with Section 3.2.2.1 of XML Schema Part 2: Datatypes, the allowable lexical representations for the xs:boolean datatype are the strings "0" and "false" for the concept 'false' and the strings "1" and "true" for the concept 'true'; implementations MUST support both styles of lexical representation.	OPTIONAL
'jid' attribute	The JabberID of the chat room.	string	REQUIRED
'name' attribute	A friendly name for the bookmark.	string	RECOMMENDED
<nick/> element	The user's preferred roomnick for the chatroom.	string	OPTIONAL
<password/> element	Unencrypted string for the password needed to enter a password-protected room. For security reasons, use of this element is NOT RECOMMENDED.	string	NOT RECOMMENDED

Note: The datatypes are as defined in [XML Schema Part 2](#)².

Listing 1: An example of the conference element

```
<storage xmlns='storage:bookmarks'>
```

²XML Schema Part 2: Datatypes <<http://www.w3.org/TR/xmlschema11-2/>>.

```

<conference name='Council_of_Oberon'
            autojoin='true'
            jid='council@conference.underhill.org'>
  <nick>Puck</nick>
</conference>
</storage>

```

This bookmark would be displayed as 'Council of Oberon' and, if activated, would attempt to join the conference room 'council@conference.underhill.org' with nickname 'Puck'.

Note: A bookmark set may contain any number of conference rooms.

2.2 The url element

The <url/> element is designed for bookmarking web pages, i.e., HTTP or HTTPS URLs. The syntax is as follows.

Attribute	Definition	Datatype	Inclusion
'name' attribute	A friendly name for the bookmark.	string	RECOMMENDED
'url' attribute	The HTTP or HTTPS URL of the web page.	anyURI	REQUIRED

When the user chooses a URL bookmark, the client should launch an appropriate browser or load the URL directly in the client (if the client is a web-client or includes web browsing functionality).

Listing 2: An example of the url element

```

<storage xmlns='storage:bookmarks'>
  <url name='Complete_Works_of_Shakespeare'
      url='http://the-tech.mit.edu/Shakespeare/' />
</storage>

```

This bookmark would be displayed in the client as 'Complete Works of Shakespeare' and would take the user to <http://the-tech.mit.edu/Shakespeare/> when selected.

Note: A bookmark set can contain any number of URLs.

3 Storage

It is RECOMMENDED to use [Publish-Subscribe \(XEP-0060\)](https://xmpp.org/extensions/xep-0060.html)³ for data storage, specifically through the use of personal data nodes hosted at the user's virtual publish-subscribe service as described in [Best Practices for Persistent Storage of Private Data via Publish-Subscribe \(XEP-0223\)](https://xmpp.org/extensions/xep-0223.html)⁴ and illustrated in the following sections.

Note: In the past, [Private XML Storage \(XEP-0049\)](https://xmpp.org/extensions/xep-0049.html)⁵ was the recommended method (see the archived version of this specification at <http://www.xmpp.org/extensions/attic/xep-0048-1.0.html>). In addition, other methods could be used, such as HTTP or WebDAV.

3.1 Uploading Data

Listing 3: Client uploads data

```
<iq from='juliet@capulet.lit/balcony' type='set' id='pip1'>
  <pubsub xmlns='http://jabber.org/protocol/pubsub'>
    <publish node='storage:bookmarks'>
      <item id='current'>
        <storage xmlns='storage:bookmarks'>
          <conference name='The_Play&apos;s_the_Thing'
            autojoin='true'
            jid='theplay@conference.shakespeare.lit'>
            <nick>JC</nick>
          </conference>
        </storage>
      </item>
    </publish>
    <publish-options>
      <x xmlns='jabber:x:data' type='submit'>
        <field var='FORM_TYPE' type='hidden'>
          <value>http://jabber.org/protocol/pubsub#publish-options</value>
        </field>
        <field var='pubsub#persist_items'>
          <value>>true</value>
        </field>
        <field var='pubsub#access_model'>
          <value>whitelist</value>
        </field>
      </x>
    </publish-options>
  </pubsub>
</iq>
```

³XEP-0060: Publish-Subscribe <https://xmpp.org/extensions/xep-0060.html>.

⁴XEP-0223: Best Practices for Persistent Storage of Private Data via Publish-Subscribe <https://xmpp.org/extensions/xep-0223.html>.

⁵XEP-0049: Private XML Storage <https://xmpp.org/extensions/xep-0049.html>.

Listing 4: Server acknowledges successful storage

```
<iq to='juliet@capulet.lit/balcony' type='result' id='pip1' />
```

3.2 Receiving Notifications

The stored data is automatically pushed to all of the user's connected resources.

