



# XMPP

## XEP-0472: Pubsub Social Feed

Timothée Jaussoin

<mailto:edhelas@movim.eu>

<xmpp:edhelas@movim.eu>

<https://edhelas.movim.eu>

2023-04-03

Version 0.2.0

Status	Type	Short Name
Experimental	Standards Track	psf

This specification defines a way of publishing social content over XMPP.

# 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

<b>1</b>	<b>Introduction</b>	<b>1</b>
<b>2</b>	<b>Glossary</b>	<b>1</b>
<b>3</b>	<b>Discovery</b>	<b>1</b>
<b>4</b>	<b>Protocol</b>	<b>2</b>
4.1	Location . . . . .	2
4.2	Publishing a Post . . . . .	2
4.2.1	Publishing a Post with Rich Content . . . . .	3
4.3	Receiving a Post . . . . .	3
4.4	Replying to a Post . . . . .	4
4.5	Repeating a Post . . . . .	6
4.6	Atom and XMPP integration . . . . .	7
4.6.1	Pubsub Item ID vs. Atom Entry id . . . . .	7
4.6.2	Message Body . . . . .	7
4.6.3	Geotagging . . . . .	7
<b>5</b>	<b>Profiles</b>	<b>8</b>
5.1	Base profile . . . . .	8
5.1.1	Pubsub Node Configuration . . . . .	9
5.2	Microblog profile . . . . .	9
5.3	Gallery profile . . . . .	9
<b>6</b>	<b>Security Considerations</b>	<b>9</b>
<b>7</b>	<b>IANA Considerations</b>	<b>10</b>
<b>8</b>	<b>XMPP Registrar Considerations</b>	<b>10</b>
<b>9</b>	<b>XML Schema</b>	<b>10</b>

## 1 Introduction

Social Networking platforms are often built around the concept of feeds. XMPP offers, through [Publish-Subscribe \(XEP-0060\)](#)<sup>1</sup>, a really generic way of Publishing and Subscribing elements on XMPP nodes. This extension defines a way to publish social content on Pubsub nodes using [RFC 4287](#)<sup>2</sup>

Social features in XMPP were historically built on the [Microblogging Over XMPP \(XEP-0277\)](#)<sup>3</sup> specifications. However, in practice, Microblogging was not clearly defined in more common cases such as on generic Pubsub nodes and it was not easy to differentiate social nodes from the other ones. The current XEP defines a more generic way of handling Social Feeds on XMPP by declaring a specific pubsub#type for those Pubsub nodes and by keeping [Microblogging Over XMPP \(XEP-0277\)](#)<sup>4</sup> as a subset of it.

## 2 Glossary

**Profile** Defined by a specific Pubsub type (pubsub#type) a profile is a Pubsub Social Feed with some constraints. See the Profiles section.

## 3 Discovery

If an entity implements Pubsub Social Feed, it MUST specify the 'urn:xmpp:pubsub-social-feed:1' feature in its service discovery information features as specified in [Service Discovery \(XEP-0030\)](#)<sup>5</sup> and the Entity Capabilities profile specified in [Entity Capabilities \(XEP-0115\)](#)<sup>6</sup>. If the entity is also implementing specific profiles, it MUST expose the specific feature defined in the related Pubsub type.

Listing 1: The entity responds with specific features profiles

```
<iq type='result'
  id='disco1'
  from='juliet@capulet.lit/sabo239'
  to='romeo@montague.lit/30d3d8'>
  <query xmlns='http://jabber.org/protocol/disco#info'>
    ...
    <feature var='urn:xmpp:pubsub-social-feed:1' />
    <feature var='urn:xmpp:pubsub-social-feed:gallery:1' />
    ...
  </query>
```

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

<sup>2</sup>RFC 4287: The Atom Syndication Format <<http://tools.ietf.org/html/rfc4287>>.

<sup>3</sup>XEP-0277: Microblogging over XMPP <<https://xmpp.org/extensions/xep-0277.html>>.

<sup>4</sup>XEP-0277: Microblogging over XMPP <<https://xmpp.org/extensions/xep-0277.html>>.

<sup>5</sup>XEP-0030: Service Discovery <<https://xmpp.org/extensions/xep-0030.html>>.

