Contents

Articles

References

Developer Guide

The developer environment can be set in any Operating System (Linux, Windows, etc.) since it is a multi-platform system, but Linux is recommended, because is what it is being used.

You need to install Maven, eclipse IDE and some plugins. For OpenKM 4.0 and 5.0 you have these requirements:

- Maven at http://maven.apache.org/
- Eclipse Juno (Eclipse IDE for Java EE Developers) at http://eclipse.org/juno/
- Subversion eclipse plugin at http://subclipse.tigris.org
- Maven to eclipse plugin at http://m2eclipse.sonatype.org
- Java JDK 1.6 at http://www.oracle.com/technetwork/java/javase/downloads/index.html

> Starting from Eclipse Galileo you have a Marketplace accessible from Help > Eclipse Marketplace... Here you can search for "maven" to install the "Maven Integration for Eclipse" and "Maven Integration for Eclipse (Extras)" solutions. Also you can search for "subversion" to find the "Subclipse" plugin.

> The easier way to get an Eclipse installation with the required plugins is using JBoss Developer Studio.

Once all them are installed, you can download OpenKM and build it.

- Configure Java in Eclipse
- Configure tomcat server in Eclipse
- Maven installation
- Checkout source from Subversion
- Development tips
- Debug with tomcat
- Debugging with GWT
- Profiling OpenKM 🔒
- Doxygen OpenKM 5.0.x [1] (OpenKM packages & classes & files documentation)
- Doxygen OpenKM 5.1.x [2] (OpenKM packages & classes & files documentation)

Note: You can configure Eclipse to integrate with MantisBT. Read Mylyn-Mantis Repository Connecto [3]for more info.

Eclipse Development Quick Install Guide

This quick install is valid for Eclipse Indigo.

1. Download **Eclipse IDE for Java EE Developers** from http://www.eclipse.org/downloads/.
2. Go to Help > Eclipse Marketplace and install these plugins:
    1. Maven Integration for Eclipse [4]
    2. Subclipse [5]

Alternative: Add repository

- Subclipse 1.6.x Update Site -> http://subclipse.tigris.org/update_1.6.x
- Maven Integration for Eclipse - http://m2eclipse.sonatype.org/sites/m2e/
- Maven Integration for Eclipse Extras - http://m2eclipse.sonatype.org/sites/m2e-extras/

Deprecated

OpenKM version 5.x and older

- Configure JBoss server in Eclipse
- Debugging remote server
- Browsing embeded database
- Debugging OpenKM with Jboss

Changelog

Changes between developer guide 5.0 to 6.0

- GWT 2.4.0
- OpenKM now is executed under tomcat
- Changed to Eclipse Juno

Changes between developer guide 4.0 to 5.0

- Development tips changes (how to enable OpenKM extensions in compilation and disable automatic GWT compilation)
- GWT 2.0.4
- Java package has been refactoring to "com.openkm" all references to older package in configuration files has been changed

Changes between developer guide 2.0 to 4.0

- Changed to Eclipse Galileo
- Now we use maven in replacement of JBoss Tools

Changes between developer guide 1.2 to 2.0

- Changed IDE to Eclipse Europa / JBoss tools, before was JBoss IDE.
- GWT 1.5.3 is needed to compile, before was version 1.4.6
- Changed packaging to new jboss tools packaging.
- Changed GWT generating to temporary JBoss folder to project folders to developing.
- Added new generateback.sh and generateback.bat to GWT admin.
- Upgraded JBoss server to version 4.2.2.GA

Changes between developer guide 1.0 and 1.1 to 1.2

- GWT 1.4.6 is needed to compile, before was version 1.3.3.
- Changes on GWT shell configuration, now not uses GWTDS variable (code deleted).
- Configured generate.sh to deploy to tmp JBoss with OpenKM.ear deployed to fast developing.
- Added gecko to Main.gwt.xml to fast compiling only gecko when is uncommented.

References

[1] http://doxygen.openkm.com/5.0.x

[2] http://doxygen.openkm.com/5.1.x

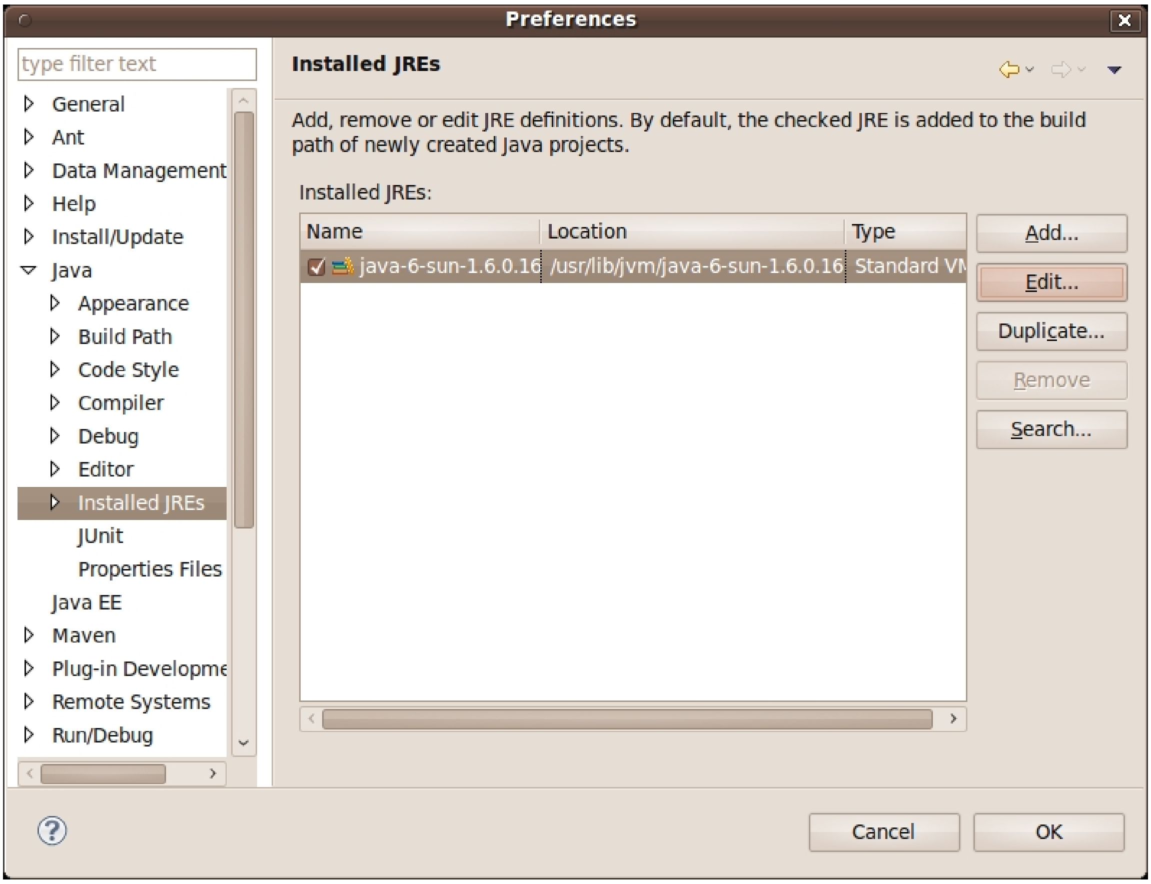
[3] http://sourceforge.net/apps/mediawiki/mylyn-mantis/index.php?title=Main_Page

[4] http://marketplace.eclipse.org/content/maven-integration-eclipse

[5] http://marketplace.eclipse.org/content/subclipse

Configure Java in Eclipse

Go to **Window / Preferences / Java / Installed JRE**. You must have JDK 1.6 defined here:

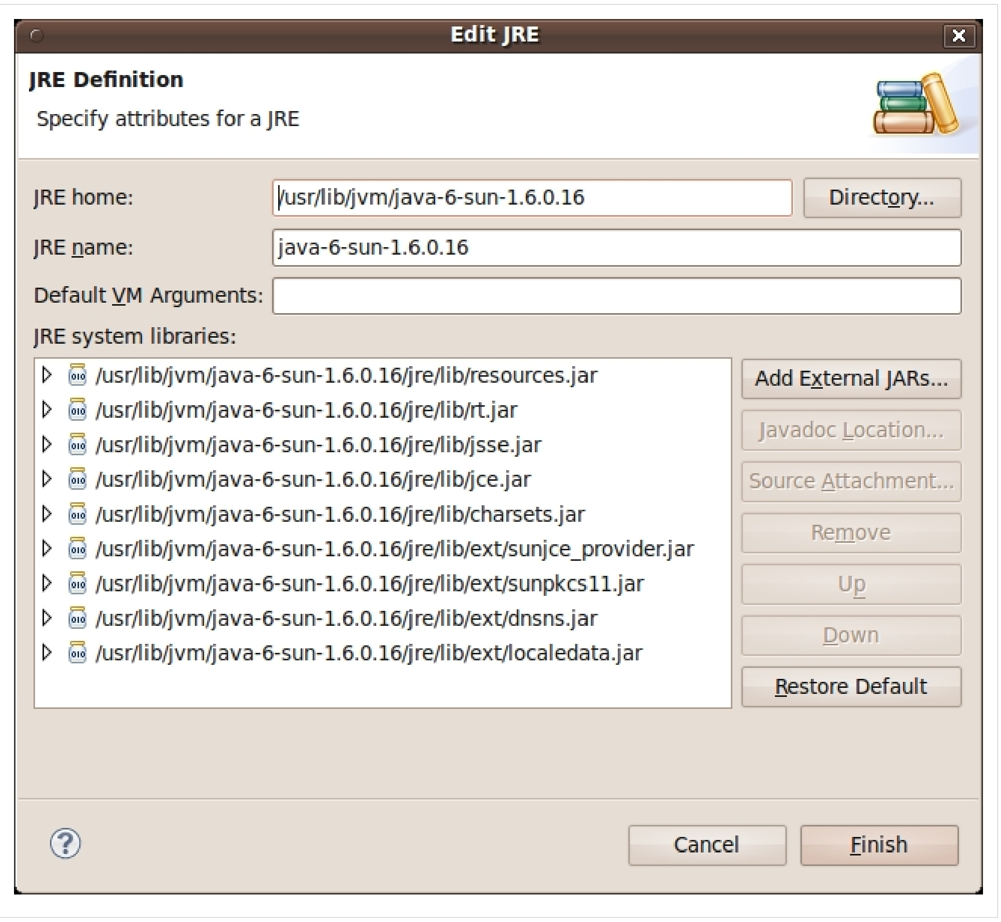


Installed JREs

Add, remove or edit JRE definitions. By default, the checked JRE is added to the build path of newly created Java projects.

