

## BEEM - Android XMPP

# BEEM

The goal of **Beem** is to provide a full featured and easy to use a Jabber/XMPP client on Android.

**Beem** is under constant development. Releases are usually frequent and driven by user contributions, such as bug reports and patches.

**Beem** is free and contains no ads. All our code is open source and licensed under the [GNU General Public License 3](#) and any later version. This means you can get **Beem's** code and modify it to suit your needs, as long as you publish the changes you make for everyone to benefit from as well.

beem\_twitter\_aos.png  
[link](#)

## Features

Available:

- Compatible with any standard XMPP server (Prosody, ejabberd, Openfire, Tigase, Google Talk, ...)
- SASL support
- Proxy (Socks4 Socks5 HTTP)
- Support of DNS SRV
- OTR

In development:

- MUC: see issue [#216](#)
- Jingle: <http://beem-project.com/hg/jingle/>
- Multi account: <http://beem-project.com/hg/account/>

## Screenshots

contact\_list.png send\_im.png settings.png

## Download

### Android package

qr-code.png [\\_en\\_generic\\_rgb\\_wo\\_60.png](#) [{{download}}](#)

Beem is available on the [Google Play Store](#) . You can also install it by flashing the QR code above.

We also provide you the Android package file (apk) for the latest version of BEEM at the [files section](#).

You can also download a live version of BEEM. This Android package is regenerated with the last source code every night: [download](#)

## Source

If you want to use the last unstable development version, then you should checkout the mercurial repository.

The mercurial repository is available at this address: <http://beem-project.com/hg/trunk> and viewable [here](#) and [here](#). You can see development repositories [here](#).

### Under GNU/Linux

In order to download BEEM sources, here is what you have to do: You first have to install mercurial (apt-get install mercurial under Debian).

Then in a console, enter the following commands:

```
hg clone http://beem-project.com/hg/trunk/ beem
```

To install the development version please see the file INSTALL for details: [source:INSTALL](#).

Once you have the repository, you can update it periodically so it matches the latest version from the main repository with the following command:

```
hg pull -u
```

## Lib XMPP

aSmack - buildsystem for Smack on Android: <https://github.com/Flowdalic/asmack>

## Support & getting help

We have a little [FAQ](#).

For getting help or discussing about BEEM, you can join #beem on the freenode IRC network or on XMPP MUC room: <xmpp:beem@conference.elyzion.net?join>.

We have a public mailing list, feel free to subscribe [here](#) and send mail to [beem-dev@list.beem-project.com](mailto:beem-dev@list.beem-project.com).

**Important Note:** this is a mailing list, **you may not receive an answer** if you have not registered.

You can also contact **privately** the **BEEM Team** at [contact@beem-project.com](mailto:contact@beem-project.com).

We are currently **not interested by any advertising services**. Such requests will just be ignored.

Before submitting a bug report, a patch or a feature request here, please read the [submission guidelines](#).

If you're interested, you can also have a look at our [Privacy Policy](#)

## Contributing and helping out

Beem is built and maintained by community volunteers. If you enjoy using it and would like to give back to the community, the [contribute](#) page has several ideas. Software development experience is not required.

# Beem Privacy Policy

You don't have to take our word on all of this. Beem is an open source project. Anyone can study our [Source Code](#) and make an informed decision.

**Beem** is an XMPP client that allows you to connect with a third party XMPP provider of your choice. **Beem** or its developers don't have control over the data you send to this provider through the application. You should also consult the privacy policy of your selected XMPP provider.

## 1. Sharing data

We never share your data with anyone.

## 2. Storing data

We never store your data.

# Contribute

So you want to contribute?

There are many ways to contribute on the BEEM project. The simplest one is just to help promote BEEM. Use it, show it, talk about it, share it with your friends.

