



Installation Guide

Contents

1	Development Environment	4
1.1	Prerequisites	4
1.1.1	Java	4
1.1.2	MySQL	4
1.1.3	Openfire	4
1.1.4	Virgo WebServer	5
1.1.5	WireShark	5
1.2	Post Installation Steps	5
1.2.1	Set environment variables	5
1.3	Directory Layout	6
1.4	Environment Settings	6
1.5	Ant	7
1.5.1	Installation	7
1.5.2	Environment Settings	7
1.6	Findbugs	7
1.6.1	Installation	7
1.6.2	Environment Settings	8

1.7	Ruby	8
1.7.1	Installation	8
1.8	Build System	8
1.9	Spring DM Server	9
1.10	SpringSource Tool Suite	9
1.11	Environment	10
1.12	Openfire Database Installation	11
1.13	OpenFire 3.6.4	11
1.14	OpenFire Configuration	11
1.15	OpenFire Demo Configuration	18
1.16	Spring DM Server Configuration	21
1.17	IM Client - Human Buddy	21
1.18	Jnodes Pre-Canned Application Demo	23
1.18.1	Initial Run of the Demo	24
1.18.2	Stopping the Spring DM Server	25
1.18.3	Post-Initial Demo	25
1.19	Development	26
1.19.1	Configuration of STS	26
1.20	Useful Links	27
1.21	Tips Jar	27
1.21.1	Logging	27
1.21.2	Eclipse wc-props warnings	28

Chapter 1

Development Environment

1.1 Prerequisites

1.1.1 Java

Java SE 6 or later is required. The setup and installation of Java SE 6 is typically operating system specific, so consult your OS provider.

1.1.2 MySQL

MySQL is required to support the installation of the Openfire XMPP server. Download and Installation information is available at :

<http://www.mysql.com>

1.1.3 Openfire

The Openfire XMPP server is required to support the construction of virtual networks within the TINOS platform. Download and Installation information is available at :

<http://www.igniterealtime.org/projects/openfire>

1.1.4 Virgo WebServer

TINOS is loaded and executed within the Virgo WebServer platform. It is recommended that the Virgo WebServer is installed in the DEV_HOME directory as TINOS users will need to interact with this server directly in order to load and execute TINOS nodes. Download and installation information is available at :

<http://www.eclipse.org/virgo/download>

1.1.5 WireShark

WireShark is a tool that is used to examine TINOS network trace logs. Download and installation information is available at :

<http://www.wireshark.org>

1.2 Post Installation Steps

1.2.1 Set environment variables

JAVA_HOME

TINOS uses the JAVA_HOME environment variable to locate the java executable. Configure this environment variable to point to the home directory of the Java 5 or 6 installation on your computer.

SERVER_HOME

As a convenience it is recommended that you create an environment variable that points to the Virgo Web Server installation directory. Note that the Virgo Web Server does not required that such an environment variable has been set. This variable may have any name of your choosing. The following documentation assumes that the variable is named SERVER_HOME.

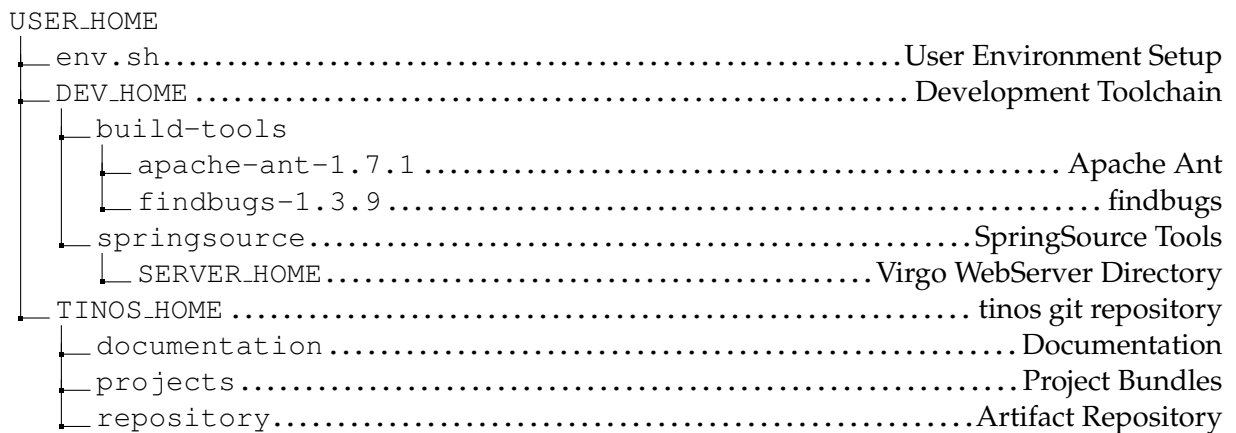


Figure 1.1: High Level Installation Layout

DEV_HOME

As a convenience it is recommended that you create an environment variable that points to the directory that contains the development environment tooling. This variable may have any name of your choosing. The following documentation assumes that the variable is named DEV_HOME.

TINOS_HOME

As a convenience it is recommended that you create an environment variable that points to the directory that contains the tinos git repository. The following documentation assumes that the variable is named TINOS_HOME.

1.3 Directory Layout

1.4 Environment Settings

Generally we attempt to isolate environments from each other as a matter of course. This supports the idea of multiple environments being available for a user dependent upon what they want to do.

To support this idea, all the enviroment settings for TINOS development are written into the file env.sh in the development directory. This file can be sourced within the shell of the user to append the relevant settings to their default shell.

In doing so, the user tailors this shell for the TINOS development environment and platform. Listings of the relevant settings under the various headings of the installation are provided just to help join the dots for the reader. A complete env.sh file is also listed at the end of the document.

1.5 Ant

The current version of Ant is apache-ant-1.7.1.

1.5.1 Installation

Task Download the Apache Ant from Apache.Org.

