Android jChat

User Guide

Contents

[Introduction 3](#_Toc204418835)

[Android jChat setup 3](#_Toc204418836)

[Setting the phone number 3](#_Toc204418837)

[Customizing the GPS location provider 3](#_Toc204418838)

[Launching JADE main container 4](#_Toc204418839)

[Using Android jChat 4](#_Toc204418840)

[Connecting to JADE 4](#_Toc204418841)

[Knowing jChat tabs 6](#_Toc204418842)

[The Contact tab 6](#_Toc204418843)

[The map tab 7](#_Toc204418844)

[Picking contacts 7](#_Toc204418845)

[Interacting with contacts 8](#_Toc204418846)

[Sending an SMS to one or more contacts 8](#_Toc204418847)

[Calling a contact 9](#_Toc204418848)

[Starting a chat session 9](#_Toc204418849)

[Using the status bar to browse chat sessions 10](#_Toc204418850)

[Closing a chat session 10](#_Toc204418851)

[FAQ 11](#_Toc204418852)

# Introduction

jChat application is an agent based IM-style application for Android , based on JADE (<http://jade.tilab.com>).

It allows the user to communicate with any other contact having the same software running and connected to the same JADE platform. Users can send text IM in a chat like way, call a contact by phone or even send an SMS to a contact.

Moreover the position of each contact can be seen on the map at any time and it is continuously updated.

# Android jChat setup

In order to be able to start and use the jChat application, a few configuration steps are required.

## Setting the phone number

jChat application uses the phone number as the agent ID of each contact. Unfortunately, the emulator phone number is not easily customizable yet.

As a workaround, the phone number can be set by changing the value in local.prop that can be found in the cfg folder. The local.prop file should then be copied on the emulator in /data with the following command:

adb push **<path-to-file>**\local.prop /data

Please note that, after copying the file, it is necessary to reboot the emulator to see the changes.

If the local.prop file cannot be found on the emulator, the jChat application shall use a random ID for the agent.

## Customizing the GPS location provider

jChat allows to a user to see both his position and his contact’s on a map. To do so, a mocked GPS location provider is used. Android provides a simple way of mocking GPS: each location provider is mapped to a directory on the emulator inside /data/misc/location. Each location provider directory contains a set of files that specifies different positions that are read and fired over time to simulate a real GPS device.

Android provides a gps directory with a readymade path, but any user is totally free to provide its own path. Currently jChat provides some tracks in KML format that can be used as an alternative.

It is possible changing the default track by creating a new folder under /data/misc/location (let’s call it *mygps*) and copying the KML file there, together with the provided *location* and *properties* files.

At the end, all these three files should be inside the mygps folder.

Now, to make the application aware of what GPS location provider should be used, the strings.xml file (found in res/values subfolder) should be modified. In particular the location\_provider\_name variable should be set with the name of the location provider to be used (mygps in this case).

Again, note that the emulator must be restarted in order to find the newly added location provider.

## Launching JADE main container

To be able to run jChat, a JADE Leap main container must be launched.

A JADE Leap container MUST be running when jChat is launched, otherwise the application will hang.

These are the steps that should be executed:

1. Download the last JADE-Leap sources from <http://jade.tilab.com>
2. Unpack the sources somewhere
3. Build JADE Leap for J2SE following the instruction provided with JADE sources. If the build is successful, a JadeLeap.jar file is available.
4. Launch a JADE Leap main container, together with JADE RMA GUI issuing the following command

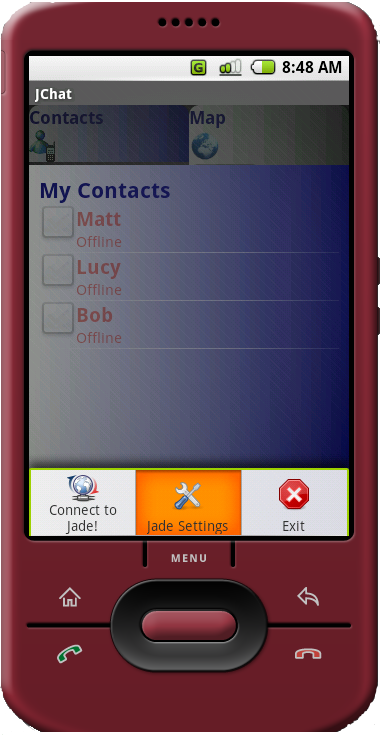
java -cp <PATH\_TO\_JADE\_LEAP>\JadeLeap.jar -Djade\_domain\_df\_autocleanup=true jade.Boot -gui

Please note that the option -Djade\_domain\_df\_autocleanup=true is necessary to correctly cleanup the DF agent.