<sup>6</sup>XEP-0115: Entity Capabilities <<https://xmpp.org/extensions/xep-0115.html>>.

```
</iq>
```

## 4 Protocol

### 4.1 Location

A Pubsub Social Feed can be located on any Pubsub node on the XMPP network. Some specific use cases, such as the personal eventing (PEP) node Microblog is defined in the [Profiles](#) section.

### 4.2 Publishing a Post

The publication of a Post consist of a valid [RFC 4287](#)<sup>7</sup> entry embedded in a [Publish-Subscribe \(XEP-0060\)](#)<sup>8</sup> item.

The post content itself can be either text (content element without "type" attribute or with "type" attribute with "text" value) or XHTML ("content" element "type" attribute with "xhtml" value). If Romeo publishes XHTML content, his client MUST publish two "content" elements: a text one, and a XHTML one. For XHTML publishing, see [Publish-Subscribe \(XEP-0060\)](#)<sup>9</sup>.

Listing 2: Publishing a post

```
<iq from='romeo@montague.lit/pda'
  id='pub1'
  to='new.montague.lit'
  type='set'>
  <pubsub xmlns='http://jabber.org/protocol/pubsub'>
    <publish node='montague-family'>
      <item id='1cb57d9c-1c46-11dd-838c-001143d5d5db'>
        <entry xmlns='http://www.w3.org/2005/Atom'>
          <title type='text'>hanging out at the Caf&#233;
            Napolitano</title>
          <id>tag:montague.lit,2008-05-08:posts-1cb57d9c-1c46-11dd-838
            c-001143d5d5db</id>
          <published>2008-05-08T18:30:02Z</published>
          <updated>2008-05-08T18:30:02Z</updated>
        </entry>
      </item>
    </publish>
  </pubsub>
</iq>
```

---

<sup>7</sup>RFC 4287: The Atom Syndication Format <<http://tools.ietf.org/html/rfc4287>>.

<sup>8</sup>XEP-0060: Publish-Subscribe <<https://xmpp.org/extensions/xep-0060.html>>.

<sup>9</sup>XEP-0060: Publish-Subscribe <<https://xmpp.org/extensions/xep-0060.html>>.

Note: The "title" element is required to be included in an "atom:entry" element according to RFC 4287<sup>10</sup>. An implementation MAY provide also "atom:summary" and/or "atom:content" elements too if it needs.

#### 4.2.1 Publishing a Post with Rich Content

It's possible to insert some rich content in the post. It can be some text formatting, images, etc. Only "xhtml" content type is supported for the moment by this document but possibly it will be extended later.

Listing 3: Publishing a post with rich content

```
<iq from='romeo@montague.lit/pda'
  id='pub2'
  to='news.montague.lit'
  type='set'>
  <pubsub xmlns='http://jabber.org/protocol/pubsub'>
    <publish node='montague-family'>
      <item id='1cb57d9c-1c46-11dd-838c-001143d5d5db'>
        <entry xmlns='http://www.w3.org/2005/Atom'>
          <title type='xhtml'>
            <div xmlns="http://www.w3.org/1999/xhtml">
              <p>hanging out at the <strong>Caf&#233; Napolitano</strong></p>
            </div>
          </title>
          <id>tag:montague.lit,2008-05-08:posts-1cb57d9c-1c46-11dd-838c-001143d5d5db</id>
          <published>2008-05-08T18:30:02Z</published>
          <updated>2008-05-08T18:30:02Z</updated>
          <generator uri='https://capu.chat' version='42'>CapuChat</generator>
        </entry>
      </item>
    </publish>
  </pubsub>
</iq>
```

#### 4.3 Receiving a Post

Because Juliet is subscribed to some Romeo's family Pubsub service nodes. Romeo's XMPP server will send a Pubsub notification to Juliet. The notification can include an XMPP message body for backwards-compatibility with Jabber clients that are not pubsub-capable (see

<sup>10</sup>RFC 4287: The Atom Syndication Format <<http://tools.ietf.org/html/rfc4287>>.

Message Body ).