URL <http://ant.apache.org/bindownload.cgi>

```
$ cd $HOME/development
```

```
$ mkdir build-tools
```

```
$ cd build-tools
```

```
$ unzip $HOME/installation/packages/apache-ant-1.7.1-bin.zip
```

Note This will extract the ant into the directory: apache-ant-1.7.1

1.5.2 Environment Settings

Ensure you add the ANT_OPTS environment variable, it is needed to workarounds a Java JVM issue in relation to the churn of the PERM cache.

```
# Ant Settings
export ANT_HOME=$DEV_HOME/build-tools/apache-ant-1.7.1
export ANT_OPTS="-Xms64m -Xmx512m -XX:PermSize=128m -XX:MaxPermSize=756m"
export ANT_EXEC=$ANT_HOME/bin
```

1.6 Findbugs

The current version of Findbugs is findbugs-1.3.9.

1.6.1 Installation

Task Download the findbugs from

URL <http://findbugs.sourceforge.net/downloads.html>

```
$ cd $HOME/development/build-tools
```

```
$ tar xzvf $HOME/installation/packages/findbugs-1.3.9.tgz
```

Note This will extract the ant into the directory: findbugs-1.3.9

1.6.2 Environment Settings

```
export FINDBUGS_HOME=$DEV_HOME/build-tools/findbugs-1.3.9
export PATH=$FINDBUGS_HOME/bin:$PATH
```

1.7 Ruby

The current version of Ruby is 1.8.7, with the "choices" and "xml-simple" rubygems. In previous installations, this part of the toolchain has been built from source code but now it has standardised enough to use the OS provided packages.

1.7.1 Installation

```
$ export GEM_HOME=$HOME/.gem
```

Task Install "choice" gem (Project Template Tool Requirement)

```
$ gem install $HOME/installation/packages/choice-0.1.3.gem
```

Task Install "xml-simple" gem (Project Template Tool Requirement)

```
$ gem install $HOME/installation/packages/xml-simple-1.0.12.gem
```

1.8 Build System

Install the core Build System files.

Task Install Core Build System Files

URL Contact TSSG - Version Controlled Files.

```
$ cd $HOME/development
```

```
$ tar xzvf $HOME/installation/packages/tssg-build-core.tgz
```

Note This will extract into the following directory : tssg-build-core

Joy This directory provides the Template Generators/Ivy for development.

1.9 Spring DM Server

Install the Spring DM Server.

Task Download SpringSource DM Server.

URL <http://www.springsource.com/download/dmserver>

```
$ cd $HOME/development
```

```
$ mkdir springsource
```

```
$ cd springsource
```

```
$ unzip $HOME/installation/packages/springsource-dm-server-2.0.0.RELEASE.zip
```

Note This will extract into the following directory : springsource-dm-server-2.0.0.RELEASE

1.10 SpringSource Tool Suite

Install the Spring Tool Suite (Branded *Eclipse*), it is invoked on the command line as "STS".

Task Download SpringSource Tool Suite.

URL <http://www.springsource.com/products/springsource-tool-suite-download>

```
$ cd $HOME/development/springsource
```

```
$ unzip $HOME/installation/packages/sts-Jan28.tgz
```

Note This will extract into the following directory : sts-2.3.0.RELEASE

Note Workaround for STS (Eclipse 3.5 variants) support on linux

```
$ export GDK_NATIVE_WINDOWS=1
```

Note This tarball was constructed from my installation based off the official 2.3.0.RELEASE but with plugins for Spring DM Server 2.0 pulled directly from the nightly builds.

1.11 Environment

All of the environmental settings from above combined and integrated. Typically this file is imported into the users shell whenever they wish to use the environment.

Task Append the CBNE Environment settings.

```
$ cd development
```

```
$ ../env.sh
```

Joy Ready to rock and roll!.

```
# Development Environment Settings
export DEV_HOME=$HOME/development

# Java JDK/JRE
export JAVADEV_HOME=$DEV_HOME/java-sdk
export JAVA_HOME=$JAVADEV_HOME/jdk1.6.0_17
export JRE_HOME=$JAVADEV_HOME/jdk1.6.0_17/jre

# Ruby Settings
export GEM_HOME=$HOME/.gem
export RUBYOPT=rubygems

# Ant Settings
export ANT_HOME=$DEV_HOME/build-tools/apache-ant-1.7.1
export ANT_OPTS="-Xms64m -Xmx512m -XX:PermSize=128m -XX:MaxPermSize=756m"
export ANT_EXEC=$ANT_HOME/bin

# FindBugs
export FINDBUGS_HOME=$DEV_HOME/build-tools/findbugs-1.3.9

# Spring DM Server
export SPRING_DM=$DEV_HOME/springsource/springsource-dm-server-2.0.0.RELEASE

#STS Settings
export STS_HOME=$DEV_HOME/springsource/sts-2.3.0.RELEASE
# Workaround for STS (Eclipse 3.5 variants) support on linux
export GDK_NATIVE_WINDOWS=1

# Setup Path(in the document)
export PATH=$JAVA_HOME/bin:$ANT_EXEC:$FINDBUGS_HOME:$PATH
export PATH=$SPRING_DM/bin:$STS_HOME:$PATH
```

```
# Alias
export EDITOR=vim
alias vi='vim'
```

1.12 Openfire Database Installation

Before the installation of Openfire, a database user and database for the Openfire server must be created.

Task Create database user and database

Note Create a database

```
$ mysql -u root -p
```

```
mysql$ create database openfire character set utf8;
```

Note Create a user

```
mysql$ grant all on openfire.* to openfire@localhost identified by 'openfire';
```

```
mysql$ commit; exit;
```

1.13 OpenFire 3.6.4

Install the OpenFire server as instructed on their website. Once this is completed, start the server and then follow the instructions below to configure the server.