# Using Android jChat

## Connecting to JADE

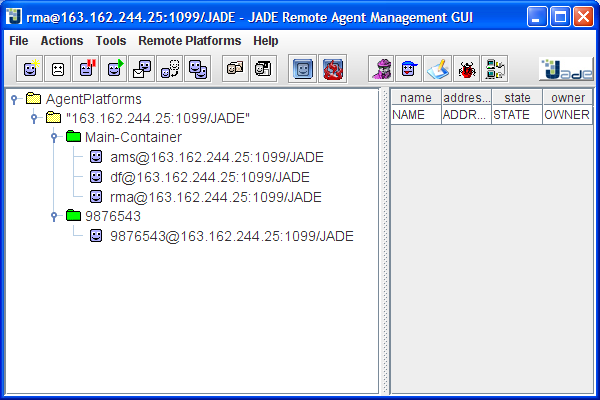
The first step for using jChat is the connection to a JADE Leap main container running on a host machine (local or remote). The default host machine and the port for the main container are specified on strings.xml file during compilation (this is the default value).



To change these settings perform the following steps (see Figure 1):

1. Press the ***MENU*** key on the emulator
2. Choose ***Jade Settings***
3. Insert the needed settings (***hos****t* and ***port***). Remember that the localhost address on Android emulator is 10.0.2.2
4. Press ***Close*** button
5. Press the ***MENU*** key on the emulator
6. Choose ***Connect to Jade!***

If the connection to Jade platform is successful you should be able to see the new agent running on Android emulator into Jade Remote Agent Management GUI.



The name of the new agent should be the same as the phone number of the phone.

# 

# Knowing jChat tabs

The main screen is divided into two different tabs, providing contacts information in different visual ways.

## The Contact tab

## contacts-tab.png

The contact tab contains a list of all currently available contacts.

Any contact can be online (connected to our JADE Leap platform) or offline.

Online contacts are shown in blue, while offline contacts are shown in red.

For each online contact the distance in km between the phone owner and the contact is shown.

## The map tab

## map-mode.pngsatellite-mode.png

The map tab shows the position of each online contact on a map and continuously updates their position.

Each contact is visible as a pin of different colors: the phone owner can see his own position as a yellow pin, while others contacts are shown as blue pins.

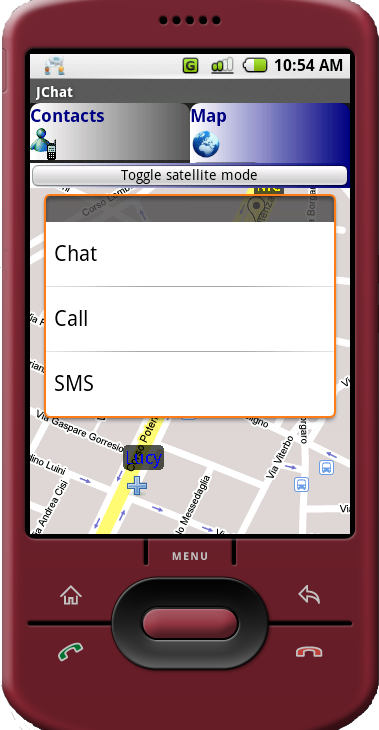
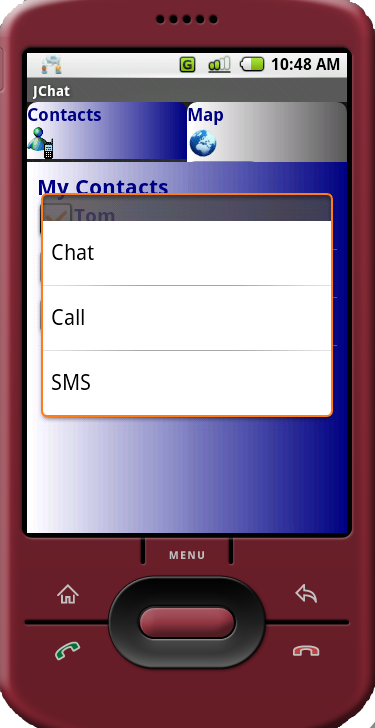
Maps can be seen both in traditional map mode and in satellite mode: the user is able to switch between the two modes using the switch button placed right above the map.