Here's a list of other things you should do:

- **Get in touch:**  
Go to [beem@conference.elyzion.net](mailto:beem@conference.elyzion.net), that is our conference room. You can always ask for help and talk about BEEM there. [Report bugs](#) and share your mind about BEEM. Another way it's to join us on <irc://freenode.net/#beem>.
- **Improve translations:**  
You can use this <http://dev.beem-project.com/translation/index.php> (temporarily unavailable) page to translate Beem easily. The translation will be added in the next release. It is better to send us an email on [beem-dev@list.beem-project.com](mailto:beem-dev@list.beem-project.com) to tell us you are doing a translation. We will contact you if you are needed to update the translation before a release.
- **Get in touch with development:**  
Get the development version of BEEM. Install mercurial (either a console tool or one of the gui wrappers, like tortoisehg) and download BEEM sources using it.  
Make a jabber account specially for testing. It will be easier to test things if you will use a small Jabber account without all your friends. You will also less annoy them ;-)  
When you are ready you can submit your code in BEEM. See [submissions](#).
- **Start Hacking:**  
Just start! There's plenty to do, you can see all open tickets, also you can think of a new feature not already listed.

## Divers

### Fourre tout XMPP/Openfire

- [Comment filtrer les échanges XMPP entre le telephone et un autre client](#)

### Jingle-RTP

- [Liste des PayLoads pour le RTP](#)
- [Liste des formats audio/video prévu pour le streaming](#)

### Patch et mercurial

- [Comment faire un patch et l'appliquer sous mercurial](#)

### Checkstyle et formatage

- [Eclipse checkstyle](#)

### XEP sympa a implementer

- [Liste des XEP](#)

### Roadmap

- [Roadmap](#)

### tuto utiles sur les listview

<http://www.androidguys.com/?s=fancy+listview&x=0&y=0>

### Interesting links

- [An other XMPP client for android](#)
- [A Smack portage to Android project](#)

## Comment filtrer les échanges XMPP entre le telephone et un autre client

Il faut lancer wireshark sur localhost, et filtrer nos échanges (tcp.port == 5222 pour une conf de base).

Après le soucis c'est que les communications vont être compressées, pour désactiver la compression :

Server > Server Settings > Compression Settings > Client Compression > Not Available.

# Eclipse checkstyle

Comme nous l'avons précédemment écrit dans le cahier des charges du projet BEEM, nous allons écrire du code qui respecte une certaine norme notamment au niveau de la documentation javadoc des classes.

Pour vous faciliter la tâche et vous pourrir la vie, je mets à votre disposition un fichier de règle pour CheckStyle qui est un programme qui vérifie certaines règles dans le code source. Celui-ci est utilisable avec le plugin eclipse

<http://eclipse-cs.sourceforge.net/> Pour les presses, vous pouvez utiliser ce site dans le software update d'eclipse

<http://eclipse-cs.sf.net/update>

Ces règles ont été testées par Marseille. Si certaines règles ne vous conviennent pas pour une raison ou une autre, on en discutera dans ce ticket.

Ce fichier sera prochainement disponible dans le dépôt des sources du projet, ne vous inquiétez pas ^^

## Files

---

checkstyle.xml	7.67 KB	05/12/2009	Nikita Kozlov
----------------	---------	------------	---------------

# Roadmap

## Multi User Chat [XEP-0045]

Roles : Participant

Participant : Joindre Room, Quitter Room, Changer Pseudo, Inviter, Transformation 1on1 -> 1on multi, Envoyer Message, Envoyer message prive, Requesting Voice.

Affiliation (lien avec le chan) : Owner

Owner : Creer Room, Detruire Room

## User Interface / Ergonomie

### Service / Smack implementation

Utilisation du ServiceDiscovery voir class ServiceDiscoveryManager de Smack

## VOIP

### Jingle

Ice

Stun

### JlibRTP

### Codec/Decodec

Troll

## ContactSyncAPI

### Trouver Documentation

## Other

### Personal Event Protocol

User Tune [XEP-0118]

User ACTivity [XEP-0108]

User Mood [XEP-0107]

User Location [XEP-0080]

### User Avatar [XEP-0084]

We need to think about how we store user avatar. also need to finish the smack implementation, this should be easy.

## Divers

### Gestion des ressources

Necessite l'utilisation du contact provider. Une activite qui affiche la liste des ressources d'un contact lorsqu'on lui passe un jid incomplet ( [test@example.com](mailto:test@example.com)) afin de choisir a qui parler. Cette activite lance directement l'activite Chat si il n'y a 0 ou 1 ressource connecté ou un jid complet (test.exemple.com/pc)

### Externalisation de l'AsyncTask de LoginAnim

On peut creer une implementation abstract de l'AsyncTask. Cette abstraction lancerait juste la connexion, a charge a ses sous classes de definir leur changement niveau UI.