1.14 OpenFire Configuration

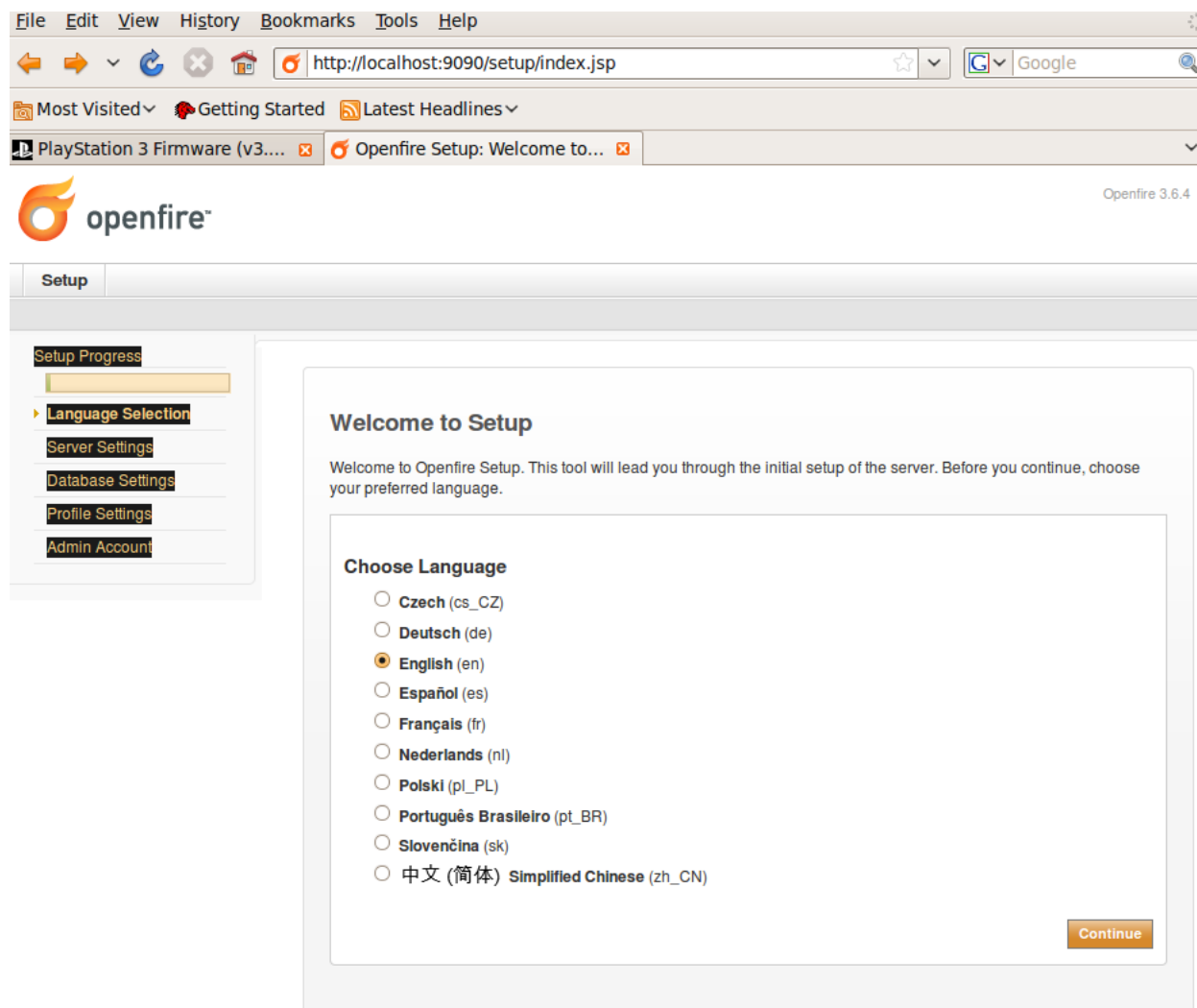
In order to complete the configuration of the OpenFire server, a web browser must be used to step through the server setup screens.

Task Configure the OpenFire server with a web browser.

Browser Enter the following URL to start the configuration.

URL <http://localhost:9090>

Browser Select the English language and click Continue.



Built by [Jive Software](#) and the [IgniteRealtime.org](#) community

Task Configure the Server Settings

Browser Enter "localhost" as the domain and leave the other settings unchanged.

Browser Click Continue

The screenshot shows a web browser window with the URL `http://localhost:9090/setup/setup-host-settings.jsp`. The browser's address bar and tabs are visible. The Openfire 3.6.4 logo is in the top left corner. The main content area is titled "Setup" and shows a "Setup Progress" bar with a green segment. Below the progress bar is a list of setup steps: "Language Selection" (checked), "Server Settings" (selected), "Database Settings", "Profile Settings", and "Admin Account". The "Server Settings" section is expanded, showing a "Server Settings" heading and a note: "Below are host settings for this server. Note: the suggested value for the domain is based on the network settings of this machine." The settings form includes three input fields: "Domain" (containing "localhost"), "Admin Console Port" (containing "9090"), and "Secure Admin Console Port" (containing "9091"). Each field has a help icon (question mark). A "Continue" button is located at the bottom right of the form.

File Edit View History Bookmarks Tools Help

http://localhost:9090/setup/setup-host-settings.jsp

Most Visited Getting Started Latest Headlines

PlayStation 3 Firmware (v3... Openfire Setup: Server Setti...

openfire™ Openfire 3.6.4

Setup

Setup Progress

Language Selection

Server Settings

Database Settings

Profile Settings

Admin Account

Server Settings

Below are host settings for this server. Note: the suggested value for the domain is based on the network settings of this machine.