Installed JREs:

Name	Location	Type
☑ java-6-sun-1.6.0.16	/usr/lib/jvm/java-6-sun-1.6.0.16	Standard V



JRE home: /usr/lib/jvm/java-6-sun-1.6.0.16

JRE name: java-6-sun-1.6.0.16

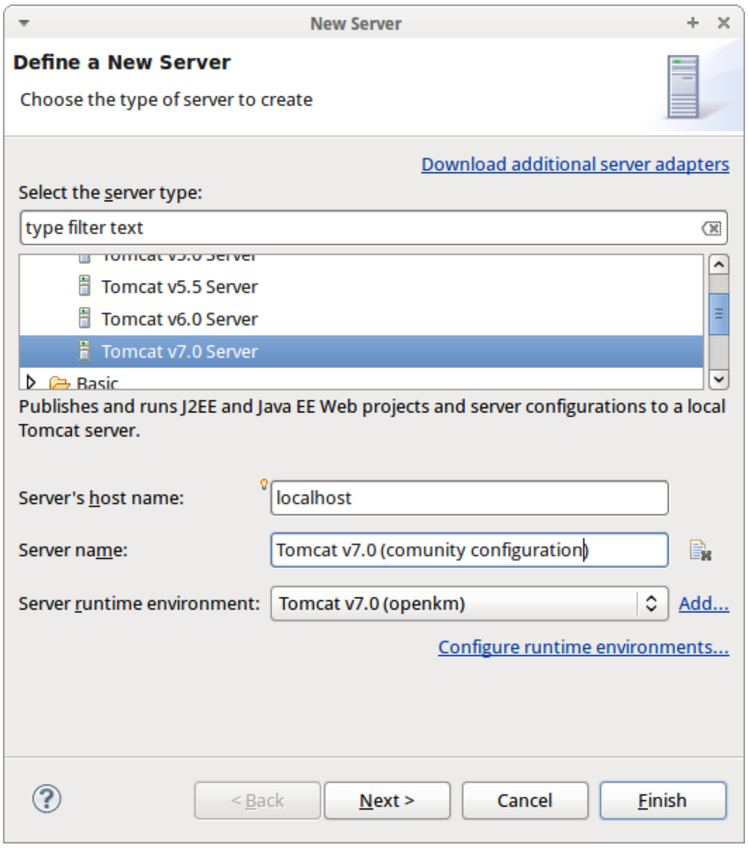
Default VM Arguments:

JRE system libraries:

- /usr/lib/jvm/java-6-sun-1.6.0.16/jre/lib/resources.jar
- /usr/lib/jvm/java-6-sun-1.6.0.16/jre/lib/rt.jar
- /usr/lib/jvm/java-6-sun-1.6.0.16/jre/lib/jsse.jar
- /usr/lib/jvm/java-6-sun-1.6.0.16/jre/lib/jce.jar
- /usr/lib/jvm/java-6-sun-1.6.0.16/jre/lib/charsets.jar
- /usr/lib/jvm/java-6-sun-1.6.0.16/jre/lib/ext/sunjce_provider.jar
- /usr/lib/jvm/java-6-sun-1.6.0.16/jre/lib/ext/sunpkcs11.jar
- /usr/lib/jvm/java-6-sun-1.6.0.16/jre/lib/ext/dnsns.jar
- /usr/lib/jvm/java-6-sun-1.6.0.16/jre/lib/ext/localedata.jar

Configure tomcat server in Eclipse

Go to servers tab Using contextual menus select Add / server. **Select Tomcat 7 server**:



New Server

Define a New Server

Choose the type of server to create

Download additional server adapters

Select the server type: type filter text

- Tomcat v5.5 Server
- Tomcat v6.0 Server
- Tomcat v7.0 Server
- Basic

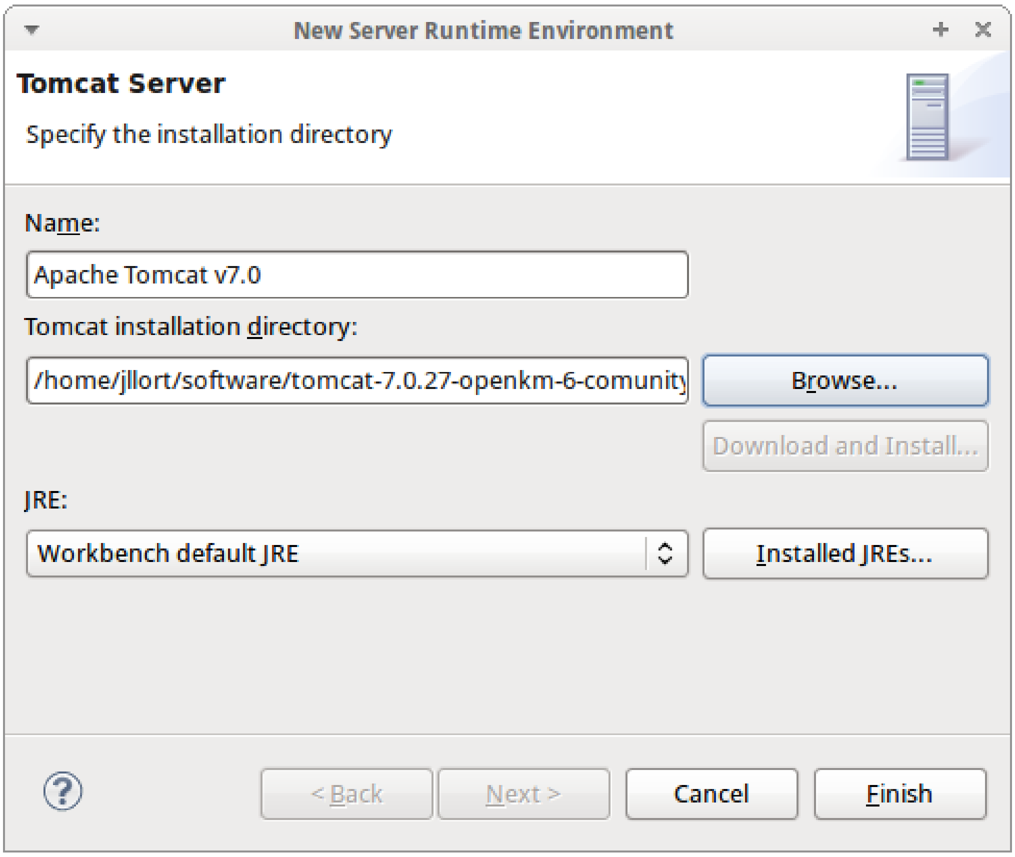
Publishes and runs J2EE and Java EE Web projects and server configurations to a local Tomcat server.

Server's host name: localhost

Server name: Tomcat v7.0 (comunity configuration)

Server runtime environment: Tomcat v7.0 (openkm)

Configure a new **Run time server environment**:



New Server Runtime Environment

Tomcat Server

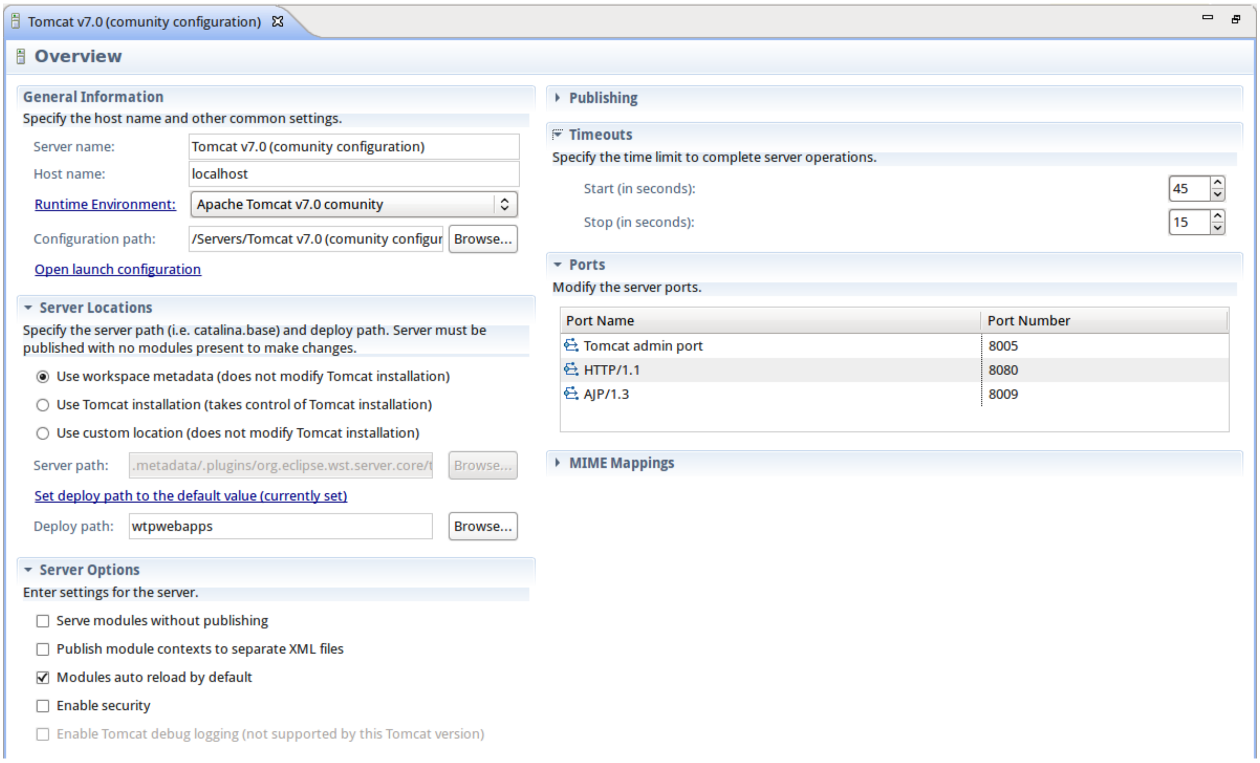
Specify the installation directory

Name: Apache Tomcat v7.0

Tomcat installation directory: /home/jllort/software/tomcat-7.0.27-openkm-6-comunity

JRE: Workbench default JRE

- Checkbox **Use Tomcat installation** (takes control of tomcat installation).
- Increment startup and stop **timeouts**.



Tomcat v7.0 (comunity configuration)
Overview

General Information

Specify the host name and other common settings.

Server name: Tomcat v7.0 (comunity configuration)

Host name: localhost

Runtime Environment: Apache Tomcat v7.0 comunity

Configuration path: /Servers/Tomcat v7.0 (comunity configur

Browse...
Open launch configuration

Server Locations

Specify the server path (i.e. catalina.base) and deploy path. Server must be published with no modules present to make changes.

● Use workspace metadata (does not modify Tomcat installation)

○ Use Tomcat installation (takes control of Tomcat installation)

○ Use custom location (does not modify Tomcat installation)

Server path: .metadata/.plugins/org.eclipse.wst.server.core/t

Deploy path: wtpwebapps

Server Options

Enter settings for the server.

☐ Serve modules without publishing

☐ Publish module contexts to separate XML files

☑ Modules auto reload by default

☐ Enable security

☐ Enable Tomcat debug logging (not supported by this Tomcat version)

Timeouts

Specify the time limit to complete server operations.

Start (in seconds): 45

Stop (in seconds): 15

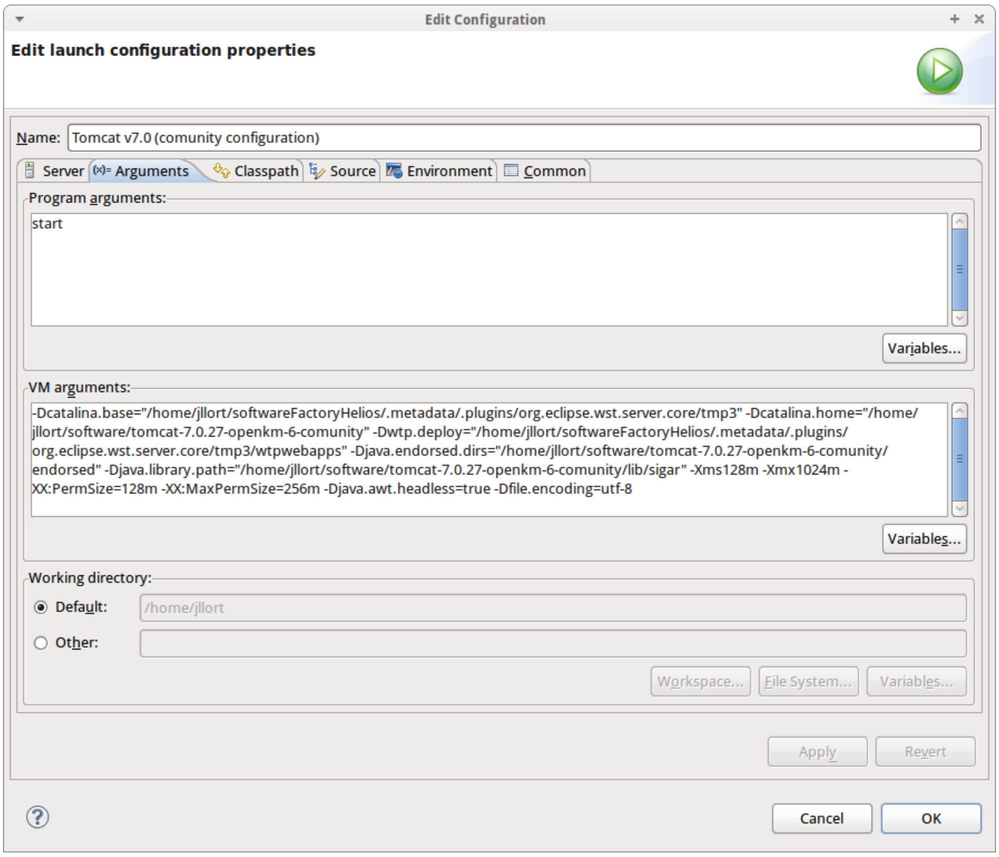
Ports

Modify the server ports.

Port Name	Port Number
Tomcat admin port	8005
HTTP/1.1	8080
AJP/1.3	8009

Edit **launch configuration**: Add **sigar** library path and **utf-8** file enconding at arguments tab

```
-Djava.library.path="/your tomcat path/lib/sigar" -Xms128m -Xmx1024m
-XX:PermSize=128m -XX:MaxPermSize=256m  -Djava.awt.headless=true
-Dfile.encoding=utf-8
```



Name: Tomcat v7.0 (comunity configuration)

Program arguments:

```
start
```

VM arguments:

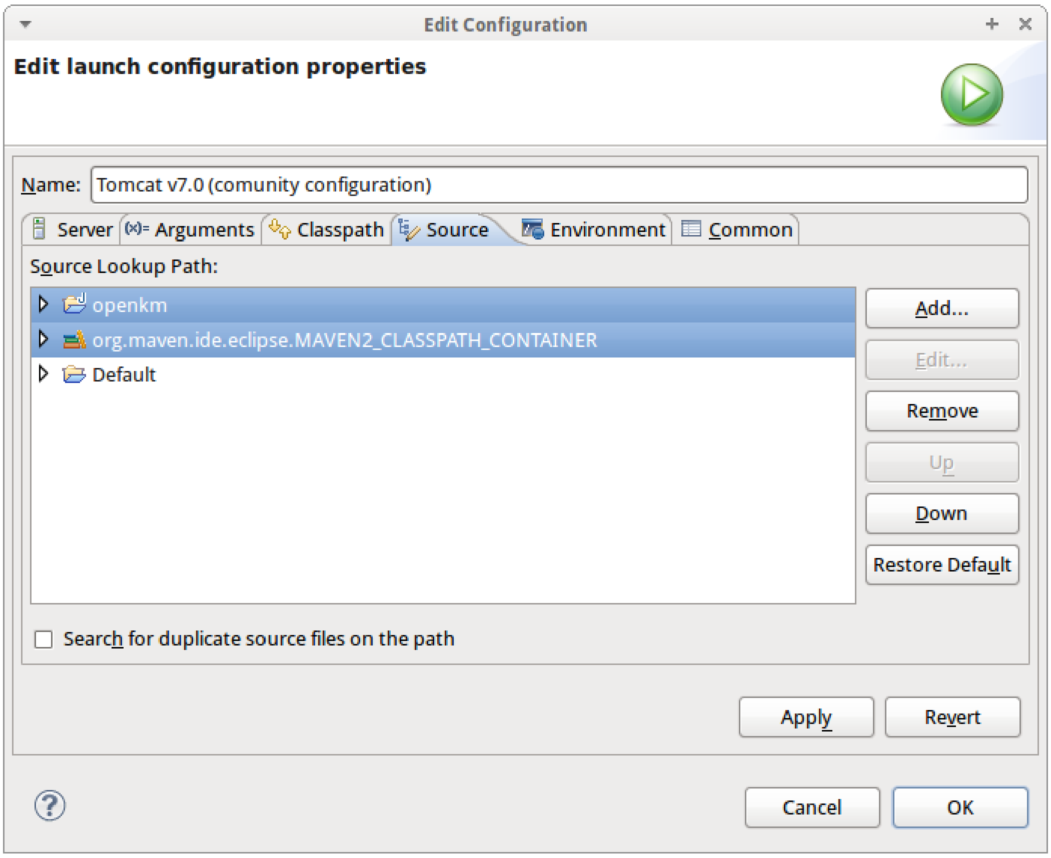
```
-Dcatalina.base="/home/jllort/softwareFactoryHelios/.metadata/.plugins/org.eclipse.wst.server.core/tmp3" -Dcatalina.home="/home/
jllort/software/tomcat-7.0.27-openkm-6-comunity" -Dwtp.deploy="/home/jllort/softwareFactoryHelios/.metadata/.plugins/
org.eclipse.wst.server.core/tmp3/wtpwebapps" -Djava.endorsed.dirs="/home/jllort/software/tomcat-7.0.27-openkm-6-comunity/
endorsed" -Djava.library.path="/home/jllort/software/tomcat-7.0.27-openkm-6-comunity/lib/sigar" -Xms128m -Xmx1024m -
XX:PermSize=128m -XX:MaxPermSize=256m -Djava.awt.headless=true -Dfile.encoding=utf-8
```

