



XMPP

XEP-0462: PubSub Type Filtering

Timothée Jaussoin
<mailto:edhelas@movim.eu>
<xmpp:edhelas@movim.eu>

Maxime Buquet
<mailto:pep@bouah.net>
<xmpp:pep@bouah.net>

2022-04-22
Version 0.1.1

Status	Type	Short Name
Experimental	Standards Track	NOT_YET_ASSIGNED

This specification provides a way to filter PubSub nodes in a disco query.

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

1	Introduction	1
2	Requirements	1
3	Use Cases	1
3.1	Discovering support	1
3.2	Sending a disco request	1
4	IANA Considerations	2
5	XMPP Registrar Considerations	2
6	XML Schema	3
6.1	urn:xmpp:pubsub-filter:0	3

1 Introduction

Implementations have been able to declare a `pubsub#type` attribute on PubSub nodes for about as long as [Publish-Subscribe \(XEP-0060\)](#)¹ has existed. This attribute doesn't seem to be widely used in the community though, maybe due to the vagueness of its description, that has recently changed, or the lack of features associated with it.

This specification provides a way for implementations to allow filtering on this attribute when discovering nodes on a PubSub service.

Filtering is particularly useful for example combined with [Microblogging Over XMPP \(XEP-0277\)](#)² and comment nodes that are created on the same service. When listing content nodes of a service, one may want to filter out comment nodes.

2 Requirements

- Allow querying only a subset of nodes in a disco items request, in the form of include/exclude

3 Use Cases

3.1 Discovering support

A service implementing this specification **MUST** advertize through [Service Discovery \(XEP-0030\)](#)³ a `urn:xmpp:pubsub-filter:0` feature.

3.2 Sending a disco request

While requesting `disco#items` on a PubSub service, an entity might want to only get nodes of certain `pubsub#type`. To do so, it may add a filter child of namespace `urn:xmpp:pubsub-filter:0` to the query element, containing a [Data Forms \(XEP-0004\)](#)⁴ form with `FORM_TYPE` set to `urn:xmpp:pubsub-filter:0` and an `included-types` or `excluded-types` list-multi type field containing the various types it wants to filter.

When `included-types` is specified, a PubSub service **MUST** return nodes of matching `pubsub#type` in its response.

When `excluded-types` is specified, a PubSub service **MUST** return every node but those of matching `pubsub#types` in its response.

Both `included` and `excluded` fields **MAY** contain an empty value to designate nodes with an empty `pubsub#type`.

¹XEP-0060: Publish-Subscribe <<https://xmpp.org/extensions/xep-0060.html>>.

²XEP-0277: Microblogging over XMPP <<https://xmpp.org/extensions/xep-0277.html>>.

³XEP-0030: Service Discovery <<https://xmpp.org/extensions/xep-0030.html>>.

⁴XEP-0004: Data Forms <<https://xmpp.org/extensions/xep-0004.html>>.

Listing 1: Requesting disco#items with only nodes of the following types, including empty ones

```
<iq type='get'
  from='rosa@com.int/desktop'
  to='news.common.social'
  id='disco1'>
  <query xmlns='http://jabber.org/protocol/disco#items'>
    <filter xmlns='urn:xmpp:pubsub-filter:0'>
      <x xmlns='jabber:x:data' type='submit'>
        <field var='FORM_TYPE' type='hidden'>
          <value>urn:xmpp:pubsub-filter:0</value>
        </field>
        <field type='list-multi' var='included-types'>
          <value>urn:xmpp:microblog:0</value>
          <value/>
        </field>
      </x>
    </filter>
  </query>
</iq>
```

If both the included and excluded fields are specified, a service MUST return an error of type modify containing a bad-request element in the urn:ietf:params:xml:ns:xmpp-stanzas namespace.

Listing 2: Error returned when a requesting entity includes both fields

```
<iq type='error'
  from='news.com.int'
  to='peter@commons.social/desktop'
  id='error1'>
  <error type='modify'>
    <bad-request xmlns='urn:ietf:params:xml:ns:xmpp-stanzas' />
  </error>
</iq>
```

4 IANA Considerations

None.

5 XMPP Registrar Considerations

None.

6 XML Schema

6.1 urn:xmpp:pubsub-filter:0

```
<xs:schema
  xmlns:xs='http://www.w3.org/2001/XMLSchema'
  xmlns:xdata='jabber:x:data'
  targetNamespace='urn:xmpp:pubsub-filter:0'
  xmlns='urn:xmpp:pubsub-filter:0'
  elementFormDefault='qualified'>

  <xs:annotation>
    <xs:documentation>
      The protocol documented by this schema is defined in
      XEP-XXXX: http://xmpp.org/extensions/xep-xxxx.html
    </xs:documentation>
  </xs:annotation>

  <xs:element name='filter'>
    <xs:complexType>
      <xs:choice xmlns:xdata='jabber:x:data'>
        <xs:element ref='xdata:x'/>
      </xs:choice>
    </xs:complexType>
  </xs:element>
</xs:schema>
```