Domain: localhost

Admin Console Port: 9090

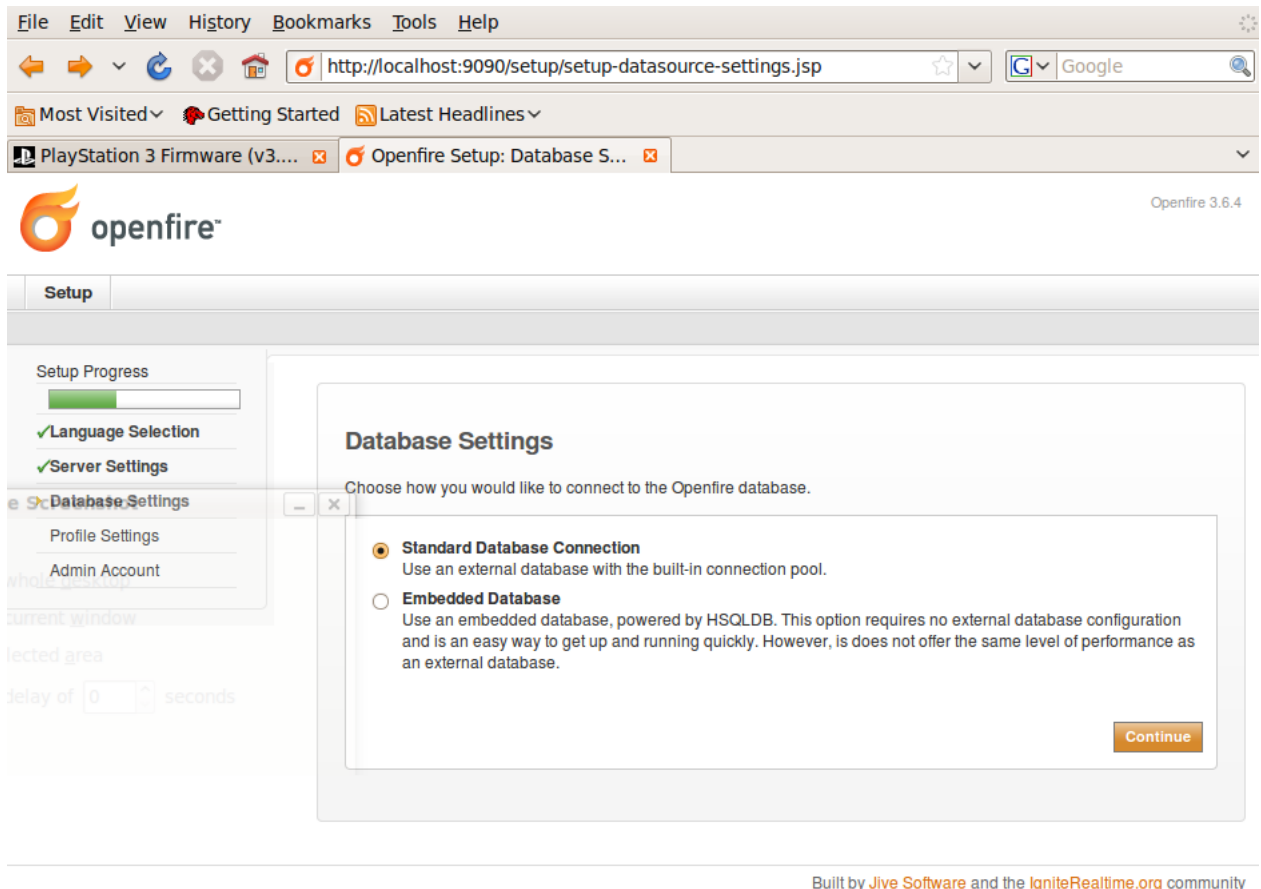
Secure Admin Console Port: 9091

Continue

Task Configure the Database Settings

Browser Select "Standard Database Connection"

Browser Click Continue



Task Configure the Standard Database Settings

Browser Select "MySQL" in the Database Driver presets.

Browser Edit the Database URL to "jdbc:mysql://localhost:3306/openfire"

Browser Edit the Username to "openfire"

Browser Edit the Password to "openfire"

Browser Click Continue

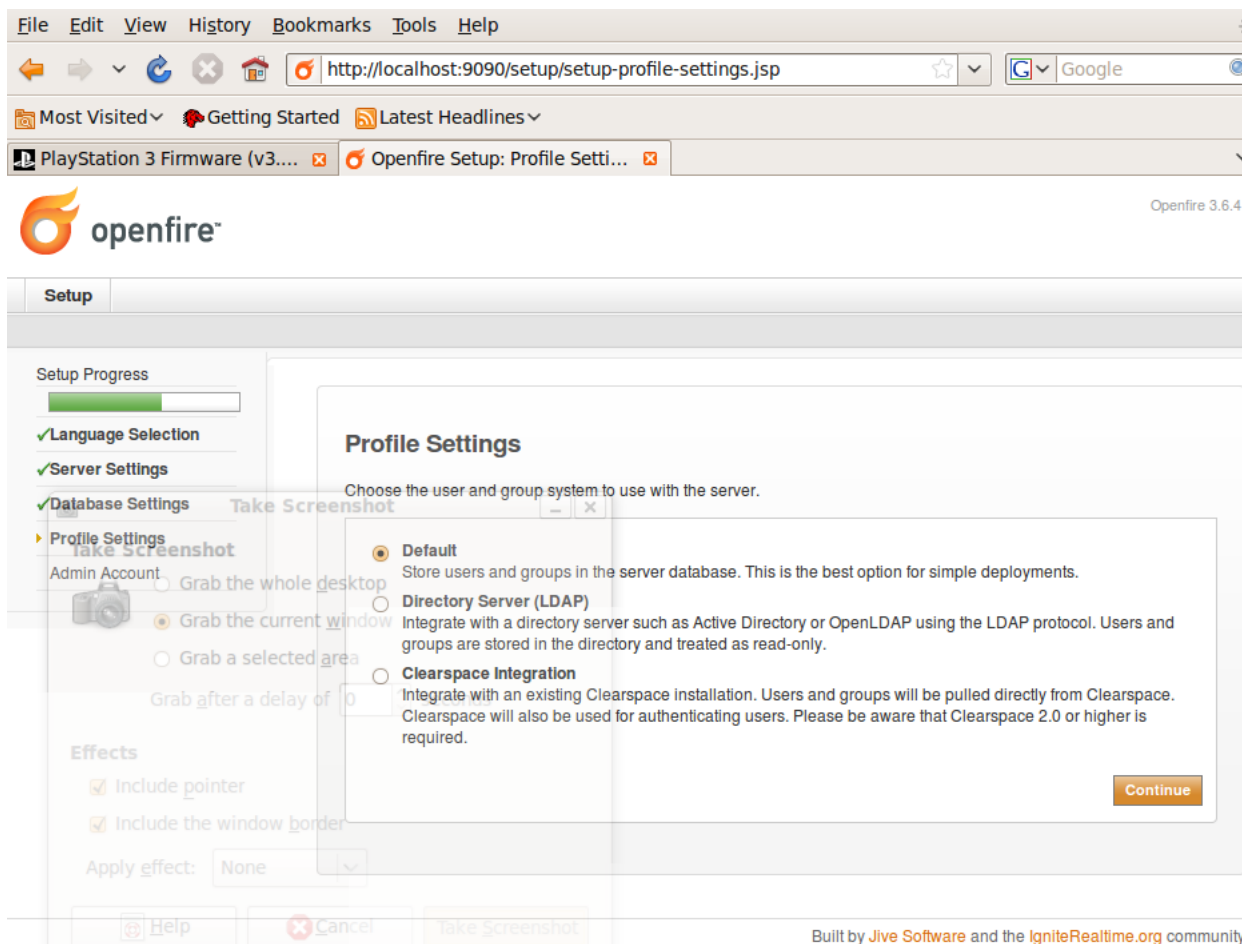
Note The database settings reflect those configured earlier.