Working directory:

● Default: /home/jllort

○ Other:

Add your **openkm project** at source tab



Name: Tomcat v7.0 (comunity configuration)

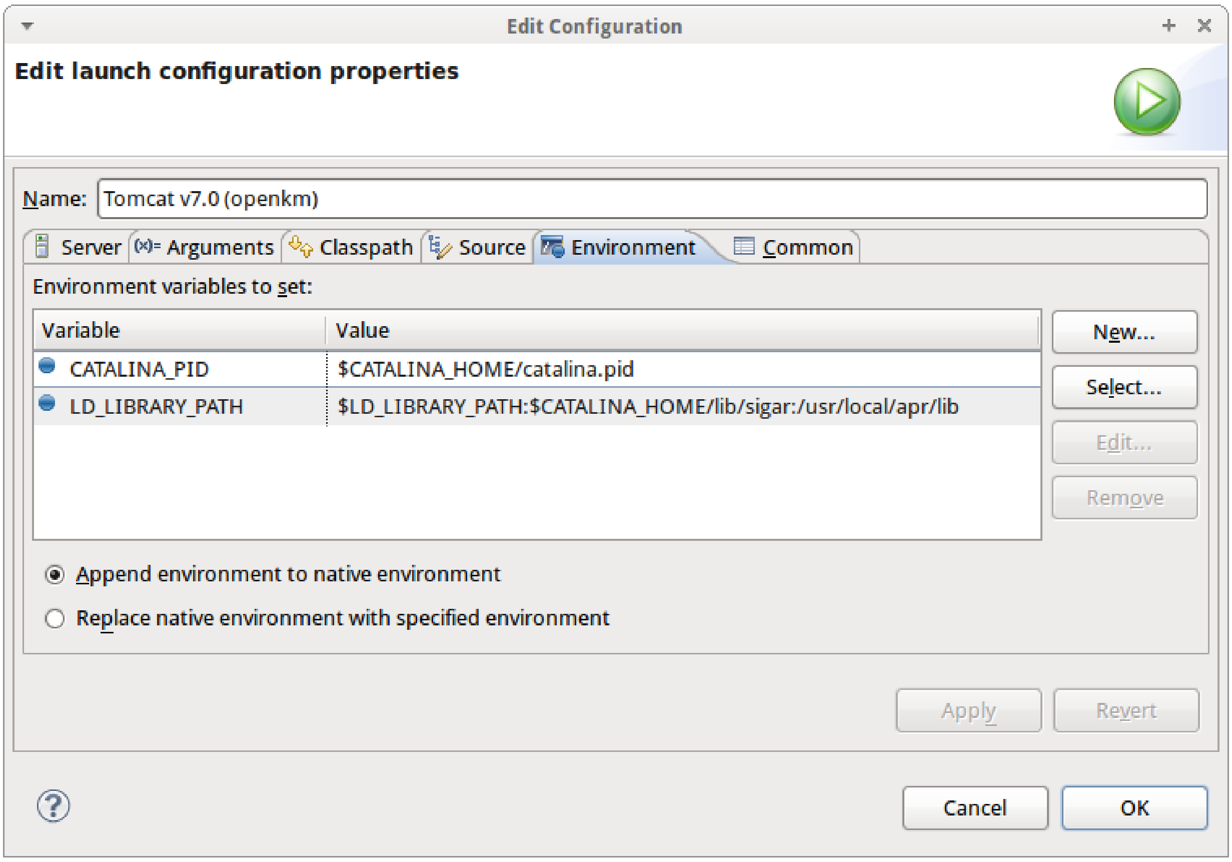
Source Lookup Path:

- openkm
- org.maven.ide.eclipse.MAVEN2_CLASSPATH_CONTAINER
- Default

☐ Search for duplicate source files on the path

Add **CATALINA_PID** and **LD_LIBRARY_PATH** environement variables:

```
CATALINA_PID=$CATALINA_HOME/catalina.pid
LD_LIBRARY_PATH=$LD_LIBRARY_PATH:$CATALINA_HOME/lib/sigar:/usr/local/apr/lib
```



Edit launch configuration properties

Name: Tomcat v7.0 (openkm)

Environment variables to set:

Variable	Value
CATALINA_PID	$CATALINA_HOME/catalina.pid
LD_LIBRARY_PATH	$LD_LIBRARY_PATH:$CATALINA_HOME/lib/sigar:/usr/local/apr/lib

● Append environment to native environment

○ Replace native environment with specified environment

Maven installation

Maven is a software tool for Java project management and build automation. It is similar in functionality to the Apache Ant tool, but is based on different concepts. Maven is hosted by the Apache Software Foundation, where it was formerly part of the Jakarta Project.

Maven uses a construct known as a Project Object Model (POM) to describe the software project being built, its dependencies on other external modules and components, and the build order. It comes with pre-defined targets for performing certain well defined tasks such as compilation of code and its packaging.

In GNU/Linux you can add the Maven executable to the binary path editing the *$HOME/.bashrc* file and adding this line at the end:

```
export PATH=$PATH:/path/to/maven/installation/bin
export MAVEN_OPTS="-Xmx512m"
```

Learn more about Maven at http://maven.apache.org.

Register missing Maven dependencies

Although project dependencies usually at located at Maven repositories, sometimes there are missing artifacts. So, we have to provide these libraries registering them in the local Maven repository. Follow these steps to register some libraries needed by OpenKM:

- Open an operating system console.
- Go to OpenKM project path in your local file system.
- Drive into *ext-libs* and execute install.sh (Windows users simply open this file with text editor and execute the command)

> Starting with OpenKM 5.0, you don't need to register manually these dependencies.

Some Maven tips

Download sources and javadoc from libraries dependency to improve the developer experience:

```
$ mvn install -DdownloadSources=true  -DdownloadJavadocs=true
```

You can make Maven ignore test failures in the build process this way:

```
$ mvn install -Dmaven.test.failure.ignore=true
```

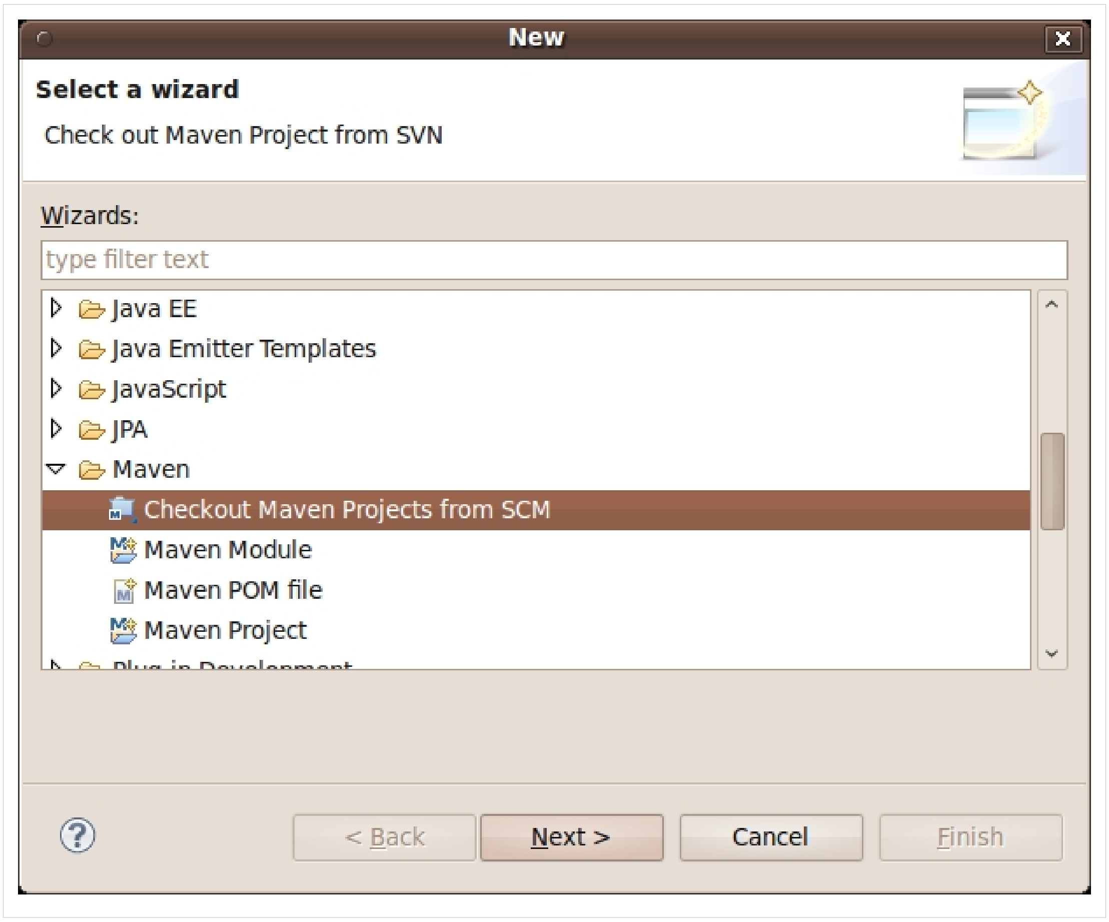
If you would like to skip tests, you can do so by executing the following command:

```
$ mvn install -Dmaven.test.skip=true
```

Checkout source from Subversion

Go to **File / New / Other** and select **Maven / Checkout Maven Projects from SCM**:

Hint: If your SCM dropdown list is empty, install the Maven Subclipse extension from (http://m2eclipse.sonatype.org/sites/m2e-extras/).