Exemple d'utilisation :

Lancement de l'activite Chat sans connexion. La connexion s'effectue en arriere plan pendant que l'user tape son message et des qu'elle disponible on active le bouton 'envoyer'

### Gestion des chats different mais avec un meme contact

On ne gere qu'un seul chat par contact. Si on obtient un nouveau chat pour le meme contact le nouveau remplace l'ancien et l'ancien est definitivement perdu.

Faire des tests en ouvrant un chat avec gajim puis en le fermant et reouvrant un chat.

### Gestion des message simples.

Ca peut être intéressant à faire rapidement. Ça permet d'envoyer des messages avec un sujet. Des messages plus en mode sms/mail.

**Liste des intents à catcher**

A définir

# FAQ

## About BEEM

### What does BEEM mean?

BEEM means "Boost your Eyes, Ears and Mouth".

### Do you guys have an XMPP MUC room?

Yes. You can find us at [beem@conference.elyzion.net](mailto:beem@conference.elyzion.net). You can also find us on the IRC at the channel #beem on freenode network.

### Why BEEM needs those crazy permissions??

Here is a list of the different permissions used by BEEM:

Permission	Need
android.permission.INTERNET	to access the internet connection
android.permission.VIBRATE	to set vibrate on Beem notifications
android.permission.WRITE_EXTERNAL_STORAGE	to stock the avatar datas on the SDCARD
android.permission.READ_EXTERNAL_STORAGE	to read the stocked avatar datas on the SDCARD
android.permission.ACCESS_NETWORK_STATE	to check the connectivity status
com.beem.project.beem.BEEM_SERVICE	Custom permission which will be used to allow other applications to use the Beem service
android.permission.GET_ACCOUNTS	As the 0.1.8 release, this is needed to be able to log you in Google Talk by using you device Google accounts
android.permission.USE_CREDENTIALS	As the 0.1.8 release, this is needed to be able to log you in Google Talk by using you device Google accounts

### Can I use BEEM in my own project?

You can use the source code of beem (in free or paid apps <http://www.gnu.org/philosophy/selling.html>) as most as you want as long as you respect the requirements of the GPLv3: <http://www.gnu.org/licenses/gpl.html>.

We have no commercial business around BEEM but we will really glad if other people can bring back to the upstream their ameliorations.

If you have any question we will be glad to help you.

### I want to help, what do you have for me?

Lots of stuff to do. Just see all those [crazy tasks](#).

### But I'm not a programmer!

That's OK. If you are into drawing stuff, you could design an icon set, emoticons set, or such. If you know some languages, you could help with translations. In either case, please just use BEEM, report your ideas, suggestions, tell us about your experience and feelings.

And, of course, help us to build a community, stay around in our XMPP MUC room, [beem@conference.elyzion.net](mailto:beem@conference.elyzion.net), chat around, answer newbies' questions and such. You can't imagine how it helps when there is at least a person or two that can help you with all that support-related stuff so that you can free a bit of time and context switches to write a bit more code.

## Common issues

### I use a Google Apps account and I can't connect with BEEM!!!

The Google Talk (GTalk) server use a non standard authentication scheme: they don't use the user part of the JID to authenticate but instead they use the full JID.

Starting to Beem-0.1.5 version, you can use an option in the advanced preferences to use the full JID to authenticate.

For previous version, if the domain part of your JID is not gmail.com or googlemail.com you have to enter in BEEM a JID in this form:

[user@example.com](mailto:user@example.com)@example.com.

### I want to use Facebook Chat with Beem.

To use your Facebook account in Beem you must:

- have a [Facebook username](#). After setting your username, you must log out of and into Facebook once.
- your Facebook Jabber ID is simply your Facebook user name with @chat.facebook.com appended (eg: [username@chat.facebook.com](mailto:username@chat.facebook.com)).
- don't set the "use full JID to authenticate" option in advanced settings.

