
Babble Documentation

Release 1.0

Jan-Carel Brand

October 26, 2013

CONTENTS

Contents:

BABBLE PLONE CHAT CLIENT

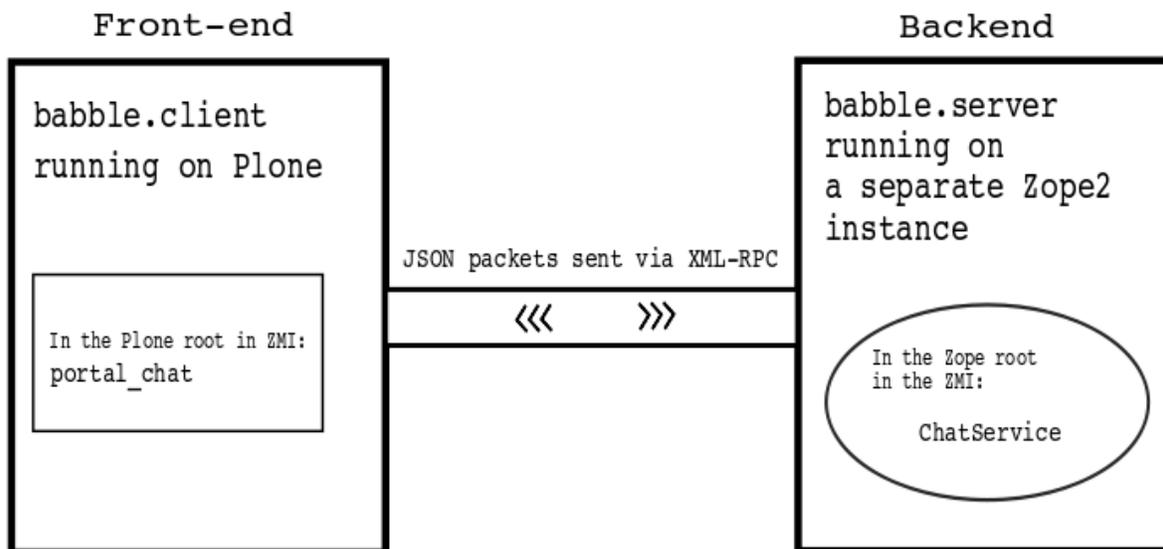
1.1 Introduction

Babble is an instant messaging service for Plone. It consists of `babble.client` as the Plone front-end client and `babble.server`, a Zope2 messaging service, as the backend chat service.

The client consists of an *Online contacts* portlet with which you can initiate new chats. Chat sessions occur in modal dialog chatboxes which make use of JQuery and Ajax polling to provide seamless, real-time messaging. During inactive periods, the polling interval gradually becomes longer until it reaches a specific (configurable) maximum polling interval.

Communication between the client and server is achieved with JSON packets sent via XML-RPC. The XML-RPC configuration values are stored in the *portal_chat* tool which can be found in the root of the Plone site inside the ZMI (Zope management interface). See the Portal Chat Tool for a breakdown of its configuration values.

On the server side (backend), a **Chat Service** object needs to be created, which will then act as a messaging server. Refer to the Chat Service section on how to create the **Chat Service**.



1.1.1 Features

- Remembers open chat windows on page reload

- Chat windows can be minimized
- New messages automatically opens chat window
- An ‘online users’ portlet provides a list of currently online users
- Configurable polling intervals
- Clickable URLs received via chat messages
- Requests to the messaging service are password authenticated
- Can be [integrated](#) with the ActionBar of `actionbar.panel`
- Can run on a different server than the messaging service

1.1.2 Compatibility

Tested to work with Plone 3.3 and Plone 4

1.2 Configuration

1.2.1 Quickly setting up a demonstration

There is a [demo](#) egg available on Plone.org and pypi, with which you can quickly set up a new Babble configured Plone 4 site for demonstration purposes.

See the README of [babble.demo](#), or the section titled **Setting up a quick chat demo**, in this document, for instructions on how to set it up.