# Picking contacts

To be able to communicate with others, users must first select contacts (“picking”).

You can pick one or more contacts in both contact tab and map tab.

In contact tab, you have to click on the checkbox beside each contact. In map tab you simply click on the pin representation of the contact.



Please note that contact selection is local to the map (you cannot begin selecting contact in a tab, then switch to the other tab and continue selection).

After contacts selection, you can decide how to interact by pressing for some seconds the emulator selection button and choosing the desired option from the context menu that will popup.

# Interacting with contacts

Users can communicate with contacts in three ways: SMS, phone calls or chat sessions.

## Sending an SMS to one or more contacts

After choosing to send an SMS message, user will be redirected to the SMS sending screen. Obviously a message can be sent to more than one contact at the same time.



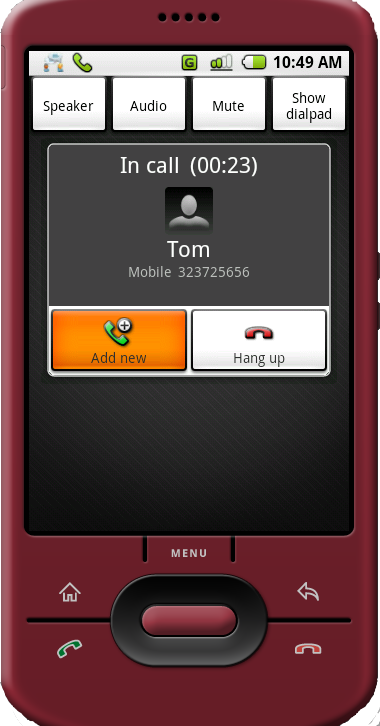
Here he will be able to enter the message text and to send it.

Message shall be split into chunks if too long and sent to the desired contact/s.

Please note that at the moment the emulator does not support SMS sending emulation, so nothing shall be received. The SMS sending process is mocked and it always succeeds.

## Calling a contact

If you choose to call a contact, the calling screen shall appear and the call shall start.



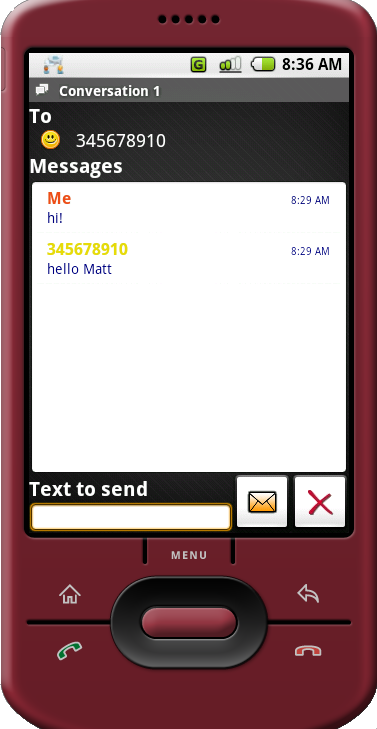
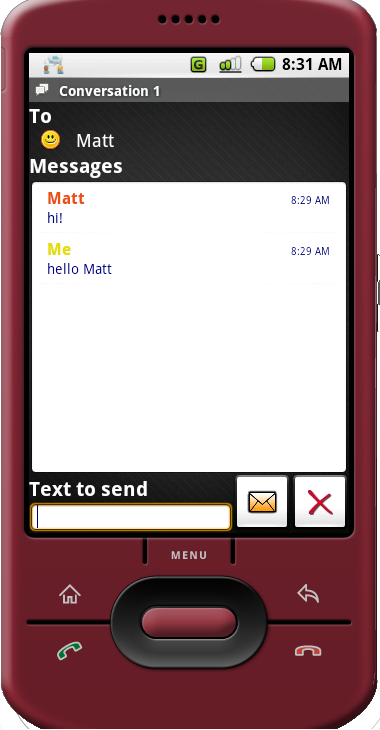
Obviously, the phone call is totally mocked.

If you try to call more than one contact at the same time (by selecting many contacts and choosing ***Call*** from context menu) a message shall appear warning that only one contact can be called at the same time.

## Starting a chat session

We define *chat session* as a group of two or more contacts talking together.