The screenshot shows a web browser window with the address bar displaying `http://localhost:9090/setup/setup-datasource-standard.jsp`. The browser's address bar also shows a search engine (Google) and a star icon. The browser's tabs include "PlayStation 3 Firmware (v3....", "Openfire Setup: Database S...", and "Openfire 3.6.4". The Openfire logo is visible in the top left corner of the page, and the version "Openfire 3.6.4" is in the top right corner. The page title is "Setup". The left sidebar shows the "Setup Progress" bar with a green progress indicator. The sidebar menu includes "Language Selection", "Server Settings", "Database Settings" (which is highlighted), "Profile Settings", and "Admin Account". The main content area is titled "Database Settings - Standard Connection". It contains a paragraph: "Specify a JDBC driver and connection properties to connect to your database. If you need more information about this process please see the database documentation distributed with Openfire." Below this is a "Note" stating: "Database scripts for most popular databases are included in the server distribution at [Openfire_HOME]/resources/database." The form fields are: "Database Driver Presets:" (a dropdown menu showing "MySQL"), "JDBC Driver Class:" (a text input field containing "com.mysql.jdbc.Driver"), "Database URL:" (a text input field containing "jdbc:mysql://localhost:3306/openfire"), "Username:" (a text input field containing "openfire"), "Password:" (a password input field with 10 dots), "Minimum Connections:" (a text input field containing "5"), "Maximum Connections:" (a text input field containing "25"), and "Connection Timeout:" (a text input field containing "1.0" followed by "Days"). A "Continue" button is located at the bottom right of the form. A note at the bottom of the form states: "Note, it might take between 30-60 seconds to connect to your database."

Task Configure the Profile Settings

Browser Select "Default"

Browser Click Continue



Task Configure the Administrator Settings

Browser Enter an Admin Email Address - You can pick whatever you want.

Browser Click "Skip this step"

Note This is only to complete the installation configuration. In the next steps a database import with change this value to the default.

File Edit View History Bookmarks Tools Help

http://localhost:9090/setup/setup-admin-settings.jsp

Most Visited Getting Started Latest Headlines

PlayStation 3 Firmware (v3.... Openfire Setup: Administrat...

openfire™ Openfire 3.6.4

Setup

Setup Progress

Language Selection

Server Settings

Database Settings

Profile Settings

Admin Account

Take Screenshot

Grab the whole desktop

Grab the current window

Grab a selected area

Grab after a delay of 0 seconds

Effects

Include pointer

Include the window border

Apply effect: None

Administrator Account

Enter settings for the system administrator account (username of "admin") below. It is important to choose a password for the account that cannot be easily guessed -- for example, at least six characters long and containing a mix of letters and numbers. You can skip this step if you have already setup your admin account (not for first time users).

Admin Email Address: admin@example.com

A valid email address for the admin account.

New Password:

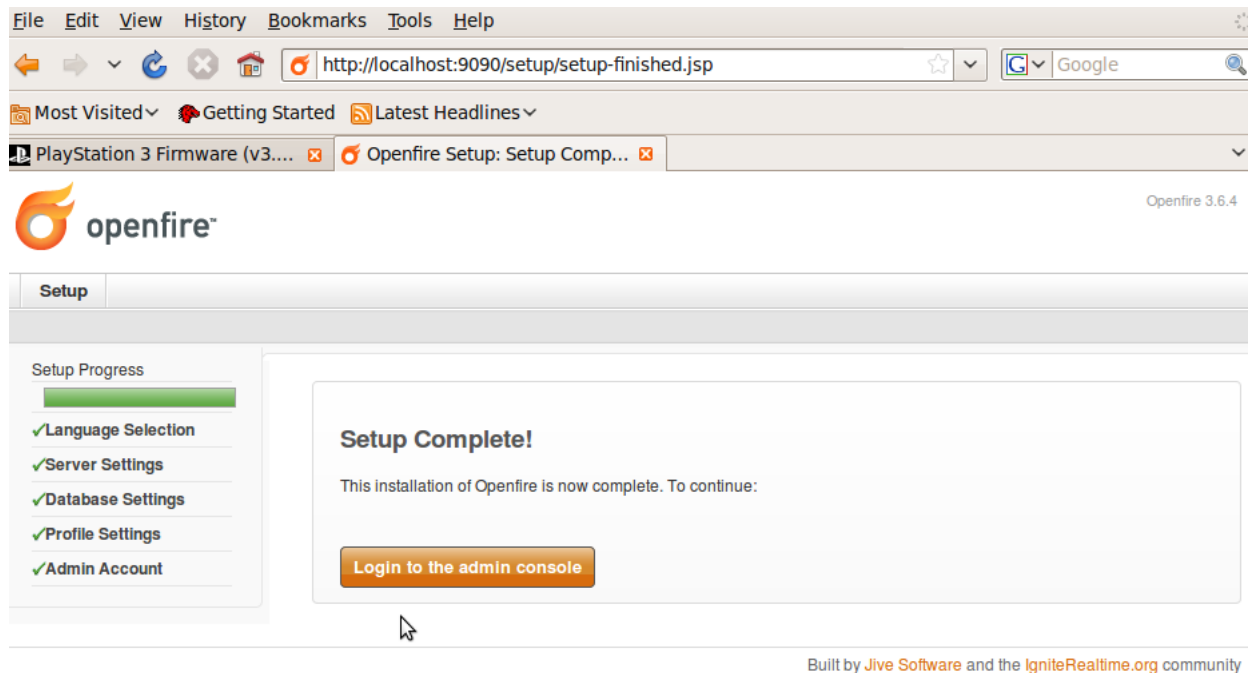
Confirm Password:

Skip This Step Continue

Built by Jive Software and the IgniteRealtime.org community

Task Setup Complete

Browser Setup should now be complete.



1.15 OpenFire Demo Configuration

The OpenFire server must be configured to match the requirements of the pre-canned demo applications. As such the existing database will be cleared and a valid configuration loaded in its place. This saves the long and tedious configurations required within OpenFire for all the users, groups and chatrooms.

Task Shutdown the OpenFire Server

```
$ cd /etc/init.d
```

```
$ sudo sh openfire stop
```

Task Clear the old database.

```
$ mysql -u root -p
```

```
mysql$ drop database openfire;
```

```
mysql$ create database openfire character set utf8;
```

```
mysql$ commit; exit;
```

Task Load the new database

```
$ mysql -u openfire -p openfire < $HOME/installation/deploy/db/openfire.db
```

Note You will be prompted for the password : "openfire"

Task Start the OpenFire Server

```
$ cd /etc/init.d
```

```
$ sudo sh openfire start
```

Task Login to the OpenFire Administration Console

Browser Goto URL: <http://localhost:9090>

Browser Username : "admin", Password : "12345"

Note You can change the password afterwards if you wish.

File Edit View History Bookmarks Tools Help

http://localhost:9090/index.jsp

Most Visited Getting Started Latest Headlines

PlayStation 3 Firmware (v3... Openfire Admin Console: Se...

openfire™ Openfire 3.6.4
Logged in as admin - [Logout](#)

Server Users/Groups Sessions Group Chat Plugins

Server Manager Server Settings Media Services

Server Information


System Properties
Language and Time
Clustering
Cache Summary
Database
Logs
Email Settings
Security Audit Viewer

Below you will find server information, ports being used and latest news about Openfire.

Server Properties

Server Uptime: Less than 1 minute -- started Sep 1, 2009 2:11:11 PM
Version: Openfire 3.6.4
Server Directory: /usr/share/openfire
Server Name: localhost

Environment

Java Version: 1.6.0_14 Sun Microsystems Inc. -- Java HotSpot(TM) 64-Bit Server VM
Appserver: jetty-6.1.x
Host Name: chimera
OS / Hardware: Linux / amd64
Locale / Timezone: en / Greenwich Mean Time (0 GMT)
Java Memory  65.68 MB of 442.69 MB (14.8%) used

Ignite Realtime News

[Tinder 1.1.0 has been released](#), Aug 27, 2009
[New Openfire monitoring plugin](#), Aug 18, 2009
[Introducing Tinder, an XMPP object implementation library](#), Jun 23, 2009
[Openfire 3.6.4 has been released](#), May 2, 2009
[Openfire 3.6.3 has been released](#), Jan 9, 2009
[Openfire 3.6.2 has been released](#), Nov 22, 2008

Server Ports

Interface	Port	Type	Description
All addresses	5222	Client to Server	The standard port for clients to connect to the server. Connections may or may not be encrypted. You can update the security settings for this port.
All addresses	5223	Client to Server	The port used for clients to connect to the server using the old SSL method. The old SSL method is not an XMPP standard method and will be deprecated in the future. You can update the security settings for this port.
All addresses	9090	Admin Console	The port used for unsecured Admin Console access.
All addresses	9091	Admin Console	The port used for secured Admin Console access.
All addresses	7777	File Transfer Proxy	The port used for the proxy service that allows file transfers to occur between two entities on the XMPP network.
All addresses	7070	HTTP Binding	The port used for unsecured HTTP client connections.
All addresses	7443	HTTP Binding	The port used for secured HTTP client connections.
	3478		

Done

1.16 Spring DM Server Configuration

Task Configure the Spring DM Server

```
$ cd $HOME/development/springsource/spring-dm-server-2.0.0.RELEASE/configs
```

Task Configure a local bundle repository for build artifacts.

```
$ cp $HOME/installation/deploy/config/com.springsource.repository.properties .
```

Task Configure logging/trace for the Jnode applications

```
$ cp $HOME/installation/deploy/config/serviceability.xml .
```

Task Add support bundles

```
$ cd $HOME/development/springsource/spring-dm-server-2.0.0.RELEASE/repository/usr
```

```
$ cp $HOME/installation/deploy/bundles/* .
```

Task Add the Plan files

```
$ cd $HOME/development/springsource/spring-dm-server-2.0.0.RELEASE
```

```
$ cp $HOME/installation/deploy/plans/* .
```

Task Setup the local build artifacts repository

```
$ cd $HOME
```

```
$ mkdir jars
```

```
$ cd jars
```

```
$ cp $HOME/installation/deploy/jars/* .
```

1.17 IM Client - Human Buddy

Once configured you can use the IM Client to visually see the presence of the nodes within the demo networks, as well as sit in the networks (via chatrooms) and see the interactions. Super cheap and cheerful GUI.