If you however want to configure the system yourself, then read the next section.

1.2.2 Manual configuration

Chat Service

`babble.client` needs a running zope instance configured with the `babble.server` chat service.

`babble.server` is a pure Zope2 application and the **Chat Service** object that it provides, should be added in the Zope Management Interface (ZMI). It **must be created in the Zope root** of your Zope instance, otherwise it won't work.

NOTE: It's strongly recommended that you run the `babble.server` messaging service in a separate server or Zeo Client. Running the client and server in the same instance sometimes causes the browser (probably Zope's ZPublisher) to become unresponsive.

Portal Chat Tool

The client is configured via the `portal_chat` tool in the ZMI:

- **Service name:** This is the name of `babble.server`'s ‘Chat Service’ object that you created in the Zope root of the Zope instance you will be using as your message server.
- **The chat service is on the same machine:** Check this if the chat service is created on the same Zope/Plone instance and is therefore not running on a different server. This will disable XML/RPC and instead use direct method calls. Note that it's recommended to run your Chat Service in it's own ZODB, so if you want to run it in the same instance, consider [mounting a separate ZODB](#).

Undo	Ownership	Interfaces	Find
			None
			<input type="button" value="Add Plone Site"/>
			<input type="text" value="Chat Service"/> <input type="button" value="Add"/>
Size	Last Modified		
	2010-04-16 21:23		
	2010-04-17 00:47		
	2010-04-17 00:10		
	2010-04-16 21:23		
	2010-04-17 00:47		
	2010-04-16 21:23		
21 Kb	2010-04-16 21:23		
1 Kb	2010-04-16 21:23		
	2010-04-16 21:23		
1 Kb	2010-04-16 21:23		
	2010-05-04 15:47		
	2010-04-16 21:23		

- **Hostname or relative path:** If the Chat Service is on a different server, then specify the hostname here. Otherwise, if the Chat Service is on the same server and not in the Zope root, then specify the path to it here.
- **Port:** This is the port number of the server running the ‘Chat Service’. Only applicable when the Chat Service is on a different server.
- **Username:** This is the username of the Zope user that you used to create the ‘Chat Service’.
- **Password:** This is the password of the Zope user that you used to create the ‘Chat Service’.
- **Maximum polling interval:** The time interval between polls to the `babble.server` chat service starts off at a specified minimum value and then gradually increases to a specified maximum value during periods where there isn’t any chat activity. Specify the minimum polling interval value here.
- **Minimum polling interval:** The time interval between polls to the `babble.server` chat service starts off at a specified minimum value and then gradually increases to a specified maximum value during periods where there isn’t any chat activity. Specify the maximum polling interval value here.

Name	Value	Type
Service name:	<input type="text" value="chatservice"/>	string
The chat service is on the same machine.	<input checked="" type="checkbox"/>	boolean
Hostname or relative path (relative to Zope root):	<input type="text"/>	string
Port:	<input type="text" value="8080"/>	string
Username:	<input type="text" value="admin"/>	string
Password:	<input type="text" value="admin"/>	string
Maximum polling interval: (milliseconds)	<input type="text" value="20000"/>	int
Minimum polling interval: (milliseconds)	<input type="text" value="5000"/>	int

1.2.3 How do I start using it?

Make sure that `babble.client` is installed via Plone’s control panel, or the `portal_quickinstaller` tool in the ZMI (Zope management interface).

In Plone, go to the portlets manage page. In the dropdown of addable portlets, there should now be a new types of portlet, *Online contacts*.

Add this portlet. If you have more than one person currently using the site, you should see them appear in this portlet.

Note: When you are running your `portal_javascripts` registry is in debug mode (or when you are running `./bin/instance fg` in Plone4), then the *Online contacts* portlet will show *all* the registered users, and not just the online ones, to make debugging easier. **As of version 1.3.2, to avoid confusion, this is no longer the case.**

Simply click on the user in the portlet, and a chatbox will appear in the bottom right of the page.

Now start babbling!