> SourceForge has updated some of its features and the OpenKM Subversion repository has changed. To get the source code: svn checkout [1] openkm-code

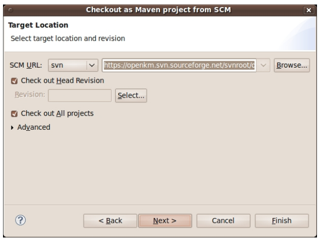
Select the **svn** type and type the url http://svn.code.sf.net/p/openkm/code/branches/6.2/openkm to refer version 6.2:

Select the **svn** type and type the url https://openkm.svn.sourceforge.net/svnroot/openkm/branches/5.1/openkm to refer version 5.1:

Select the **svn** type and type the url https://openkm.svn.sourceforge.net/svnroot/openkm/branches/5.0/openkm to refer version 5.0:

Select the **svn** type and type the url https://openkm.svn.sourceforge.net/svnroot/openkm/branches/4.1/openkm to refer version 4.1:

Select the **svn** type and type the url https://openkm.svn.sourceforge.net/svnroot/openkm/branches/4.0/openkm to refer version 4.0:

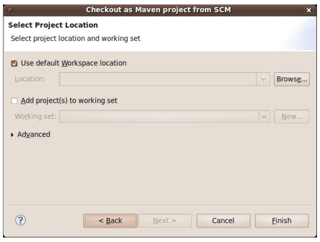

Checkout as Maven project from SCM
Target Location
Select target location and revision
Browse...
Select...
Advanced
< Back
Next >
Cancel
Finish

SCM URL: svn https://openkm.svn.sourceforge.net/svnroot/c

☑ Check out Head Revision

Revision:

☑ Check out All projects


Checkout as Maven project from SCM
Select Project Location
Select project location and working set
Browse...
New...
Advanced
< Back
Next >
Cancel
Finish

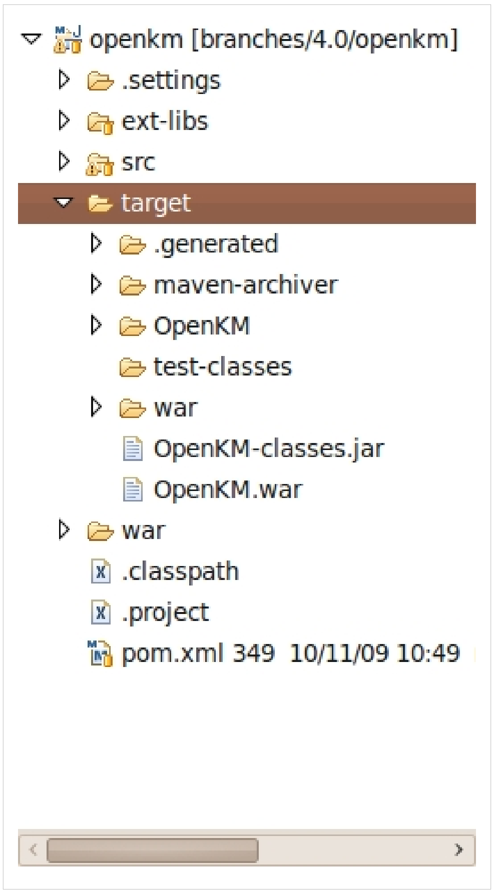
☑ Use default Workspace location

Location:

☐ Add project(s) to working set

Working set:

Be patient, first time you downloading OpenKM from svn, you're downloading all libraries to your maven repository. It could take several minutes depending your Internet connection.



After OpenKM will be downloaded, and automatically compiled.

If you want native support for Subversion, you have to install the *libsvn-java* package in Linux. The edit the *eclipse.ini* configuration file and add the following line:

```
-Djava.library.path=/usr/lib/jni
```

References

[1] http://svn.code.sf.net/p/openkm/code/branches/6.2/openkm

JBoss Developer Studio

You can get a free copy of JBoss Developer Studio from http://www.jboss.com/products/devstudio/. Once registered you can download the program compiled for your preferred Operating System.

In order to install Maven and Subversion integration, go to **Help** > **Install New Software...** and select:

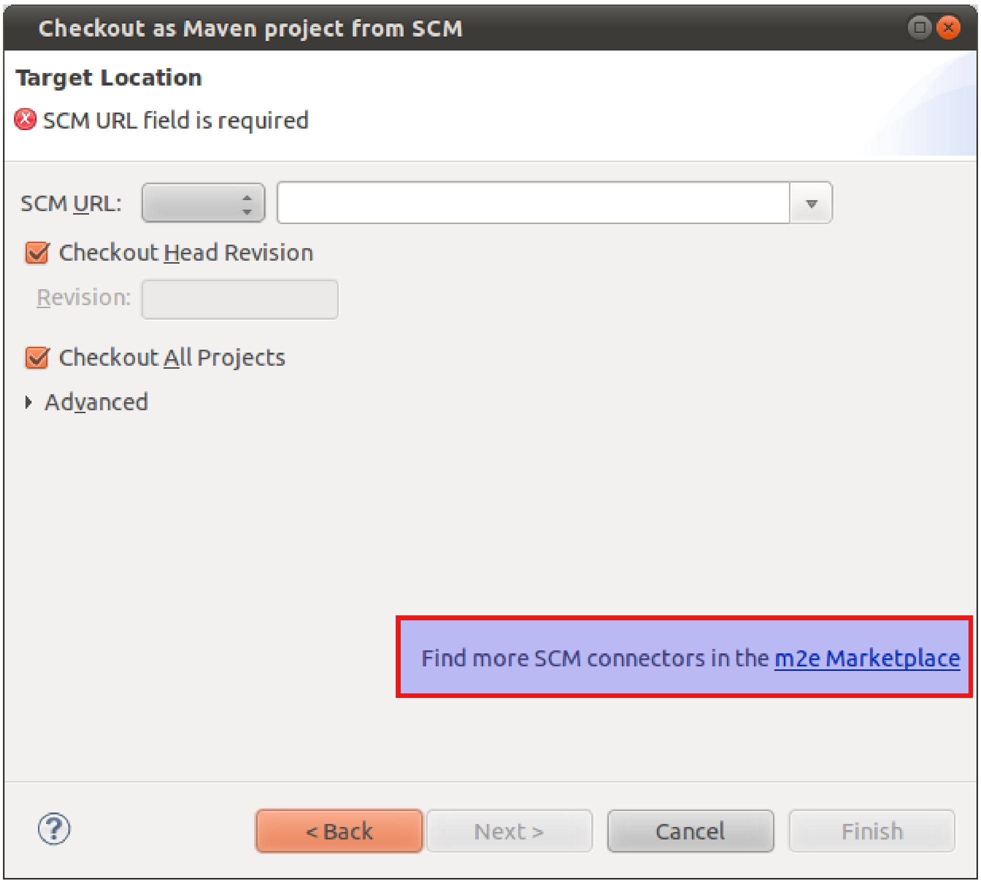
```
"JBoss Developer Studio 4.0 Extras - https://devstudio.jboss.com/updates/4.0/extras/"
```

Open "JBoss Selected Features" and select

- Maven Integration
- Subclipse

Once selected, click on the **Next** button and follow the installation wizard until finished. In the last step JBoss Studio need to be restarted.

When you import a Subversion project, you will see this dialog:



Checkout as Maven project from SCM

Target Location

SCM URL field is required

SCM URL:

☑ Checkout Head Revision

Revision:

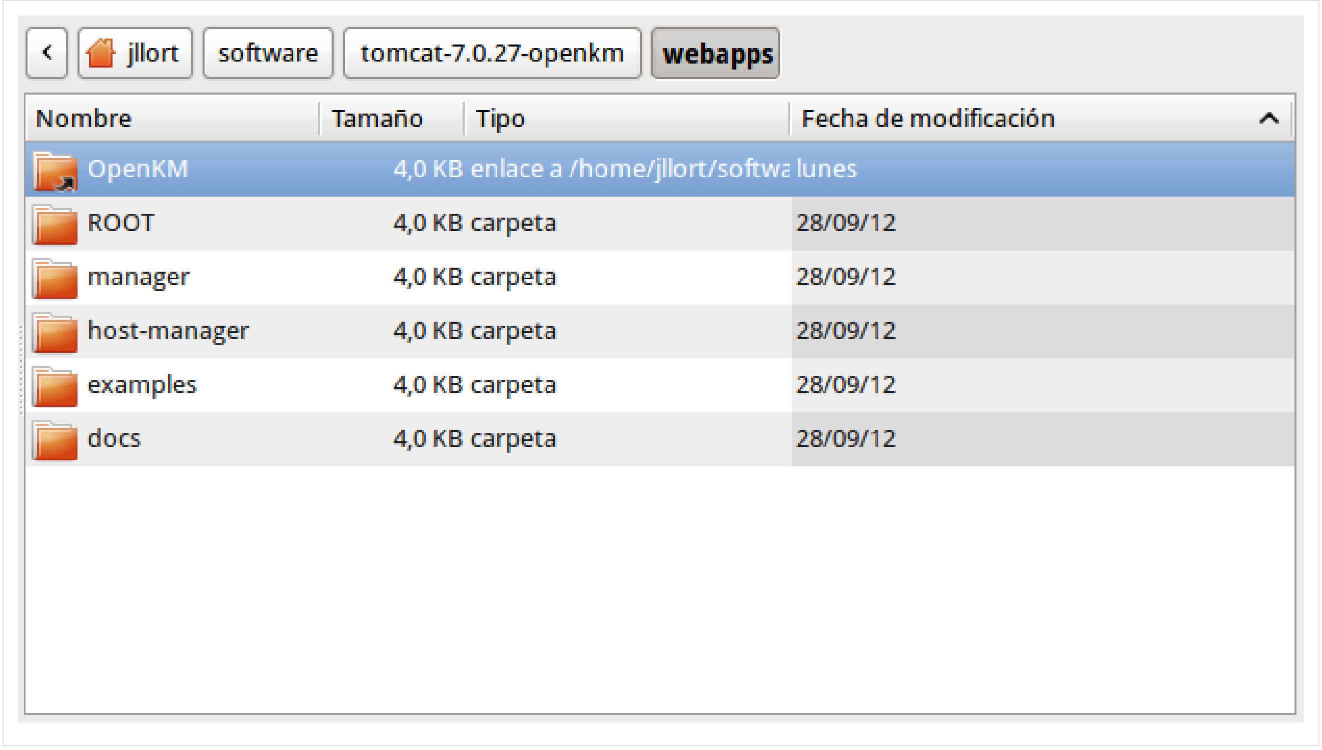
☑ Checkout All Projects

Find more SCM connectors in the m2e Marketplace

The first time you will need to install the Subversion connector from the **m2e Marketplace**. Go to the **m2e Team providers** group and select **m2e-subclipse**. After that, click on the **Finish** button.