Listing 5: Publisher receives event notification

```
<message from='juliet@capulet.lit'
  to='juliet@capulet.lit/balcony'
  type='headline'
  id='rnfoo1'>
  <event xmlns='http://jabber.org/protocol/pubsub#event'>
    <items node='storage:bookmarks'>
      <item id='current'>
        <storage xmlns='storage:bookmarks'>
          <conference name='The_Play&apos;s_the_Thing'
            autojoin='true'
            jid='theplay@conference.shakespeare.lit'>
            <nick>JC</nick>
          </conference>
        </storage>
      </item>
    </items>
  </event>
</message>

<message from='juliet@capulet.lit'
  to='juliet@capulet.lit/chamber'
  type='headline'
  id='rnfoo2'>
  <event xmlns='http://jabber.org/protocol/pubsub#event'>
    <items node='storage:bookmarks'>
      <item id='current'>
        <storage xmlns='storage:bookmarks'>
          <conference name='The_Play&apos;s_the_Thing'
            autojoin='true'
            jid='theplay@conference.shakespeare.lit'>
            <nick>JC</nick>
          </conference>
        </storage>
      </item>
    </items>
  </event>
</message>
```


3.3 Retrieving Data

In order to retrieve stored data without receiving notifications (e.g., upon initial login), the user's client sends a retrieve-items request as specified in [Publish-Subscribe \(XEP-0060\)](#)⁶.

Listing 6: Client requests all items

```
<iq from='juliet@capulet.lit/randomID' type='get' id='retrieve1'>
  <pubsub xmlns='http://jabber.org/protocol/pubsub'>
    <items node='storage:bookmarks' />
  </pubsub>
</iq>
```

Listing 7: Server returns all items

```
<iq type='result'
  to='juliet@capulet.lit/randomID'
  id='retrieve1'>
  <pubsub xmlns='http://jabber.org/protocol/pubsub'>
    <items node='storage:bookmarks'>
      <item id='current'>
        <storage xmlns='storage:bookmarks'>
          <conference name='The_Play's_the_Thing'
            autojoin='true'
            jid='theplay@conference.shakespeare.lit'>
            <nick>JC</nick>
          </conference>
        </storage>
      </item>
    </items>
  </pubsub>
</iq>
```

4 Security Considerations

Security considerations related to object persistent via publish-subscribe are described in XEP-0060 and [Best Practices for Persistent Storage of Private Data via Publish-Subscribe \(XEP-0223\)](#)⁷.

Use of the `<password/>` child of the `<conference/>` element is NOT RECOMMENDED, since the password could be discovered by a third party, e.g. an eavesdropper (if channel encryption is not used) or a server administrator. However, the element MAY be used in suitably secure environments (e.g., where it is known that communications will not be sent over unencrypted channels and the server administrators are trusted). Clients SHOULD NOT default to storing

⁶XEP-0060: Publish-Subscribe <<https://xmpp.org/extensions/xep-0060.html>>.

⁷XEP-0223: Best Practices for Persistent Storage of Private Data via Publish-Subscribe <<https://xmpp.org/extensions/xep-0223.html>>.

passwords and MUST enable users to disable any password storage.

5 IANA Considerations

This document requires no interaction with the [Internet Assigned Numbers Authority \(IANA\)](#)⁸.

6 XMPP Registrar Considerations

No action by the [XMPP Registrar](#)⁹ is required, since the 'storage:bookmarks' namespace is already included in the protocol namespaces registry (see <https://xmpp.org/registrar/namespaces.html>).

7 XML Schema

```
<?xml version='1.0' encoding='UTF-8'?>
<xs:schema
  xmlns:xs='http://www.w3.org/2001/XMLSchema'
  targetNamespace='storage:bookmarks'
  xmlns='storage:bookmarks'
  elementFormDefault='qualified'>
  <xs:annotation>
    <xs:documentation>
      The protocol documented by this schema is defined in
      XEP-0048: http://www.xmpp.org/extensions/xep-0048.html
    </xs:documentation>
  </xs:annotation>
  <xs:element name='storage'>
    <xs:complexType>
      <xs:choice>
        <xs:element ref='conference' />
        <xs:element ref='url' />
      </xs:choice>
    </xs:complexType>
  </xs:element>
</xs:schema>
```

⁸The 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/>.

⁹The 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/>.

```
</xs:complexType>
</xs:element>

<xs:element name='conference'>
  <xs:complexType>
    <xs:sequence>
      <xs:element name='nick' type='xs:string' minOccurs='0' />
      <xs:element name='password' type='xs:string' minOccurs='0' />
    </xs:sequence>
    <xs:attribute name='autojoin' type='xs:boolean' use='optional'
      default='false' />
    <xs:attribute name='jid' type='xs:string' use='required' />
    <xs:attribute name='name' type='xs:string' use='required' />
  </xs:complexType>
</xs:element>

<xs:element name='url'>
  <xs:complexType>
    <xs:simpleContent>
      <xs:extension base='empty'>
        <xs:attribute name='name' type='xs:string' use='required' />
        <xs:attribute name='url' type='xs:string' use='required' />
      </xs:extension>
    </xs:simpleContent>
  </xs:complexType>
</xs:element>

<xs:simpleType name='empty'>
  <xs:restriction base='xs:string'>
    <xs:enumeration value='' />
  </xs:restriction>
</xs:simpleType>

</xs:schema>
```

8 Author Note

Peter Millard, a co-author of this specification from version 0.1 through version 1.0, died on April 26, 2006.