Search Site
 only in current section

Alex Pilz [Dashboard](#) [Preferences](#) [Log out](#)

Who's online?

- Cillian de Roiste
- JC Brand

owner of this web site has just installed Plone. Do not contact the Plone Team or the Plone mailing lists about this.

se do the following:

(You should have a Site Setup entry in the menu in the top right corner)

verify users and send out password reminders)

elf registration, password policies, etc)

content editors and [Server setup recommendations](#).

it do what you want it to. Some examples:

s, or one of the [available themes from plone.org](#). (Make sure the theme is compatib

efault is a website-centric workflow, if you want to use Plone as a closed intranet, you sh

fer text-based syntax and/or wiki syntax, you can change this in the [markup control pane](#)

jcbrand	cillian
<p>jcbrand 15:49: This is a very important chat message...</p> <p>alex 15:49: Good</p>	<p>alex 15:48: Hi Cillian</p> <p>cillian 15:49: Hey...</p>

1.3 Troubleshooting

1. I get a ‘connection error’ message inside the chat window

The ‘connection error’ message happens whenever the chat client (*babble.client*) cannot communicate with the chat server (*babble.server*).

This is usually because the chat server is not running (i.e when you restart the zope server).

Check that the chat server is running and that your settings in the portal_chat tool are correct.

2. Plone freezes or locks up completely and I have to restart the instance before this goes away

This happens when there is only one Zope instance that has both the chat client and the chat server running in it.

The chat client ‘polls’ the chat server to find new messages. If the server stops responding because it is overwhelmed or because it is restarted, then the browser hangs forever as it waits for a response, even after Zope was restarted.

The best solution is to run the chat server in a separate dedicated Zeo instance not being used for anything else.

Make sure to change the port in the portal_chat tool to point to this Zeo instance.

A second thing you can do, is to increase the minimum poll time in the portal_chat tool, for example from 3000 ms to 6000 ms or more.

This will of course mean that messages can take longer to appear for the conversation partners.

3. Nothing happens when I click on a contact’s name in the portlet

Check the debug output on Firebug for any clues. If you get messages that something is undefined, such as “jQuery is not defined”, then the ordering in portal_javascripts is probably wrong.

chat.js, jquery.ba-dotimeout.min.js, jquery.cookie.js and quicksearch.js must all come *after* jquery.js.

In Plone 3, you might have collective.js.jquery installed, in which case a newer jQuery version is installed and jquery.js is disabled. Make sure that the newer library is *before* the above mentioned scripts.

4. Babble doesn’t work with deliverance or collection.xdv

That is because deliverance and collective.xdv need to be told to ignore the ajax calls that Babble makes.

For collective.xdv, add the following to the *Unstyled paths* textbox in the @@xdv-settings configuration view (found via *Site setup*):

```
^.*/@@babblechat.*$
^.*/@@render_chat_box
```

For deliverance, you’ll need proxy declarations similar to the following in your rules.xml file:

```
<proxy path="regex:^.*/@@babblechat.*$">
  <dest href="http://localhost:8080/VirtualHostBase/http/localhost:5099/MySite/VirtualHostRoot" />
</proxy>

<proxy path="regex:^.*/@@render_chat_box">
  <dest href="http://localhost:8080/VirtualHostBase/http/localhost:5099/MySite/VirtualHostRoot" />
</proxy>
```

Where MySite is your Plone site’s name, running on 8080 and the deliverance proxy on 5099.

5. When I chat, a different user’s name appears as the message author in the chat box. When I reload the page, the problem goes away.

Babble makes use of a dynamically generated javascript file named *chat.js*. Make sure that this file is not cached or merged, otherwise this problem can occur. Go to the `portal_javascripts` tool in the Zope Management Interface (ZMI) and uncheck those two options for this file.

BABBLE MESSAGING SERVER

2.1 Introduction

babble.server is the messaging service for the Babble instant messaging service for Plone. It's however a pure Zope2 product and not dependent on Plone. It doesn't depend on any of the *babble.client* frontend code and can be completely decoupled from it, by running it on a separate server.