Listing 4: Receiving a post

```
<message from='news.montague.lit'
  to='juliet@capulet.lit'
  type='headline'>
  <body>hanging out at the Caf&#233; Napolitano</body>
  <event xmlns='http://jabber.org/protocol/pubsub#event'>
    <items node='montague-family'>
      <item id='1cb57d9c-1c46-11dd-838c-001143d5d5db' publisher='
        romeo@montague.lit'>
        <entry xmlns='http://www.w3.org/2005/Atom'>
          <title type='text'>hanging out at the Caf&#233;
            Napolitano</title>
          <link rel='alternate'
            type='text/html'
            href='http://montague.lit/romeo/posts/1cb57d9c-1c46-11
              dd-838c-001143d5d5db' />
          <link rel='alternate'
            href='xmpp:news.montague.lit?;node=montague-family;
              item=1cb57d9c-1c46-11dd-838c-001143d5d5db' />
          <id>tag:montague.lit,2008-05-08:posts-1cb57d9c-1c46-11dd-838
            c-001143d5d5db</id>
          <published>2008-05-08T18:30:02Z</published>
          <updated>2008-05-08T18:30:02Z</updated>
          <generator uri='https://capu.chat' version='42'>CapuChat</
            generator>
        </entry>
      </item>
    </items>
  </event>
</message>
```

Note: these alternate links were not posted by the original client because some clients can't compute them themselves. These things SHOULD be inserted at server side though.

#### 4.4 Replying to a Post

Anyone can publish a post in reply to Romeo's post. Here we assume that a reply comes from Benvolio.

Note: Inclusion of the "thr:in-reply-to" element defined in RFC 4685<sup>11</sup> indicates the post to which the user is replying. This reply includes two such elements (one pointing to the HTTP URL for the post and the other pointing to the XMPP URI for the post).

Note: The post can be a reply to more than the only one another.

<sup>11</sup>RFC 4685: Atom Threading Extensions <<http://tools.ietf.org/html/rfc4685>>.

Listing 5: Publishing a reply

```

<iq from='benvolio@montague.lit/mobile'
  id='uv2x37s5'
  to='news.montague.lit'
  type='set'>
  <pubsub xmlns='http://jabber.org/protocol/pubsub'>
    <publish node='montague-family'>
      <item id='c4145006-1c53-11dd-b2d5-000bcd82471e'>
        <entry xmlns='http://www.w3.org/2005/Atom'
          xmlns:thr='http://purl.org/syndication/thread/1.0'>
          <author>
            <name>Benvolio Montague</name>
            <uri>xmpp:romeo@montague.lit</uri>
          </author>
          <title type='text'>@romeo cappuccino this late in the day?</title>
          <link rel='alternate'
            type='text/html'
            href='http://montague.lit/benvolio/posts/c4145006-1c53-11dd-b2d5-000bcd82471e' />
          <link rel='alternate'
            href='xmpp:news.montague.lit?;
            .....node=montague-family;
            .....item=c4145006-1c53-11dd-b2d5-000bcd82471e' />
          <id>tag:montague.lit,2008-05-08:posts-c4145006-1c53-11dd-
            b2d5-000bcd82471e</id>
          <published>2008-05-08T18:31:21Z</published>
          <updated>2008-05-08T18:31:21Z</updated>
          <thr:in-reply-to
            ref='tag:montague.lit,2008-05-08:posts-1cb57d9c-1c46-11
              dd-838c-001143d5d5db'
            type='application/xhtml+xml'
            href='http://montague.lit/romeo/posts/1cb57d9c-1c46-11
              dd-838c-001143d5d5db' />
          <thr:in-reply-to
            ref='tag:montague.lit,2008-05-08:posts-1cb57d9c-1c46-11
              dd-838c-001143d5d5db'
            href='xmpp:news.montague.lit?;node=montague-family;item
              =1cb57d9c-1c46-11dd-838c-001143d5d5db' />
        </entry>
      </item>
    </publish>
  </pubsub>
</iq>

```

Assuming that Romeo has also subscribed to the same node he will receive the reply that Benvolio sent.



## 4.5 Repeating a Post

When Benvolio wants to repeat a Romeo's post, his client publishes the same post under a different name. But to be able to track the repeated post original author, Benvolio's client MAY use specific "atom:author" child node, "atom:name" and "atom:uri", containing, respectively, the name of the original post author, and his XMPP URI (JID).

The client SHOULD also put an "atom:link" element with "rel" attribute set to "via" and point it to the original post.