A chat session can be started by the phone owner (by selecting contacts and choosing ***Chat***from the context menu ) or by another contact that wants to talk with the phone owner himself.

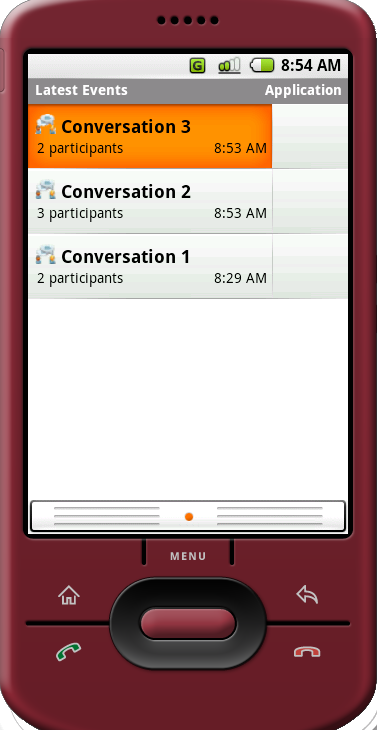
 

If the user chooses to start a chat session, the chat screen shall appear. In this screen you will be able to see the list of the other session participants, to enter a text message, to send it and to see a list of all the conversation messages with timings.

## Using the status bar to browse chat sessions

Whenever a chat message is received from another contact, a toast notification is shown and a persistent notification is added to the status bar. Also starting a new session adds a new persistent notification.

In this way the user can switch between many different ongoing conversations by using the status bar.



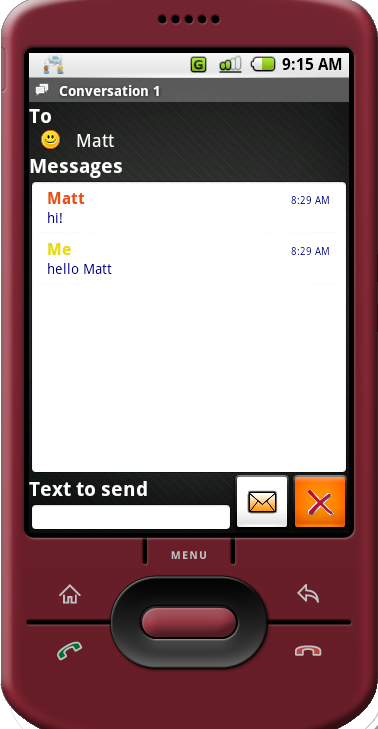
These are the required steps:

1. Drop down the Android status bar. The list of active sessions notifications shall appear. Please note that you’ll have a notification for each ongoing chat session, it does not matter who started the conversation.
2. Choose the session you’re interested in and click on its notification.
3. A new chat session shall start (or the session switches to the new one if a chat session was already opened)

Each time a conversation notification is selected the chat session is redrawn.

## Closing a chat session

To close a chat session, simply press the cross button next to the send button in the chat screen.



Please note that the emulator’s ***BACK*** key will simply close the chat screen but the session will remain open (all conversation messages will be kept) and the user will be able to restore it by choosing the notification on the status bar.

After choosing the cross button the session will be removed and all previously sent messages will be lost.

# FAQ

*Q: JADE complains that two agents cannot be created with the same id. I can I have different instances of the emulator with different phone numbers?*

A: The easiest way to have two different emulators with different phone numbers is two launch two emulators with different image files. To do so, go into the emulator image folder (default on windows is Documents and Settings\<USERNAME>\Local Settings\Application Data\Android ), copy the userdata.img file and rename it (let’s say to userdata2.img), then launch the emulator with the following command:

emulator –data <PATH\_TO\_IMAGE>\userdata2.img

Now you have a totally independent instance of the emulator and you can copy another local.prop file in /data to set a different phone number.

*Q: I launch the application but when I choose Connect to Jade from the menu, the application hangs. Why?*

A: Probably you forgot to launch a JADE Leap main container. A main container is needed for the application to start.

*Q: When I start the application it suddenly terminates saying “Unable to retrieve the given location provider!”*

A: This means that the directory for the location provider cannot be found. Try to change the strings.xml file to the name of the location provider directory in /data/misc/location

*Q: Sometimes I got a message saying “Error sending broadcast null” when closing the application. Why?*

A: This seems to be a known bug of Android emulator when shutting down the location update.