2.2 Features

- **User accounts:** users must be registered for the chat service independent of a Plone site's users.
- **Security:** messaging actions require the user to authenticate.
- **User status support:** users can set their status, such as 'busy', 'chatty' or 'invisible'. Note: the *babble.client* frontend doesn't support this.
- **Web service:** all public API methods return JSON strings.
- **Tests:** 100% test coverage.

2.3 Setup

The chatserver is designed to be a standalone and lightweight Zope2 app. It doesn't depend on Plone at all.

Since instant messages result in many writes to the database, it's best to run the chat server on it's own ZODB instance. You can do this by either dedicating a standalone Zope2 instance with its own ZODB, or by [mounting a separate ZODB](#) into your site.

In the case of using a standalone Zope2 instance, you could do the following in your buildout.cfg:

```
[buildout]
parts += chatserver

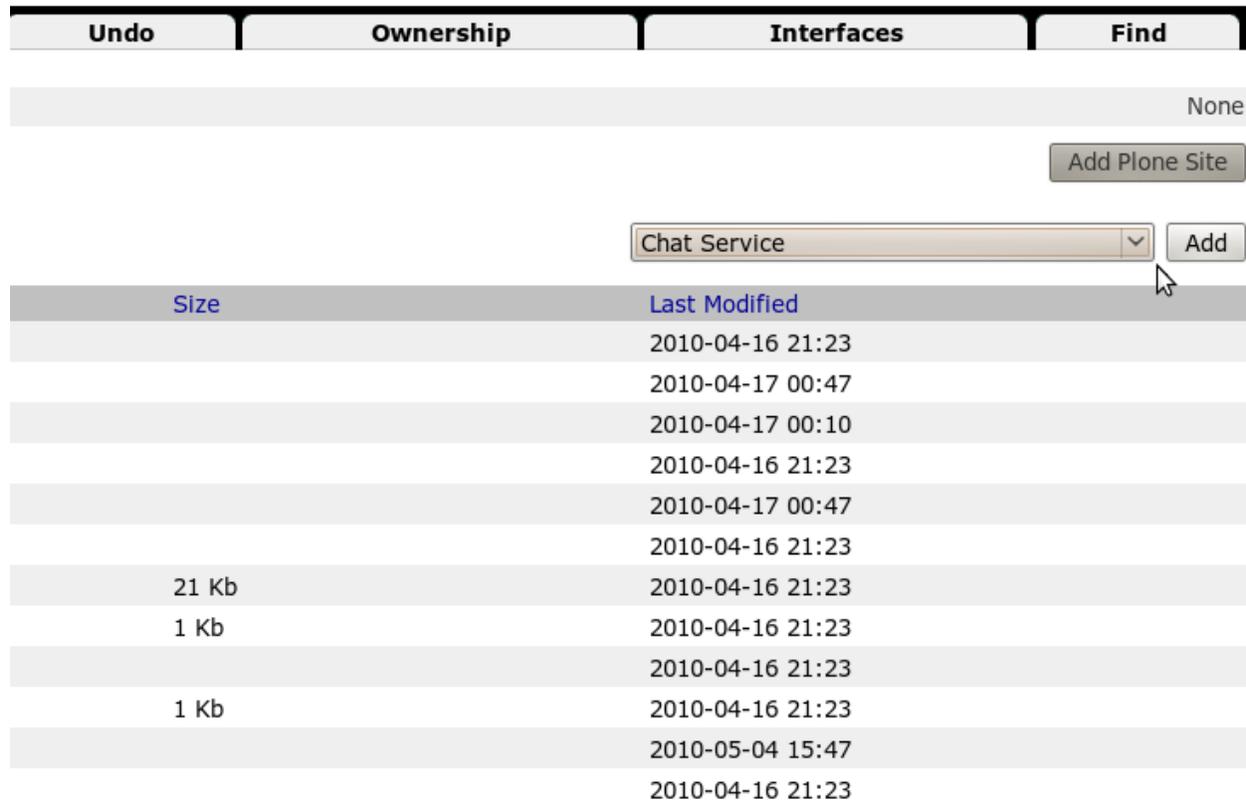
[chatserver]
recipe = plone.recipe.zope2instance
http-address = 8081
zeo-client = off
shared-blob = off
user = admin:admin
file-storage = filestorage/DataChat.fs
blob-storage = var/blobstorage-chat
```

```
eggs = babble.server
zcml = babble.server
```

Note: It's important to include the zcml slug for babble.server.