Listing 6: Repeating a Post

```
<iq from='benvolio@montague.lit/mobile'
  id='pub2'
  to='news.montague.lit'
  type='set'>
  <pubsub xmlns='http://jabber.org/protocol/pubsub'>
    <publish node='montague-family'>
      <item id='1re57d3c-1q46-11dd-748r-024943d2d5rt'>
        <entry xmlns='http://www.w3.org/2005/Atom'>
          <author>
            <name>Romeo Montague</name>
            <uri>xmpp:romeo@montague.lit</uri>
          </author>
          <title type='text'>hanging out at the Caf&#233;
            Napolitano</title>
          <link rel='alternate'
            type='text/html'
            href='http://montague.lit/benvolio/posts/1re57d3c-1q46-
              11dd-748r-024943d2d5rt' />
          <link rel='alternate'
            href='xmpp:news.montague.lit?;node=montague-family;
              item=1re57d3c-1q46-11dd-748r-024943d2d5rt' />
          <link rel='via'
            href='xmpp:rnews.montague.lit?;node=montague-family;
              item=1cb57d9c-1c46-11dd-838c-001143d5d5db'
            ref='tag:montague.lit,2008-05-08:posts-1cb57d9c-1c46-
              11dd-838c-001143d5d5db' />
          <id>tag:montague.lit,2008-05-08:posts-1re57d3c-1q46-11dd-748
            r-024943d2d5rt</id>
          <published>2008-05-08T18:30:02Z</published>
          <updated>2008-05-08T18:32:02Z</updated>
        </entry>
      </item>
    </publish>
  </pubsub>
</iq>
```

Thus, a different author JID value lets the client know the item has been repeated from another one.

It's also possible for Benvolio to add his own thoughts to the repost. To do this he SHOULD wrap the original content in the "xhtml:blockquote" element and after it add his own content. Also, the client MAY post reply without quotation to the original thread to inform users about the repost.

## 4.6 Atom and XMPP integration

### 4.6.1 Pubsub Item ID vs. Atom Entry id

There are two different things that carry a similar sense: the XMPP Pubsub Item ID and the "atom:id" element. This section is devoted to make a separation between them.

The pubsub Item ID MUST be used when linking to an entry with an XMPP channel (i.e. by including it in the URI with the "xmpp" schema). the Atom entry ID MUST be built according to [RFC 4287](#)<sup>12</sup> and used in aggregators with the aim of revealing of post duplicates, reposts, mentions, syndications, etc.

Note that the rules of comparing, building and security notes for "atom:id" are listed in the [RFC 4287](#)<sup>13</sup>.

### 4.6.2 Message Body

Depending on service policies and the value of the "pubsub#include\_body" node configuration option, Pubsub Social Feed notifications SHOULD include a message "body" element for backwards-compatibility with Jabber clients that are not pubsub-capable. It is RECOMMENDED for the XML character value of the "body" element to be the same as that of the "atom:title" child of the "atom:entry".

### 4.6.3 Geotagging

Juliet may want to know which places are Romeo's notices related to. That's why Romeo's client MAY geotag microblog entries, using the [User Geolocation \(XEP-0080\)](#)<sup>14</sup> protocol for storing geolocation information.

Romeo's client MUST create a "geoloc" element, with the [User Geolocation \(XEP-0080\)](#)<sup>15</sup> reference namespace: "http://jabber.org/protocol/geoloc".

Listing 7: Geotagging a Post

```
<iq from='romeo@montague.lit/mobile'  
  id='pub7'
```

---

<sup>12</sup>RFC 4287: The Atom Syndication Format <<http://tools.ietf.org/html/rfc4287>>.

<sup>13</sup>RFC 4287: The Atom Syndication Format <<http://tools.ietf.org/html/rfc4287>>.

<sup>14</sup>XEP-0080: User Geolocation <<https://xmpp.org/extensions/xep-0080.html>>.