# How to generate a patch file

(From : <http://mercurial.selenic.com/wiki/CommunicatingChanges>)

## import/export

This method is useful for receiving small numbers of changes from contributors and is a traditional method for open source projects. This is primarily used in scenarios where changes get reviewed before being applied to a central repository.

In combination with the push/pull method, it's common to post patches generated by export to a mailing list, along with instructions for pulling from their source repository.

This can also be useful for "cherry-picking" individual patches from one repo into another.

```
$ hg export 1135
# HG changeset patch
# User Thomas Arendsen Hein <thomas@intevation.de>
# Node ID e455d91f62598b8f255ce6c0291afe8f8565e0d2
# Parent 2cd33ea2f66bae0eb7415cfcd7eab88566fdb1aa
Variable 'body' was missing in patchbomb script.

diff -r 2cd33ea2f66b -r e455d91f6259 contrib/patchbomb
--- a/contrib/patchbomb Sun Aug 28 16:30:40 2005
+++ b/contrib/patchbomb Sun Aug 28 16:52:55 2005
@@ -107,6 +107,7 @@
     def makepatch(patch, idx, total):
         desc = []
         node = None
+        body = ''
         for line in patch:
             if line.startswith('#'):
                 if line.startswith('# Node ID'): node = line.split()[-1]
$ hg export 1135 > ../body.patch
$ cd ../work
$ hg import ../body.patch
```

## bundle/unbundle

This method allows communication of patches by exchanging "bundles": a compressed group of changesets in a native file format. These bundle files can then be exchanged via email attachments, FTP, floppy disk, etc.

This also allows you to publish your changes without publishing a copy of the entire project history.

```
$ hg bundle changes.hg http://upstream/repo
searching for changes
$ scp changes.hg server:public_html
$ cd ../other
$ hg unbundle http://server/~user/changes.hg
adding changesets
adding manifests
adding file changes
added 13 changesets with 20 changes to 6 files
```

# Liste des formats audiovideo prévu pour le streaming

<http://www.networksorcery.com/enp/rfc/rfc3551.txt>

## Liste des PayLoads pour le RTP

PT	encoding name	audio/video (A/V)	clock rate (Hz)	c
hannels	(audio) Reference			
0	PCMU	A	8000	1
	[RFC3551]			
1	Reserved			
2	Reserved			
3	GSM	A	8000	1
	[RFC3551]			
4	G723	A	8000	1
	[Kumar]			
5	DVI4	A	8000	1
	[RFC3551]			
6	DVI4	A	16000	1
	[RFC3551]			
7	LPC	A	8000	1
	[RFC3551]			
8	PCMA	A	8000	1
	[RFC3551]			
9	G722	A	8000	1
	[RFC3551]			
10	L16	A	44100	2
	[RFC3551]			
11	L16	A	44100	1
	[RFC3551]			
12	QCELP	A	8000	1
13	CN	A	8000	1
	[RFC3389]			
14	MPA	A	90000	
	[RFC3551] [RFC2250]			
15	G728	A	8000	1
	[RFC3551]			
16	DVI4	A	11025	1
	[DiPol]			
17	DVI4	A	22050	1
	[DiPol]			
18	G729	A	8000	1
19	Reserved	A		

20	Unassigned	A	
21	Unassigned	A	
22	Unassigned	A	
23	Unassigned	A	
24	Unassigned	V	
25	CelB	V	90000
	[RFC2029]		
26	JPEG	V	90000
	[RFC2435]		
27	Unassigned	V	
28	nv	V	90000
	[RFC3551]		
29	Unassigned	V	
30	Unassigned	V	
31	H261	V	90000
	[RFC2032]		
32	MPV	V	90000
	[RFC2250]		
33	MP2T	AV	90000
	[RFC2250]		
34	H263	V	90000
	[Zhu]		
35-71	Unassigned	?	
72-76	Reserved for RTCP conflict avoidance		
	[RFC3550]		
77-95	Unassigned	?	
96-127	dynamic	?	
	[RFC3551]		

Retrouvable sur <http://www.iana.org/assignments/rtp-parameters>

# Liste des XEP

## Intro

Voici une liste d'extension qu'ils nous faudrait implementer dans BEEM. Cette liste est non exhaustive et sert surtout pour avoir une idee des differents ecrans utilisateur (activite android) qu'il faut realiser.

