XEP-0123: Entity Metadata

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

2003-12-16
Version 0.3

Status	Type	Short Name
Retracted	Standards Track	N/A

NOTE: This proposal was retracted by the author on 2004-02-19.

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 Background and Requirements **1**

2 Protocol **1**

3 Use Cases **2**

3.1 Discovering Support 2

3.2 Requesting Metadata About Another Entity 3

4 Integration with Directory Services **3**

5 Security Considerations **4**

6 IANA Considerations **4**

7 XMPP Registrar Considerations **4**

7.1 Service Discovery Nodes 4

1 Background and Requirements

Traditionally, the only mechanism for communicating detailed information about entities on the Jabber network has been an XML version of the vCard format for electronic business cards (see vcard-temp (XEP-0054) [1]). Unfortunately, the vCard format has several major drawbacks:

1. It is mainly limited to data about persons (although it has been used on the Jabber network to describe things like servers).
2. The format contains relatively few data fields.
3. The format is not extensible.
4. As implemented, the data is not searchable.
5. As implemented, the data cannot be filtered depending on the identity of the requestor.

It is becoming increasingly important to define a robust, extensible format for describing entities on the Jabber network. Such a format should be:

1. Applicable not just to people but to any entity on the network, including but not limited to servers, components, bots, Multi-User Chat (XEP-0045) [2] rooms, Publish-Subscribe (XEP-0060) [3] nodes, and in general anything that can be addressed as a Jabber ID (as defined in XMPP Core [4]).
2. Usable in encapsulating any information about the entity itself (name, address, description, title, etc.).
3. Extensible enough to handle any metadata that may be needed for current and future applications (including, at a minimum, everything that can be defined in vCard); it must be possible to use it for public protocols defined by the IETF or XMPP Standards Foundation as well as for custom or private protocols.
4. Well-defined enough, through datatyping and public registries where applicable, to enable robust searching and filtering based on defined data fields and their values.

2 Protocol

Information about entities is provided using the Infobits (XEP-0120) [5] protocol and registered infobit keynames (mainly those specified in vCard Infobits Mapping (XEP-0125) [6] although

[1] XEP-0054: vcard-temp <https://xmpp.org/extensions/xep-0054.html>.
[2] XEP-0045: Multi-User Chat <https://xmpp.org/extensions/xep-0045.html>.
[3] XEP-0060: Publish-Subscribe <https://xmpp.org/extensions/xep-0060.html>.
[4] RFC 6120: Extensible Messaging and Presence Protocol (XMPP): Core <http://tools.ietf.org/html/rfc6120>.
[5] XEP-0120: Infobits <https://xmpp.org/extensions/xep-0120.html>.
[6] XEP-0125: vCard Infobits Mapping <https://xmpp.org/extensions/xep-0125.html>.

entity metadata is by no means limited to vCard information and could include infobits such as those specified in Dublin Core Infobits Mapping (XEP-0121) [7]). The metadata is discovered by interacting with a common Service Discovery (XEP-0030) [8] node named ”metadata”. The queried entity replies with a service discovery result containing any infobits that the entity wishes to reveal about itself to the requesting entity. This information is always metadata about the entity itself, not any other entities or any relationships that the entity may have to other entities.

3 Use Cases

3.1 Discovering Support

Support for entity metadata is discovered by means of Service Discovery. If the queried entity provides metadata about itself, it SHOULD advertise that fact by listing an item named ”metadata” in response to a disco#items query.

Listing 1: One Entity Queries Another via Disco

```
<iq type='get'
    from='juliet@capulet.com/balcony'
    to='romeo@montague.net/orchard'
    id='disco1'>
  <query xmlns='http://jabber.org/protocol/disco#items'/>
</iq>
```

The entity returns its associated items:

Listing 2: Entity Returns Disco Item Results

```
<iq type='result'
    from='romeo@montague.net/orchard'
    to='juliet@capulet.com/balcony'
    id='disco1'>
  <query xmlns='http://jabber.org/protocol/disco#items'>
    ...
    <item jid='juliet@capulet.com'
          node='metadata'
          name='Information about Juliet Capulet'/>
    ...
  </query>
</iq>
```

[7] XEP-0121: Dublin Core Infobits Mapping <https://xmpp.org/extensions/xep-0121.html>.
[8] XEP-0030: Service Discovery <https://xmpp.org/extensions/xep-0030.html>.

3.2 Requesting Metadata About Another Entity

In order to request the advertised metadata, the requesting entity sends a disco#info request to the 'metadata' node of the JID communicated in the previous result.

Listing 3: Requestor Requests Metadata

```
<iq type='get'
    from='juliet@capulet.com/balcony'
    to='romeo@montague.net'
    id='request1'>
  <query xmlns='http://jabber.org/protocol/disco#info'
         node='metadata'/>
</iq>
```

The entity returns its metadata to the requestor.

Listing 4: Entity Returns Metadata Result

```
<iq type='result'
    from='romeo@montague.net'
    to='juliet@capulet.com/balcony'
    id='request2'>
  <query xmlns='http://jabber.org/protocol/disco#info'
         node='metadata'>
    <info 'http://jabber.org/protocol/infobits'>
      <bit key='fn'>Romeo Montague</bit>
      <bit key='country'>Italy</bit>
      <bit key='city'>Verona</bit>
      <bit key='gender'>male</bit>
      <bit key='nickname'>loverboy</bit>
    </info>
  </query>
</iq>
```

4 Integration with Directory Services

One of the primary motivations behind this proposal is to enable the construction of useful directory services on the Jabber network. Examples of such services include but are not limited to:

- robust user directories
- directories of groupchat rooms
- directories of pubsub nodes

- server directories

Although such directories will be a valuable addition to the network, it is imperative to understand that the canonical source for metadata about an entity is the entity itself. Mechanisms for keeping directories synchronized with entities are outside the scope of this document, and in any case a directory may not be privy to all information about an entity (since in general a user should publish to a directory only the information that he or she deems world-readable). Directories SHOULD require registration using In-Band Registration (XEP-0077) [9]. Before registering with a directory, an entity SHOULD adjust its access controls or privacy rules accordingly, including appropriate definition of classes and addition of the directory server's JID to the relevant privacy rules. Upon accepting registration from an entity, a directory SHOULD immediately send a metadata request to the registering entity. Synchronization of metadata is a matter for the directory implementation to determine, and perhaps negotiate with the registering entity; all such synchronization and negotiation is out of scope for this document.

5 Security Considerations

Metadata MAY be world-readable. Entities MUST take care to ensure that they exercise proper control over access to such information. Users of IM clients SHOULD be warned that their data may be world-readable and be given the option to not publish such information or control it via appropriate mechanisms (such as privacy rules).

6 IANA Considerations

This document requires no interaction with the Internet Assigned Numbers Authority (IANA) [10].

7 XMPP Registrar Considerations

7.1 Service Discovery Nodes

Upon advancement of this proposal to a status of Draft, the XMPP Registrar [11] shall add the 'metadata' node to its registry of common Service Discovery nodes.

[9] XEP-0077: In-Band Registration <https://xmpp.org/extensions/xep-0077.html>.

[10] 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/>.

[11] 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/>.