Development tips OpenKM 6.0

We recommend downloading Tomcat+OpenKM from SourceForge, remove *OpenKM.war* and develop with this JBoss server configuration. For better development we recommend not deploying *OpenKM.war* file it's better setting in *$TOMCAT_HOME/webapps* some alias to *target/OpenKM* folder:



Nombre	Tamaño	Tipo	Fecha de modificación
OpenKM	4,0 KB	enlace a /home/jllort/softwa	lunes
ROOT	4,0 KB	carpeta	28/09/12
manager	4,0 KB	carpeta	28/09/12
host-manager	4,0 KB	carpeta	28/09/12
examples	4,0 KB	carpeta	28/09/12
docs	4,0 KB	carpeta	28/09/12

> To create symbolic links in windows use juntion.exe [1]

In case you're making strong changes in OpenKM UI (GWT) we recommend disabling pom compile directive

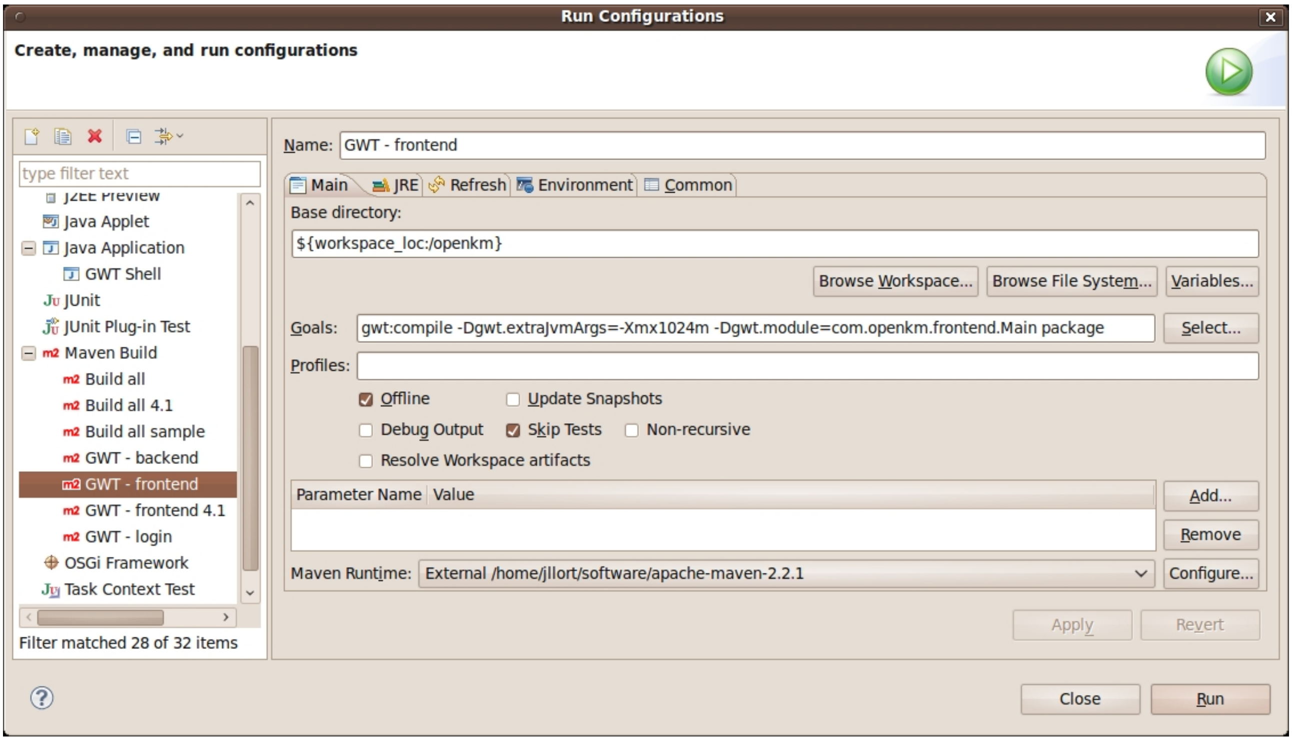
```
<plugins>
      <plugin>
        <groupId>org.codehaus.mojo</groupId>
        <artifactId>gwt-maven-plugin</artifactId>
        <version>${gwt.version}</version>
        <executions>
          <execution>
            <goals>
              <goal>compile</goal>
              <!-- <goal>generateAsync</goal> -->
              <!-- <goal>test</goal> -->
            </goals>
          </execution>
        </executions>
        <configuration>
          <runTarget>com.openkm.frontend.Main/index.html</runTarget>
          <modules>
            <module>com.openkm.frontend.Main</module>
          </modules>
          <localWorkers>4</localWorkers>
        </configuration>
      </plugin>
```

Important, depending the number of cores and memory decrease the number of gwt instances compiled at same time decreasing locarlWorkers, otherside your computer may hang

Is possible to compile GWT modules individually by this way:

```
$ mvn gwt:compile -Dgwt.module=com.openkm.frontend.Main
```

You might be interested in running frontend and backend GWT packaging from eclipse. Go to run configurations and add in maven build



Name: GWT - frontend

Base directory: ${workspace_loc:/openkm}

Goals: gwt:compile -Dgwt.extraJvmArgs=-Xmx1024m -Dgwt.module=com.openkm.frontend.Main package

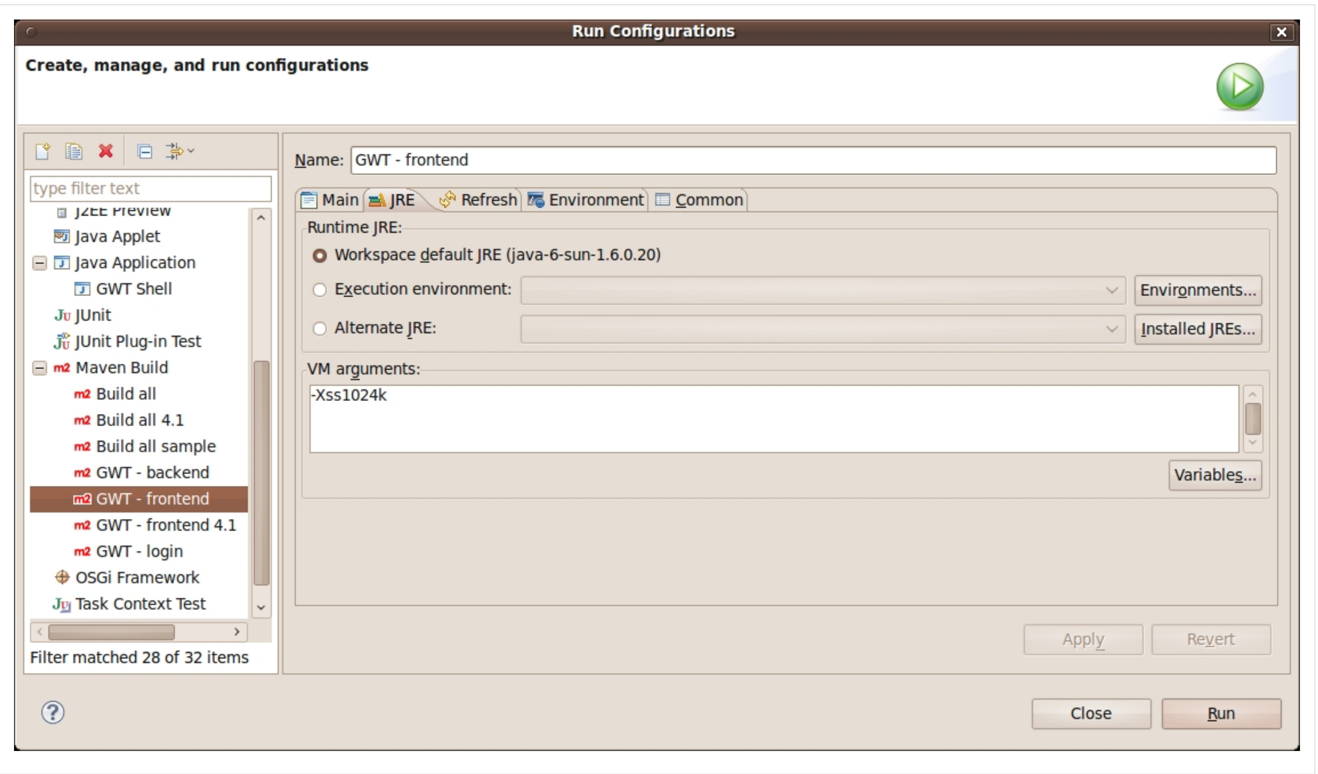
Profiles:

☑ Offline ☐ Update Snapshots

☐ Debug Output ☑ Skip Tests ☐ Non-recursive

☐ Resolve Workspace artifacts

Maven Runtime: External /home/jllort/software/apache-maven-2.2.1



Run Configurations

Create, manage, and run configurations

Name: GWT - frontend

Runtime JRE:
● Workspace default JRE (java-6-sun-1.6.0.20)
○ Execution environment:
○ Alternate JRE:

VM arguments:
-Xss1024k

Filter matched 28 of 32 items

To achieve a faster compile we use to development Firefox, uncommenting gecko line on file **Main.gwt.xml** GWT compilation only for gecko browser runs fine. Don't forget at finish to comment it and try application with IE or other browsers.

```
<!-- Compile for Firefox only -->
<set-property name="user.agent" value="gecko"/>
```

User contribution script modified to tomcat deploy

```
#!/bin/bash
# @author: Kenneth Walter

#Cleans and re-compiles the source files
mvn -Dmaven.test.skip=true clean gwt:compile install $*

#Set the TARGET_DIRECTORY to the path of your JBoss installation
TARGET_DIRECTORY=~/Downloads/tomcat-7.0.27/
FILE_TO_MOVE=target/OpenKM.war
#This will only attempt to replace the existing WAR if the new WAR
exists
if [ -f $FILE_TO_MOVE ]
then
   echo 'Deploying WAR to JBoss Directory'
   cp -v $FILE_TO_MOVE $TARGET_DIRECTORY/webapps/.
   echo 'Done'
fi
```

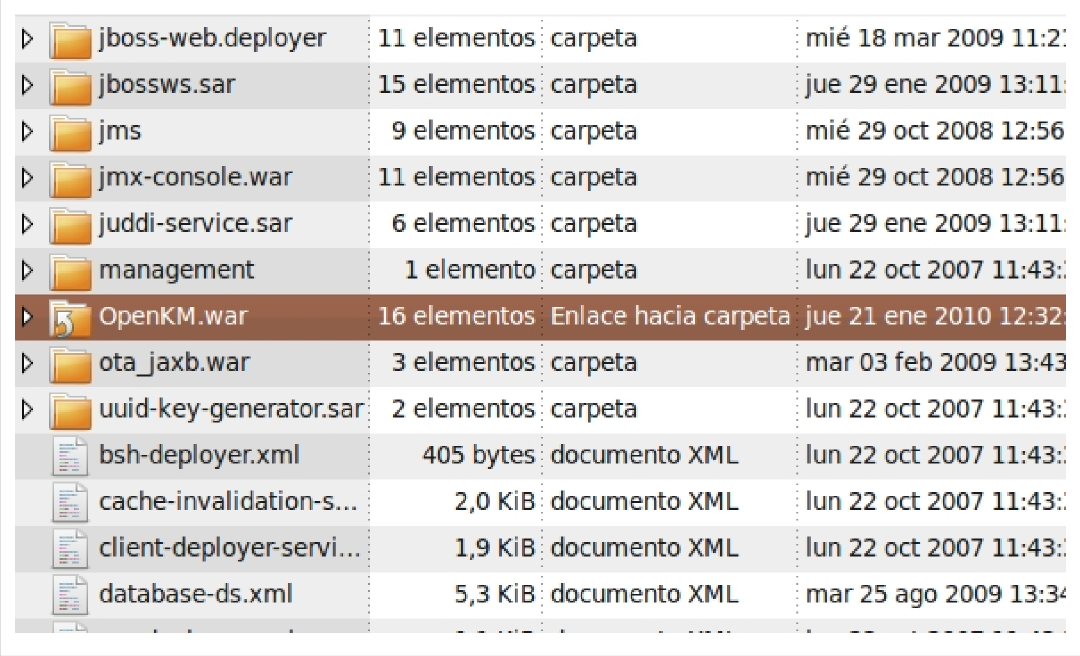
More info about this script at Cannot create or use example extensions [2] forum thread.

References

[1] http://technet.microsoft.com/en-us/sysinternals/bb896768
[2] http://forum.openkm.com/viewtopic.php?f=31&t=5332

Development tips OpenKM 5.1

We recommend downloading JBoss+OpenKM from SourceForge, remove *OpenKM.war* and develop with this JBoss server configuration. For better development we recommend not deploying *OpenKM.war* file it's better setting in *$JBOSS_HOME/server/default/deploy* some alias to *target/OpenKM* folder:



jboss-web.deployer	11 elementos	carpeta	mié 18 mar 2009 11:2
jbossws.sar	15 elementos	carpeta	jue 29 ene 2009 13:11
jms	9 elementos	carpeta	mié 29 oct 2008 12:56
jmx-console.war	11 elementos	carpeta	mié 29 oct 2008 12:56
juddi-service.sar	6 elementos	carpeta	jue 29 ene 2009 13:11
management	1 elemento	carpeta	lun 22 oct 2007 11:43:
OpenKM.war	16 elementos	Enlace hacia carpeta	jue 21 ene 2010 12:32
ota_jaxb.war	3 elementos	carpeta	mar 03 feb 2009 13:43
uuid-key-generator.sar	2 elementos	carpeta	lun 22 oct 2007 11:43:
bsh-deployer.xml	405 bytes	documento XML	lun 22 oct 2007 11:43:
cache-invalidation-s...	2,0 KiB	documento XML	lun 22 oct 2007 11:43:
client-deployer-servi...	1,9 KiB	documento XML	lun 22 oct 2007 11:43:
database-ds.xml	5,3 KiB	documento XML	mar 25 ago 2009 13:3

To create symbolic links in windows use juntion.exe [1]

In case you're making strong changes in OpenKM UI (GWT) we recommend disabling pom compile directive

```
<plugins>
      <plugin>
        <groupId>org.codehaus.mojo</groupId>
        <artifactId>gwt-maven-plugin</artifactId>
        <version>${gwt.version}</version>
        <executions>
          <execution>
            <goals>
              <goal>compile</goal>
              <!-- <goal>generateAsync</goal> -->
              <!-- <goal>test</goal> -->
            </goals>
          </execution>
        </executions>
        <configuration>
```

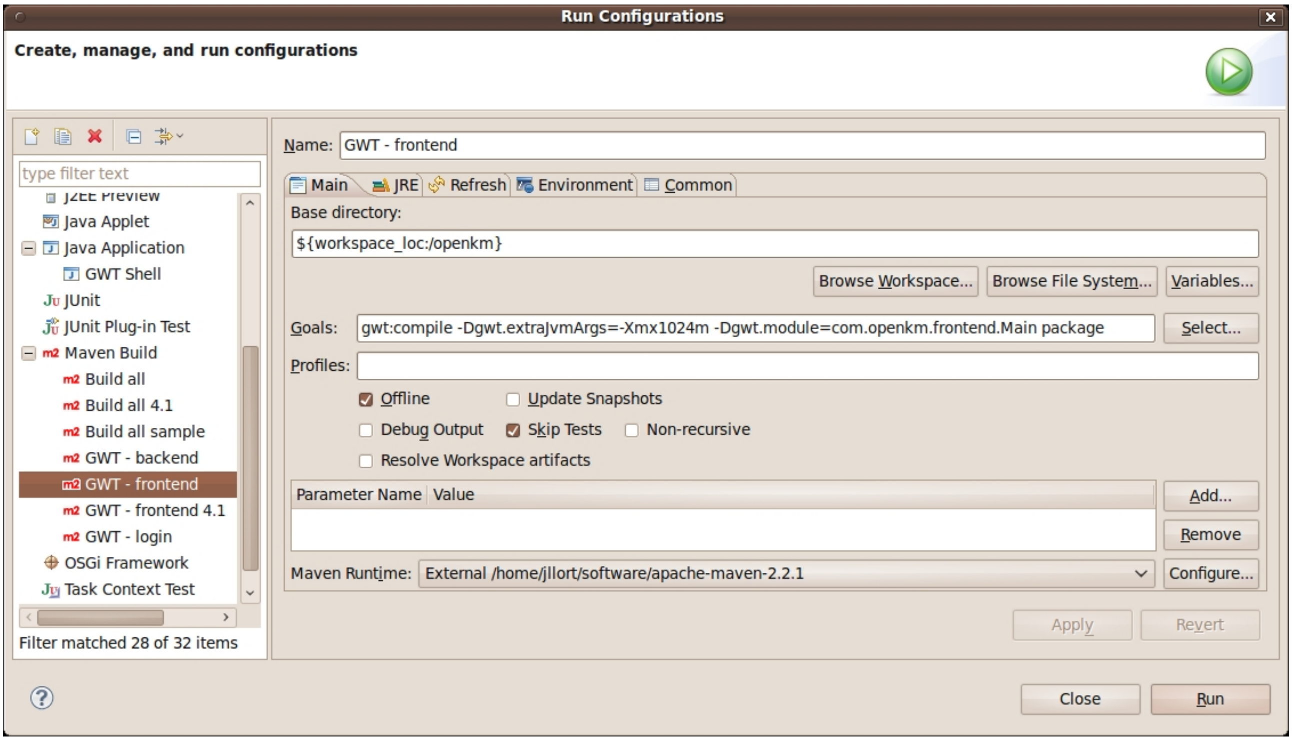
```
          <runTarget>com.openkm.frontend.Main/index.html</runTarget>
          <modules>
            <module>com.openkm.frontend.Main</module>
          </modules>
          <localWorkers>4</localWorkers>
        </configuration>
      </plugin>
```