<sup>15</sup>XEP-0080: User Geolocation <<https://xmpp.org/extensions/xep-0080.html>>.

```

to='romeo@montague.lit'
type='set'>
<pubsub xmlns='http://jabber.org/protocol/pubsub'>
  <publish node='urn:xmpp:microblog:0'>
    <item id='1zr23z8a-3g12-34fh-750b-120867gjc1sqh'>
      <entry xmlns='http://www.w3.org/2005/Atom'>
        <author>
          <name>Romeo Montague</name>
          <uri>xmpp:romeo@montague.lit</uri>
        </author>
        <title type='text'>Is lost in the forest. Need help!</title>
        <id>tag:montague.lit,2008-05-08:posts-1zr23z8a-3g12-34fh-750
          b-120867gjc1sqh</id>
        <published>2008-05-08T18:43:01Z</published>
        <updated>2008-05-08T18:43:01Z</updated>
        <geoloc xmlns="http://jabber.org/protocol/geoloc">
          <lat>48.171761</lat>
          <lon>-3.3667986</lon>
          <country>France</country>
          <countrycode>FR</countrycode>
          <region>Brittany</region>
        </geoloc>
      </entry>
    </item>
  </publish>
</pubsub>
</iq>

```

## 5 Profiles

Pubsub Social feeds are specified under profiles. Those profiles are constraints applied to nodes and defined by a specific Pubsub type (see [Publish-Subscribe \(XEP-0060\)](#)<sup>16</sup>#registrar-formtypes-metadata)

All the profiles MUST base their default configuration on the [Base profile](#).

### 5.1 Base profile

This profile is specified by the Pubsub type "urn:xmpp:pubsub-social-feed:1" and define a generic Pubsub Social Feed that can be hosted on any Pubsub service node or [Personal Eventing Protocol \(XEP-0163\)](#)<sup>17</sup> node.

Those restrictions MUST be used by all the other profiles defined bellow and in other XEPs based on Pubsub Social Feed.

<sup>16</sup>XEP-0060: Publish-Subscribe <<https://xmpp.org/extensions/xep-0060.html>>.

<sup>17</sup>XEP-0163: Personal Eventing Protocol <<https://xmpp.org/extensions/xep-0163.html>>.

### 5.1.1 Pubsub Node Configuration

1. The "pubsub#notify\_retract" MUST be set to "true" to provide clients the ability to track if some items were retracted and reflect such changes in the UI correctly.
2. The "pubsub#max\_items" MUST be set to the "max" value, as defined in the [Publish-Subscribe \(XEP-0060\)](#)<sup>18</sup>.
3. The "pubsub#persist\_items" MUST be set to "true".
4. The "pubsub#send\_last\_published\_item" SHOULD be set to "never".
5. The "pubsub#deliver\_payloads" SHOULD be set to "false". The social content can be quite large so it is advised to let the clients to manually query the content if it is not already cached.

## 5.2 Microblog profile

This profile is defined by the Pubsub type "urn:xmpp:microblog:0" as defined in [Microblogging Over XMPP \(XEP-0277\)](#)<sup>19</sup> and MUST be created and configured under the PEP "urn:xmpp:microblog:0" node.

## 5.3 Gallery profile

This profile is defined by the Pubsub type "urn:xmpp:pubsub-social-feed:gallery:1" and can be hosted on any Pubsub service.

All the items published in a gallery node MUST only have at least one attached picture. This picture MUST be of type "enclosure" as specified in [RFC 4287](#)<sup>20</sup>.

Listing 8: An Atom attached picture

```
<link rel='enclosure' href='https://capulet.lit/upload/romeo.jpg' type
  = 'image/jpeg' title='Romeo_Portrait' />
```

## 6 Security Considerations

There are no security features or concerns related to this proposal.

---

<sup>18</sup>XEP-0060: Publish-Subscribe <<https://xmpp.org/extensions/xep-0060.html>>.

<sup>19</sup>XEP-0277: Microblogging over XMPP <<https://xmpp.org/extensions/xep-0277.html>>.

<sup>20</sup>RFC 4287: The Atom Syndication Format <<http://tools.ietf.org/html/rfc4287>>.

## 7 IANA Considerations

This document requires no interaction with the [Internet Assigned Numbers Authority \(IANA\)](#) <sup>21</sup>.

## 8 XMPP Registrar Considerations

The [XMPP Registrar](#) <sup>22</sup> is requested to issue an initial namespace of "urn:xmpp:pubsub-social-feed:1" and "urn:xmpp:pubsub-social-feed:gallery:1".

## 9 XML Schema

This specification re-uses the schema for the Atom content format, i.e., the 'http://www.w3.org/2005/Atom' namespace (see RFC 4287).

---

<sup>21</sup>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/>.

<sup>22</sup>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/>.