## Extensions requise pour BEEM

### XMPP Core Client 2009

Un client XMPP doit implementer ces specifications:

- [rfc3920](#) Cette specification definie le coeur du protocole XMPP
- [rfc3921](#) Cette specification definie un ensemble d'extension basique pour le XMPP
- [XEP-0030: Service Discovery](#) Cette specification permet de decouvrir les services disponible avec un agent XMPP, aussi bien un client que le serveur ou autre chose. Ainsi on peut par exemple gerer que notre contact supporte la video donc on active la possibilite de communiquer avec lui par webcam.
- [XEP-115: Entity capabilities](#) Cette specification permet de gagner de la bande passante lors de la decouverte des services disponibles.

Les rfc 3920 et 3921 sont mises a jours par ces 2 rfc [rfc3920bis](#) et [rfc3921bis](#) . Dans l'ensemble ces 4 specifications sont implementes dans Smack donc niveau protocole, il ne devrait pas y avoir grand chose a coder. Je vous conseille de parcourir le sommaire de ces specs afin de voir quel fonctionnalites activer dans l'interface utilisateur.

### XMPP Advanced Client 2009

Un client XMPP "avance" doit implementer ces specifications :

- [XEP-0016: Privacy Lists](#) Cette extension specifie le protocole pour implementer le blocage d'un contact (comme msn)
- [XEP-0191: Simple Communications Blocking](#) est une simplification de la XEP-0016 afin de faciliter l'implementation de cette fonctionnalite donc c'est surtout elle qu'il faut regarder
- [XEP-0045: Multi-User Chat](#) Cette specification permet d'implementer des salons de discussion like irc
- [XEP-0054: vcard-tmp](#) permet de stocker une v-card sur le serveur et ainsi permettre de recuperer des informations sur un contact
- [XEP-0085: Chat State Notifications](#) Lors d'une discussion, cette extension permet de signaler que quelqu'un est en train d'ecrire un message (par exemple)

La XEP-0085 ne semble pas tres interessantes pour BEEM car elle cause aussi une utilisation supplementaire de la bande passante sans apporter de reel interet pour une utilisation sur mobile.

La XEP-0054 est deja assez ancienne et risque fort d'etre remplacer par la [XEP-0154: User Profile](#) qui est en status experimental.

## Jingle

Comme vous le savez tous Jingle est l'extension XMPP qui permet de realiser des echanges multimedia. Cette extension est plutot grosse, elle est donc definie en plusieurs XEP.

- [XEP-0166: Jingle](#) definit le coeur du protocole Jingle
- [XEP-0167: Jingle RTP Sessions](#) specifie le protocole pour realiser une session en utilisant le protocole RTP
- [XEP-0176: Jingle ICE-UDP Transport Method](#) specifie une methode de transport utilisant ICE-UDP. Cette methode de transport permet notamment de regler les problemes de NAT.
- [XEP-0177: Jingle Raw UDP Transport Method](#) specifie une methode de transport en UDP. C'est une methode simple qui marche dans les configurations simples.
- [XEP-0262: Use of ZRTP in Jingle RTP Sessions](#) specifie comment utiliser zrtp a la place du simple RTP
- [XEP-0272: Multiparty Jingle](#) specifie comment coordonner des sessions jingle entre un groupe de personnes et donc permet de faire des conferences multi.

## Autres extensions interessantes

### [Personnal Eventing Protocol](#) ou PEP

PEP est une simplification de l'extension [Publish-Subscribe](#) realise dans l'optique de permettre a l'utilisateur de definir des information contextuelle tel que la musique qu'il ecoute ou encore son activite actuelle. Elle definie un ensemble de specifications pour chaque possibilite.

[XEP-0118: User Tune](#) pour la fonction "Je suis en train d'ecouter ..."