Task Configure Pidgin Client

GUI Start Pidgin; (Applications > Internet > Pidgin Internet Messenger)

Pidgin Add Account

Pidgin Basic Tab > Protocol: XMPP, User: Human, Domain: localhost, Password: Human, Remember Password: Tick.

Pidgin Advanced Tab> Connect Server: localhost.

Pidgin Click Add.

The image displays two side-by-side screenshots of the Pidgin account creation dialog box, showing the 'Basic' and 'Advanced' tabs.

Left Screenshot (Basic Tab):

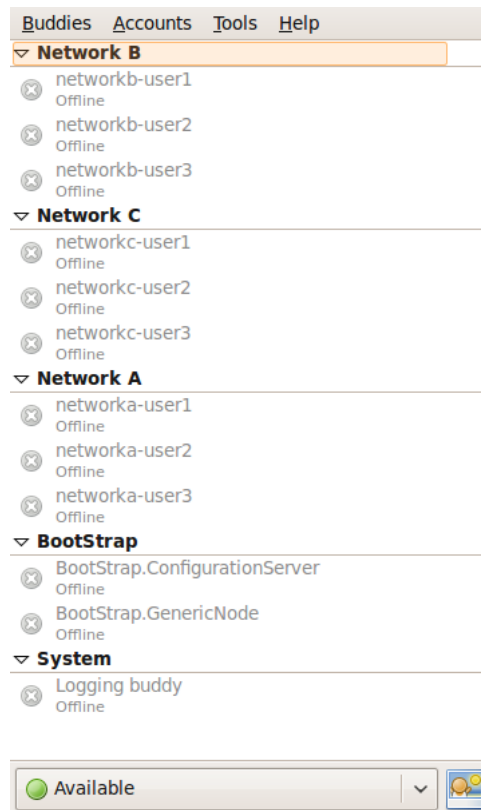
- Login Options:**
 - Protocol: XMPP (dropdown)
 - Username: human
 - Domain: localhost
 - Resource: (empty)
 - Password: (masked with dots)
 - ☒ Remember password
- User Options:**
 - Local alias: (empty)
 - ☐ New mail notifications
 - ☐ Use this buddy icon for this account: (with a small icon and a 'Remove' button)
- ☐ Create this new account on the server
- Buttons: Cancel, Add

Right Screenshot (Advanced Tab):

- XMPP Options:**
 - ☐ Require SSL/TLS
 - ☐ Force old (port 5223) SSL
 - ☐ Allow plaintext auth over unencrypted streams
 - Connect port: 5222
 - Connect server: localhost
 - File transfer proxies: proxy.jabber.org
 - ☒ Show Custom Smileys
- Proxy Options:**
 - Proxy type: Use GNOME Proxy Settings (dropdown)
- ☐ Create this new account on the server
- Buttons: Cancel, Add

Task Ensure you can see Offline Buddies

Pidgin Buddies > Show > Offline Buddies (Tick/Enable)



1.18 Jnodes Pre-Canned Application Demo

For the first initial loading of the demo - the following actions must be performed in exact sequence. This is primarily to make life easier for the person giving / viewing the demo as everything will start in an order that will match a presentation.

Task Start the Spring DM Server

```
$ . $HOME/development/env.sh
```

```
$ cd $SPRING_DM/bin
```

```
$ ./startup.sh
```

Note Wait for this to complete.

If you are doing a demo - it is most useful to have the IM Client open during the demo as you will be able to watch the bootstrap in progress (via presence) and also the nodes as they are configured and come online.

The demo scenario is the almost the most basic possible with a simple ping scenario being enacted between the nodes but in order to do this the nodes, drivers, stacks (IPv4/TCP/Socket API/Name Resolver/Routes) and simple traffic generator (ping) are configured and enabled within the OSGi environment.

This is a starting point for more elaborate scenarios - such as the 4WARD WP2 / 4 Integration demo

1.18.1 Initial Run of the Demo

Task Start Pidgin IM Client

GUI Application > Internet > Pidgin

Task New Terminal Shell - Load the Demo Applications

```
$ . $HOME/development/env.sh
```

```
$ cd $SPRING_DM
```

Task Start the BootStrap configuration Manager

```
$ mv org.tssg.config.manager.plan pickup
```

Note Watching the other terminal - wait until the application in successfully loaded.

Task Start the Logger Agent

```
$ mv org.tssg.logger.plan pickup
```

Note Watching the other terminal - wait until the application in successfully loaded.

Task Start Jnode0

```
$ mv org.tssg.node.plan pickup
```

Note Watching the other terminal - wait until the application in successfully loaded.

Note Wait for IM Buddy - networkb-user1 to come online (Jnode0 fully configured).

Task Start Jnode1

```
$ mv org.tssg.node1.plan pickup
```

Note Watching the other terminal - wait until the application in successfully loaded.