2.4 Usage

Simply add the *Chat Service* in the ZMI (Zope Management Interface) by clicking on the dropdown box and choosing it from the list.



The screenshot shows the ZMI 'Interfaces' tab. At the top, there are four tabs: 'Undo', 'Ownership', 'Interfaces', and 'Find'. Below the tabs, there is a search bar with 'None' and an 'Add Plone Site' button. A dropdown menu is open, showing 'Chat Service' selected, with an 'Add' button next to it. Below the dropdown is a table with two columns: 'Size' and 'Last Modified'.

Size	Last Modified
	2010-04-16 21:23
	2010-04-17 00:47
	2010-04-17 00:10
	2010-04-16 21:23
	2010-04-17 00:47
	2010-04-16 21:23
21 Kb	2010-04-16 21:23
1 Kb	2010-04-16 21:23
	2010-04-16 21:23
1 Kb	2010-04-16 21:23
	2010-05-04 15:47
	2010-04-16 21:23

2.5 API

For a detailed look at the API methods with a description of each ones purpose and the JSON values it returns, please see the [interfaces.py](#) module.

SETTING UP A QUICK CHAT DEMO

3.1 Introduction

babble.demo sets up a fully configured and running **Babble** instant messaging environment for Plone.

It's intended as a demo and aid to help integrators familiarise themselves with how Babble works and what it has to offer.

All the eggs (currently) related to Babble are installed. These are:

- **babble.server:**
 - The Zope messaging service. This service should ideally be run in it's own dedicated server or Zeo client (as in this case).
- **babble.client:**
 - The Plone client side program. This provides the main chat functionality such as the *online contacts* portlet and the chat boxes.
- **actionbar.panel:**
 - This add-on is not directly associated with Babble and is not a requirement of it. It provides an extendable action panel that hovers at the bottom of the screen.
- **actionbar.babble:**
 - The Babble extension for *actionbar.panel*. It provides a special action-box that provides a list of currently online users. Additionally, chatboxes are docked on this panel.

3.2 Steps to get a running system:

babble.demo provides its own buildout configuration file, *plone4.cfg*.

You should use this file, together with buildout, to create a new Zope/Plone instance.

In the examples below, assume that the python version is 2.6.* with PIL (the python imaging library) installed:

```
python bootstrap.py -c plone4.cfg
```

```
./bin/buildout -c plone4.cfg
```

The buildout will install a Zeo server running on port 2000 with two clients, *instance* on port 8080 and *chatserver* on port 8081.

supervisor is installed to manage these processes. After the buildout has completed successfully, start the supervisor:

```
./bin/supervisord
```

You can check on the commandline whether the processes started successfully:

```
./bin/supervisorctl status
```

Or in your browser by visiting <http://localhost:9000>, with username and password 'admin'.

Now, in your browser, go to <http://localhost:8080>, and click on the *Create a new Plone site* button. In the next page, choose `babble.demo` from the list of 'add-ons' and then click *Create Plone site*.

Babble and the ActionBar should now be installed and correctly configured.

Head to <http://localhost:8080/Plone/@@usergroup-userprefs>, and add some users.

Happy Babbling!



Create a Plone site

Path identifier

The id of the site. This ends up as part of the URL.

No special characters are allowed.

Title

A short title for the site. This will be shown in the title of the browser window on each page.

Language

The main language of the site.

Add-ons

Select any add-ons you want to activate immediately. You can also activate add-ons after the site has been created using the Add-ons control panel.