[XEP-0108: User Activity](#) pour "Je suis en train de dormir"

[XEP-0107: User Mood](#) pour "Je suis fatigue"

[XEP-0080: User Location](#) pour "Je suis a la Tour Eiffel plus precisement a latitude 42 longitude 42"

[XEP-0084: User Avatar](#) pour envoyer son nouvel avatar

## Avatars

Afin de gerer les avatars de l'utilisateur on peut implementer ces specification:

- [XEP-0153: vCard-Based Avatars](#) cette extension est actuellement la plus utilise pour gerer les avatars. Cependant elle est destinee a etre remplacee par la suivante

- [XEP-0084: User Avatar](#) Cette extension elle utilise Publish-Subscribe

## Gains en bande passante

[XEP-0138: Stream Compression](#) pour compresser la connexion avec le serveur

[XEP-0237: Roster Versioning](#) Cette extension permet d'éviter l'envoi de toute la liste de contact. Elle met en place un mécanisme de version de la liste envoie uniquement les changements entre les différentes versions de la liste de contact. Elle est également spécifiée dans la RFC3921bis

## Divers

[XEP-0126: Invisibility](#) pour apparaître déconnecté.

[XEP-0154: User Profile](#) Cette extension est destinée à remplacer les vCards. Elle permet le stockage de beaucoup plus d'informations et pourrait même servir à mettre en place un social networking <-- Deferred by the XMPP Standards Foundation, en gros faut plus l'utiliser!

# Comment faire un patch et l'appliquer sous mercurial

## Comment generer un patch ?

simple. dans votre depot vous faite : (ou \$base est le numero de la revision de base de vos modifications)

```
hg export $base > monpatch.patch
```

Vous obtenez un joli fichier de diff que vous pouvez utiliser normalement (voir man patch)

L'inconvenient est que ce fichier ne contient pas les informations de commits, etc. On va d'autre utiliser une autre commande de mercurial pour obtenir un fichier bundle qui contiendra le patch plus les infos de commits etc

```
hg bundle --base $base monbundle.bz2
```

ou

```
hg bundle monbundle.bz2 $depot
```

Il est alors possible d'importer ce bundle

```
hg pull monbundle.bz2
```

Bon pour plus d'infos n'hesiter pas a voir la doc de mercurial (man hg)

# Submissions on BEEM

You can make a submission anonymously. However, to comment on your issue, you [must register](#) on the issue system.  
**Do not create a new issue just to comment.**

Submissions should be written in english, french is also accepted but less desirable.

## Submitting a Bug

[How to report bugs effectively](#)

This document contains general information about submitting a bug report. Please read it carefully.

Before submitting a bug report here, please make sure:

- that you're using the last version of BEEM

And have a look at the [bug list](#) and the [FAQ](#) to see if it has not already been reported.

When filling a bug report, please give the following information with the description of your problem:

- your Android device
- your Android version
- your Beem version (pick the version number in the *Affected version* field) and the exact revision (eg. r1142) if you're using repository code
- the error stack trace that you should find in the log file if your report is about an error. The message displayed by the app (eg: *...An error occurred when trying to connect...*) is useless here

If you have access to an Android SDK, please give the stacktrace of the crash. You can obtain it by typing in a console when the phone is plugged on your computer:

```
$> adb logcat
```

You can also use the application [alogcat](#) available on the Android Market.

The simplest way to get information about your Android version is:  
Settings -> About Phone -> Model Number + Model Version

## Submitting a Feature Request

Before submitting a feature request here, please make sure:

- fill the *subject* as a sum up of your description
- fill the *description* as detailed as you can, don't hesitate to describe precisely how this feature should behave, try to add what benefits you're expecting from it
- please **DON'T** fill the *target version* field: let the developers decide when integrate
- please **DON'T** set a high priority, let the developers deal with it
- please **DON'T** assign the feature request to anybody, let the developers deal with it

## Submitting a Wiki improvement

When you edit a wiki page to improve its content, make sure that you fill the *comment* field with a descriptive summary of the changes you've made to the page. This way we'll be better able to "monitor" and "maintain" the quality of the actual wiki content.

## Submitting codes

The preferred way to submit code is to make a mercurial pull request. Submit an issue to the "**Pull request**" tracker. Don't forget to fill the *repository* field and add a descriptive summary of your changes. Please specify also the revision you wish us to pull.

Alternatively, you can submit changes by sending us a patch:

- Changes should be submitted as a single patch file. Instructions for creating a patch using mercurial can be found [here](#)
- You should mention the branch/revision your patch is based on and attach it