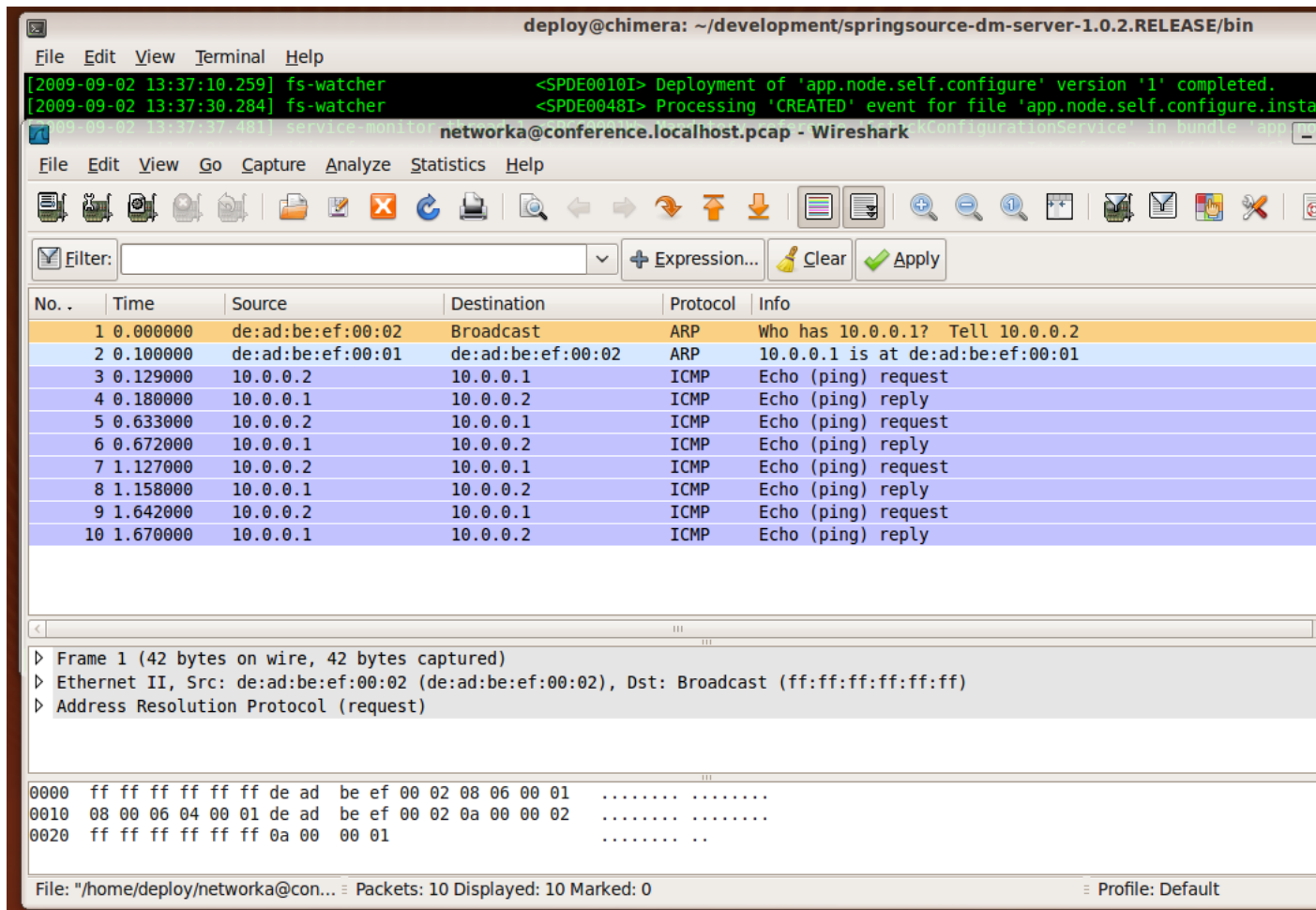
Note Wait for IM Buddy - networkb-user2 to come online (Jnode1 fully configured).

Note The logger application will produce PCAP files in the "/tmp" directory.

Task Start WireShark

GUI Applications > Internet > WireShark

Note The file to load into WireShark is `"/tmp/networka@conference.localhost.pcap"` as this will have the ARP/Ping traffic present in it.



1.18.2 Stopping the Spring DM Server

Task Stop the Spring DM Server

Note Simply CTRL-C in the shell you started the server.

1.18.3 Post-Initial Demo

Task Running it Again

Note Simply start the server. Do not copy demo plan files into the pickup directory. The server will automatically pick them up on all the subsequent server startups.

Note Delete the pcap files under the `"/tmp"` directory.

1.19 Development

1.19.1 Configuration of STS

Note: All packages/software installed under the linux user `"demo"`. You may need to change the path entered depending on your installation.

Task Add a Server to the GUI

STS Select Window > Show View > Servers

STS Right Click within the Servers Panel

STS Select New > Server

STS Servers host name : `"localhost"`

STS Server Type: SpringSource > SpringSource dm Server v2.0

STS Click Next

STS Name : SpringSource dm Server (Runtime) v2.0

STS SpringSource dm Server installation directory : `/home/demo/development/springsource/springsource-dm-server-2.0.0.RELEASE`

STS JRE: `jdk1.6.0_17`

STS Click Finish

Task Start the Spring Server

STS Select the Servers Panel

STS Right-Click `"SpringSource dm Server v2.0 at localhost"`, Select Start

STS In the console, you should see that the demo is started.

STS In the console, once the demo has completed.

STS Select the Servers Panel

STS Right-Click `"SpringSource dm Server v2.0 at localhost"`, Select Stop

Note This verifies your configuration.

1.20 Useful Links

Note Free Book - OSGi in Practise

URL <http://neilbartlett.name/blog/osgibook/>

–

Note SpringSource Enterprise Bundle Repository

URL <http://www.springsource.com/repository/app/>

–

Note SpringSource dm Server Programmer Guide

URL <http://static.springsource.com/projects/dm-server/1.0.x/programmer-guide/htmlsingle/programmer-guide.html>

–

Note OSGi Service Platform R4 Specification

URL <http://www.osgi.org/Download/Release4V41?info=nothanks>

–

Note Tutorial for Spring Dynamic Modules (DM) for OSGi Service Platforms

URL <http://springosgi.googlepages.com/>

1.21 Tips Jar

1.21.1 Logging

Tracing and logging is configured at a server level. See the file : serviceability.xml in the \$HOME/development/dm-server-2.0.0.RELEASE/config.

Apache Commons

```
import org.apache.commons.logging.Log;
import org.apache.commons.logging.LogFactory;
:
private static final Log logger = LogFactory.getLog(ClassName.class);
:
logger.debug(message);
```

1.21.2 Eclipse wc-props warnings

Preferences -> Java -> Compiler -> Building

Add .svn to the "Filtered resources" list -> "*.launch, .svn"