Site Instant Messenger 4.6.0 installation instructions

Eugeny N Dzhurinsky, JDevelop.com, 2003-2005

31st January 2005

1 Pre-requirements

The server to install messenger on requires the JDK 1.3.1 or higher to be installed.

2 Package contains

1. Server package

   (a) server.jar - main archive, containing the server classes
   (b) mysql.jar / Opta.jar - the JDBC driver to connect the server with mysql / MS SQL (can be replaced with JDBC driver for specific system)
   (c) xmlrpc.jar - the package, which allows to access some IM-server features with XML-RPC protocol
   (d) runtime.props - the main server configuration file
   (e) serverstart.sh / serverstart.bat - SH / BAT script to start server
   (f) recycle.sh - cron job to auto-restart server if it down for some reason (server restart, killed or so). Uses serverstart.sh to start server. **Unix-only!**
   (g) wrapper.cgi or wrapper.php - the script, which wraps the requests for server over HTTP protocol. This allows to use messenger by customers behind firewalls or proxy servers.

2. Client package

   (a) client.jar - the JAR archive, which contains the applet
   (b) skin.jar or skin.zip - the skin file to be used with messenger
   (c) applet.php - sample PHP page, which contains the sample applet tag

3 IM Server configuration

The IM server configuration is placed in the **runtime.props** file. This file has set of configuration options for database, timeouts, etc. Here is the description of options below.

- **xml-rpc.port -** the port, on which XML-RPC module listing for the requests
- **driver** - the name of JDBC driver (don't change if you not sure what are you doing)
- **dbprovider** - the name of using RDBMS (don't change if you not sure what are you doing)
- **dbhost** - the host of the database server
- **dbport** - the port of database server
- **dbname** - name of the database to use
- **dbuser** - user to access database
- **dbpassword** - the password to access database

- **AUTH_TABLE_NAME** - the name of table, where user login and password fields are

- **AUTH_TABLE_LOGIN** - name of field in table, which contains the login of user

- **AUTH_TABLE_PASSWORD** - the name of field which contains the password of user

- **AUTH_TABLE_ID** - the name of field, which contains the user ID (may have numeric or string type)

- **PROFILE_TABLE_NAME** - the name of table, which contains the profile information

- **PROFILE_TABLE_USERNAME** - the name of field, which contains the username to display. In this parameter you can use the SQL statements, like concat(firstname,' ',lastname) to get the complicated user names from several fields

- **PROFILE_TABLE_ID** - the user ID field name. This is used as foreign key to table AUTH_TABLE_NAME.

- **PROFILE_TABLE_SEX** - the name of the field, which contains the gender information. 1 means male, any other value - female. You can use the SQL statements like if(gender='male',1,2) as value for this parameter.

- **PROFILE_TABLE_AGE** - the optional information, which is used to provide additional information, like age of user, or state, or something else. You can use SQL statements here as well.

- **PROFILE_TABLE_PAID** - the field name, which shows if user is a paid member and can start chat. Users with value 1 can start the chat, users with any other value only can accept chat, can't to initiate it by theirselves. You can use the SQL statements here.

- **IGNORE_TABLE_NAME** - the name of table, where blocked list for user is stored.

- **IGNORE_TABLE_UID** - the ID from user. This is foreign key to AUTH_TABLE_NAME.

- **IGNORE_TABLE_IGNORED** - the name of field, which contains the list of IDs for ignored users, separated by commas. This field must have the type of VARCHAR, but MEDIUMTEXT is even better.

- **BUDDY_TABLE_NAME** (optional) - the name of table, where buddy list is stored.

- **BUDDY_TABLE_ID** (optional) - the name of field, which contains the ID of user. This is foreign key for AUTH_TABLE_NAME

- **BUDDY_TABLE_BUDDY** (optional) - the ID of buddy, foreign key for AUTH_TABLE_NAME

- **timeout** - the parameter, which defines the timeout*30 seconds, after which user will be logged out if he didn't send any requests to a server (no requests for new messages, no requests for the buddy list and no requests for the sending new message)

- **initial_timeout** - the timeout value which user gets once he's logged in.

- **acceptedHosts** - the parameter, which controls the hosts name, where applet can be integrated in the page. Names of hosts separated by ',' , ' ' , ';' . This will avoid some to use your messenger client on another site, if you didn't specify this site in the list of accepted hosts.

Now about the **recycle.sh**. You will need to change the **SIM_HOME** variable so it will points to directory, where the **recycle.sh** is located, and **PORT** variable, which describes the port on which the IM server will be started.

4 IM client configuration.

Take a look at the **applet.php** file. There are following parameters:

1. **login** - the login of current logged user

2. **password** - the password of user

3. **port** - the port to connect to when direct connections are used

4. **prefix** - the relative URL of the wrapper script on your server

5. **profileURL** - the absolute URL to script, which returns the name of user image. This script will returns the single line, like this
http://yourserver.com/images/profile/userimage.jpg
In this case the applet will get the image from server and include it in the message window.

Now about the **skin.zip** file. This file contains the skin. Ski is set of images, sounds, and it has it's own configuration file called **layout.conf**. Most of parameters there are self-explaining, but several are not. Here is the list:

- **smileys_per_row** - the parameter, which shows how many smileys will be shown per row in a dropdown list for smileys.

- **buddiesPerRow** - the parameter, which controls how many columns will have the buddy list

- **profileURL** - the URL, which will be opened when user icon in buddy list or user image in message window will be clicked. This URL will looks like
http://yourserver.com/profile.php?userid=
applet will append the user ID to this url and open it in a new window

- **profileURLTarget** - the target window where profile will be opened

- **client_pull_delay** - the delay in client_pull_delay*5 seconds, which sets the delay between requests to server

- **client_ignore_timeout** - in seconds, shows how many time the messages from user, which just was "declined" to start chat, will not be accepted from the current user.

- **maxsessions** - controls how many concurrent chat windows will have user at the same time

5 Installation

5.1 Linux/FreeBSD/Solaris

Unpack the ZIP file, and place the **server** directory somewhere on server, where you will use it, and contents **client** directory move to a directory, accessible by your customers from a WEB. Configure the **runtime.props** to match your system and start the server with **recycle.sh**. Then place the client applet tag to some of your pages, create your own **skin.zip** and enjoy the application.

5.2 Windows

Unpack the ZIP file, and place the **server** directory somewhere on server, where you will use it, and contents **client** directory move to a directory, accessible by your customers from a WEB. Configure the **runtime.props** to match your system and start the server with **serverstart.bat**. Then place the client applet tag to some of your pages, create your own **skin.zip** and enjoy the application.

6 Troubleshooting

There will be no problems installing and deploying the messenger on most systems, if you have one - please report to **sim_troubles@jde** you will get response and support within 24 hours.