- OpenID Authentication Support**
Adds support for authenticating with OpenID credentials in a Plone site
- Working Copy Support (Iterate)**
Adds working copy support (aka. in-place staging) to Plone.
- actionbar.babble**
Babble ActionBar integration
- actionbar.panel**
Extension profile for ActionBarPanel.
- babble.client**
Instant messaging client for Plone
- babble.demo**
Demo for Babble instant messaging

BABBLE INTEGRATION WITH ACTIONBAR

4.1 Introduction

actionbar.babble is the **Babble** chat client's integration package for the **ActionBar**, an extensible floating panel with links at the bottom of your Plone site.

For more information regarding Babble and ActionBar, please see their respective packages: *babble.client*, *babble.server* and *actionbar.panel*.

4.2 How to use

Just install *actionbar.babble* via the Plone control panel, or the *portal_quickinstaller* tool. *actionbar.panel* and *babble.client* will be installed automatically as well.

Make sure that the *portal_chat* tool for *babble.client* is configured correctly.

You should now automatically have the ActionBar at the bottom of your Plone site. At the bottom right, you will see a section called *Online Users*. Clicking on it will popup a dialog showing all the current online users you can chat with.

4.3 Troubleshooting:

1. The *Online Users* box, and the individual chatboxes appear slightly lower and behind the ActionBar panel. When you minimize a box, it disappears behind the panel.

Make sure that the *babblefeeder.css* file is loaded *after* the *chat.css* file in the *portal_css* resource registry in the Zope management interface. You can do this by hand, or simply re-run the *actionbar.babble* CSS import step in the *portal_setup* tool.

me News Events Users

Welcome to Plone

Imin — last modified Apr 17, 2010 12:46 AM — [History](#)
Congratulations! You have successfully installed Plone.
[Viewable in presentation mode...](#)

If you're seeing this instead of the web site you were expecting, the owner of this web site has just installed Plone. Do not contact the Plone Team or the Plone mailing lists about this.

What you should do next

When you start exploring your newly created Plone site, please do the following:

- Make sure you are logged in as an admin/manager user. (You should have a Site Setup entry in the menu in the top right corner)
- Set up your mail server. (Plone needs a valid SMTP server to verify users and send out password reminders)
- Decide what security level you want on your site. (Allow self registration, password policies, etc)

What you should do next

When you start exploring your newly created Plone site, please do the following:

- Find out [What's new in Plone](#).
- Read the [documentation](#), especially the [Documentation for content editors and Server setup recommendations](#).
- Explore the [available add-ons](#) for Plone.
- Read and/or subscribe to the [support forums](#).

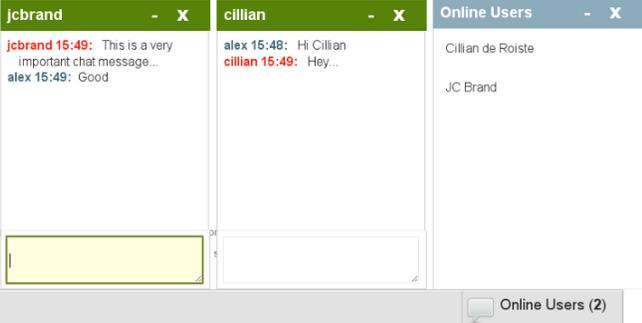
Make it your own

Plone has a lot of different settings that can be used to make it do what you want it to. Some examples:

- Try out a different theme, either pick from [the included ones](#), or one of the [available themes from plone.org](#). (Make sure you have the theme installed first)
- Decide what kind of workflow you want in your site. (The default is a website-centric workflow, if you want to use Plone as a content management system)
- By default, Plone uses a visual editor for content. (If you prefer text-based syntax and/or wiki syntax, you can change this in the Site Setup control panel)

jcbrand	cillian	Online Users
jcbrand 15:49: This is a very important chat message... alex 15:49: Good	alex 15:48: Hi Cillian cillian 15:49: Hey...	Cillian de Roiste JC Brand

Online Users (2)



INDICES AND TABLES

- *genindex*
- *modindex*
- *search*