XEP-0058: Multi-User Text Editing

Alexey Shchepin
mailto:alexey@sevcom.net
xmpp:aleksey@jabber.ru

2002-11-12
Version 0.1

Status	Type	Short Name
Deferred	Standards Track	

This document defines how several people may simultaneously edit text.

Legal

Copyright

Permissions

Warranty

Liability

Conformance

Contents

1 Introduction

This document defines an XMPP protocol extension for editing one text document by several people. This can be useful when several people write different parts of one single document, or one person edits the text and other people can see the changes. The advantage of using this protocol in compared to using a version control systems is that all changes are distributed between editors in real-time.

2 Protocol description

To start editing a file, the user must subscribe to it on special "server" The server stores text files. A User first can ask for list of these documents.

Listing 1: User asks the server list of editable texts

```
<iq type='get' to='server' id='mute1'>
  <list xmlns='http://jabber.org/protocol/mute#server'/>
</iq>
```

After that the server returns a list of the documents. For each list item an ID and description is given.

Listing 2: Ancestor returns list of editable texts

```
<iq type='result' to='editor1' id='mute1'>
  <list xmlns='http://jabber.org/protocol/mute#server'>
    <item id='id1' desc='Useful_text'/>
    <item id='id2' desc='Useless_text'/>
  </list>
</iq>
```

Then the user can send a subscribe request to the server by sending the documents ID. Attribute type can be rdwr for full access or rdonly for read-only access.

Listing 3: User sends subscribe request

```
<iq type='set' to='server' id='mute2'>
  <subscribe xmlns='http://jabber.org/protocol/mute#server'>
    <item id='id1' type='rdwr'/>
  </subscribe>
</iq>
```

Listing 4: Positive response from server on subscribe request

```
<iq type='result' to='editor1' id='mute2'/>
```

Listing 5: Negative response from server on subscribe request

```
<iq type='error' to='editor1' id='mute2'>
  <subscribe xmlns='http://jabber.org/protocol/mute#server'>
    <item id='id1' type='rdwr'/>
  </subscribe>
  <error code="405">Not Allowed</error>
</iq>
```

After successful subscription, the server sends the full contents of the document to the new editor.

Listing 6: Server sends text to editor

```
<iq type='set' to='editor1' id='mute3'>
  <text xmlns='http://jabber.org/protocol/mute#editor' id='id1'>
...
8. Admin Use Cases
   8.1. Banning a User
   8.2. Modifying the Ban List
   8.3. Granting Membership
   8.4. Revoking Mambership
   8.5. Modifying the Member List
   8.6. Granting Moderator Privileges
   8.7. Revoking Moderator Privileges
   8.8. Modifying the Moderator List
...
  </text>
</iq>
```

Listing 7: Positive response from editor

```
<iq type='result' to='server' id='mute3'/>
```

If an editor wants to commit the changes he made, then he sends a patch file to this text in GNU diff(1) unified format without first two lines (they specify file names and modification dates, that is not needed).

Listing 8: Editor sends patch to server

```
<iq type='set' to='server' id='mute4'>
  <patch xmlns='http://jabber.org/protocol/mute#server' id='id1'>
@@ -2,7 +2,7 @@
   8.1. Banning a User
   8.2. Modifying the Ban List
   8.3. Granting Membership
-  8.4. Revoking Mambership
+  8.4. Revoking Membership
   8.5. Modifying the Member List
```

```
        8.6.  Granting Moderator Privileges
        8.7.  Revoking Moderator Privileges
    </patch>
</iq>
```

Listing 9: Server accepts patch

```
<iq type='result' to='editor1' id='mute4'/>
```

Listing 10: Server not accepts patch

```
<iq type='error' to='editor1' id='mute4'>
    <patch xmlns='http://jabber.org/protocol/mute#server' id='id1'>
@@ -2,7 +2,7 @@
        8.1.  Banning a User
        8.2.  Modifying the Ban List
        8.3.  Granting Membership
-       8.4.  Revoking Mambership
+       8.4.  Revoking Membership
        8.5.  Modifying the Member List
        8.6.  Granting Moderator Privileges
        8.7.  Revoking Moderator Privileges
    </patch>
    <error code="405">Not Allowed</error>
</iq>
```

If server accepts the patch, then it resends it to all subscribed text editors and readers.

Listing 11: Server resend accepted patch to all editors and readers of this text

```
<iq type='set' to='editor1' id='mute5'>
    <patch xmlns='http://jabber.org/protocol/mute#editor' id='id1'>
@@ -2,7 +2,7 @@
        8.1.  Banning a User
        8.2.  Modifying the Ban List
        8.3.  Granting Membership
-       8.4.  Revoking Mambership
+       8.4.  Revoking Membership
        8.5.  Modifying the Member List
        8.6.  Granting Moderator Privileges
        8.7.  Revoking Moderator Privileges
    </patch>
</iq>
```

Listing 12: Positive response from editors and readers

```
<iq type='result' to='server' id='mute5'/>
```

If server receives an error IQ reply from one of the editors/readers on any of above requests, then it must remove him from subscribers list and may send a message to him about this.

3 Formal Definitions

Listing 13: DTD for http://jabber.org/protocol/mute#server

```
<?xml version="1.0" encoding="UTF-8" ?>

<!ELEMENT list ((item)*)>
<!ELEMENT subscribe (item)>

<!ELEMENT item EMPTY>
<!ATTLIST item
          id CDATA #REQUIRED
          desc CDATA #IMPLIED
          type ( rdwr | rdonly ) #IMPLIED >

<!ELEMENT patch (#PCDATA)>
```

Listing 14: DTD for http://jabber.org/protocol/mute#editor

```
<?xml version="1.0" encoding="UTF-8" ?>

<!ELEMENT patch (#PCDATA)>
<!ELEMENT text (#PCDATA)>
```