Important, depending the number of cores and memory decrease the number of gwt instances compiled at same time decreasing locarlWorkers, otherside your computer may hang

Is possible to compile GWT modules individually by this way:

```
$ mvn gwt:compile -Dgwt.module=com.openkm.frontend.Main
```

You might be interested in running frontend and backend GWT packaging from eclipse. Go to run configurations and add in maven build



Name: GWT - frontend

Base directory: ${workspace_loc:/openkm}

Goals: gwt:compile -Dgwt.extraJvmArgs=-Xmx1024m -Dgwt.module=com.openkm.frontend.Main package

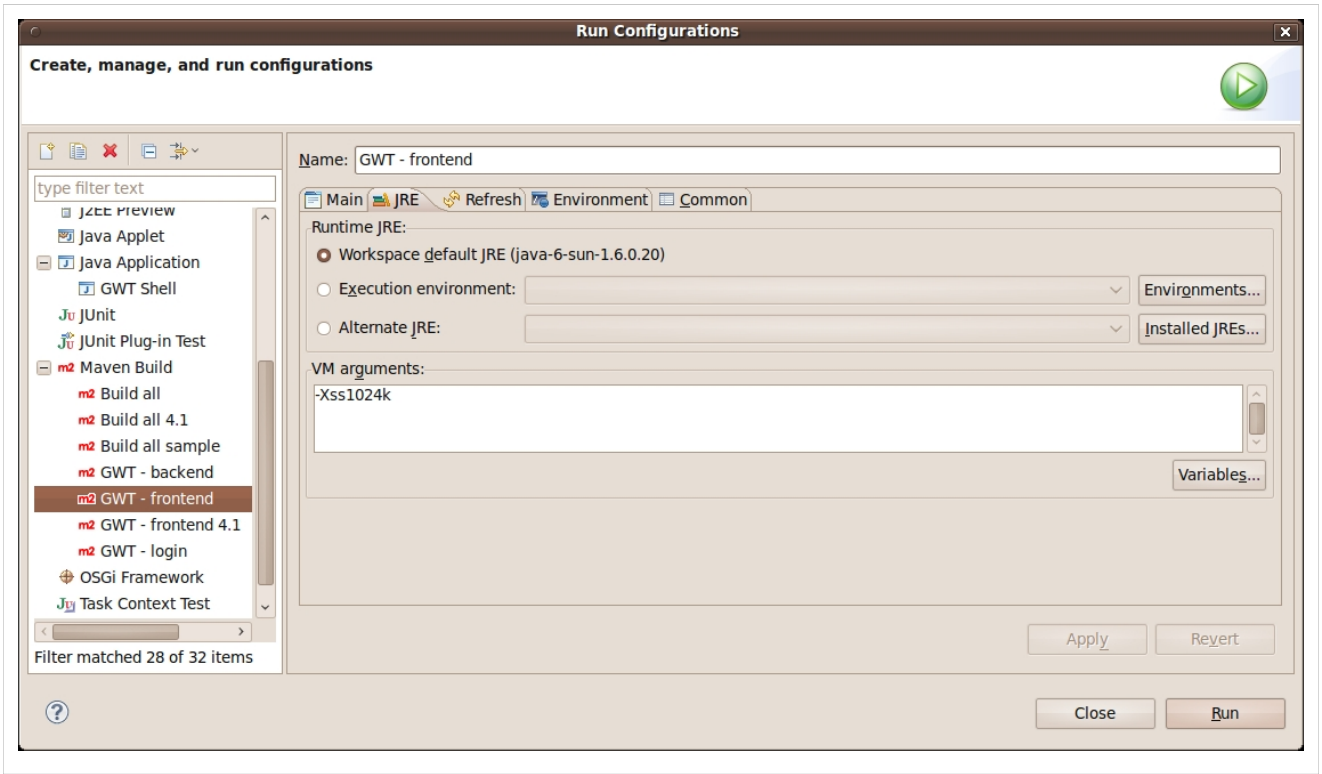
Profiles:

☑ Offline ☐ Update Snapshots

☐ Debug Output ☑ Skip Tests ☐ Non-recursive

☐ Resolve Workspace artifacts

Maven Runtime: External /home/jllort/software/apache-maven-2.2.1



Name: GWT - frontend

Runtime JRE:

● Workspace default JRE (java-6-sun-1.6.0.20)

VM arguments:

```
-Xss1024k
```

To achieve a faster compile we use to development Firefox, uncommenting gecko line on file **Main.gwt.xml** GWT compilation only for gecko browser runs fine. Don't forget at finish to comment it and try application with IE or other browsers.

```
<!-- Compile for Firefox only -->
<set-property name="user.agent" value="gecko"/>
```

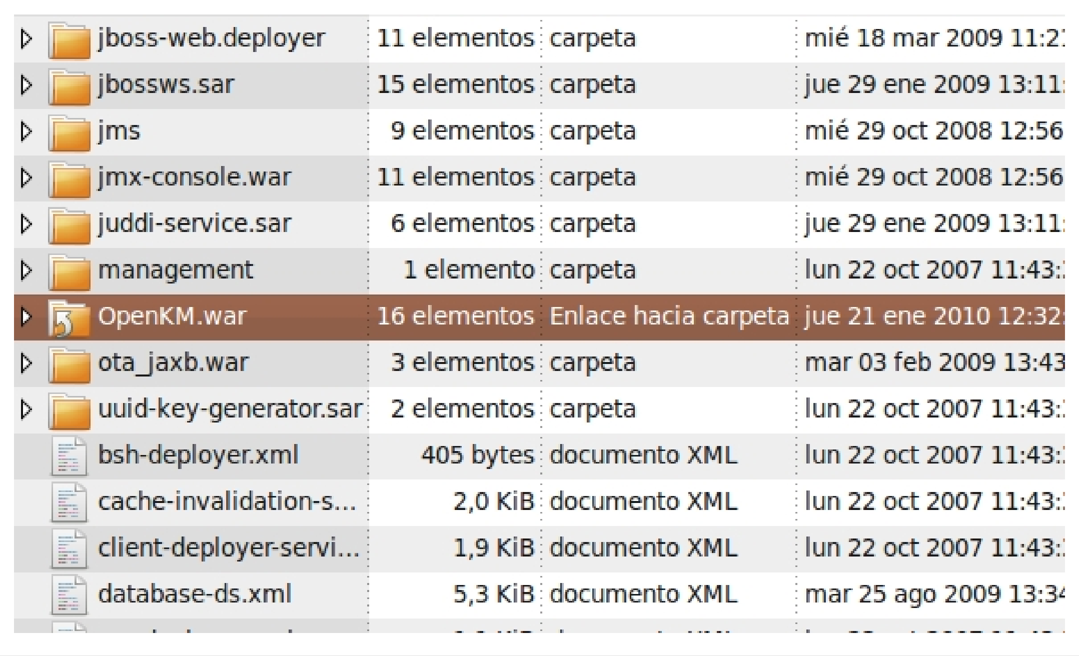
User contribution script

```
#!/bin/bash
# @author: Kenneth Walter

#Cleans and re-compiles the source files
mvn -Dmaven.test.skip=true clean gwt:compile install $*

#Set the TARGET_DIRECTORY to the path of your JBoss installation
TARGET_DIRECTORY=~/Downloads/jboss-4.2.3.GA/
FILE_TO_MOVE=target/OpenKM.war
#This will only attempt to replace the existing WAR if the new WAR
exists
if [ -f $FILE_TO_MOVE ]
then
   echo 'Deploying WAR to JBoss Directory'
   cp -v $FILE_TO_MOVE $TARGET_DIRECTORY/server/default/deploy/.
   echo 'Done'
fi
```

More info about this script at Cannot create or use example extensions [2] forum thread.

Development tips OpenKM 5.0

We recommend downloading JBoss+OpenKM from SourceForge, remove *OpenKM.war* and develop with this JBoss server configuration. For better development we recommend not deploying *OpenKM.war* file it's better setting in *$JBOSS_HOME/server/default/deploy* some alias to *target/OpenKM* folder:



jboss-web.deployer	11 elementos	carpeta	mié 18 mar 2009 11:2
jbossws.sar	15 elementos	carpeta	jue 29 ene 2009 13:11
jms	9 elementos	carpeta	mié 29 oct 2008 12:56
jmx-console.war	11 elementos	carpeta	mié 29 oct 2008 12:56
juddi-service.sar	6 elementos	carpeta	jue 29 ene 2009 13:11
management	1 elemento	carpeta	lun 22 oct 2007 11:43
OpenKM.war	16 elementos	Enlace hacia carpeta	jue 21 ene 2010 12:32
ota_jaxb.war	3 elementos	carpeta	mar 03 feb 2009 13:43
uuid-key-generator.sar	2 elementos	carpeta	lun 22 oct 2007 11:43
bsh-deployer.xml	405 bytes	documento XML	lun 22 oct 2007 11:43
cache-invalidation-s...	2,0 KiB	documento XML	lun 22 oct 2007 11:43
client-deployer-servi...	1,9 KiB	documento XML	lun 22 oct 2007 11:43
database-ds.xml	5,3 KiB	documento XML	mar 25 ago 2009 13:3

To create symbolic links in windows use juntion.exe [1]

It's mandatory registering **sample-5.0-full.jar** into maven repository. File is into **ext-libs** folder. It must be executed some maven command to registering (then compilation will execute right). Take a look at install.sh, if sometimes is required some new library, the reference it'll be included on that file.

```
mvn install:install-file -DgroupId=com.openkm.extension -DartifactId=sample-full -Dversion=5.0 -Dpackaging=jar -Dfile=sample-5.0-full.jar
```

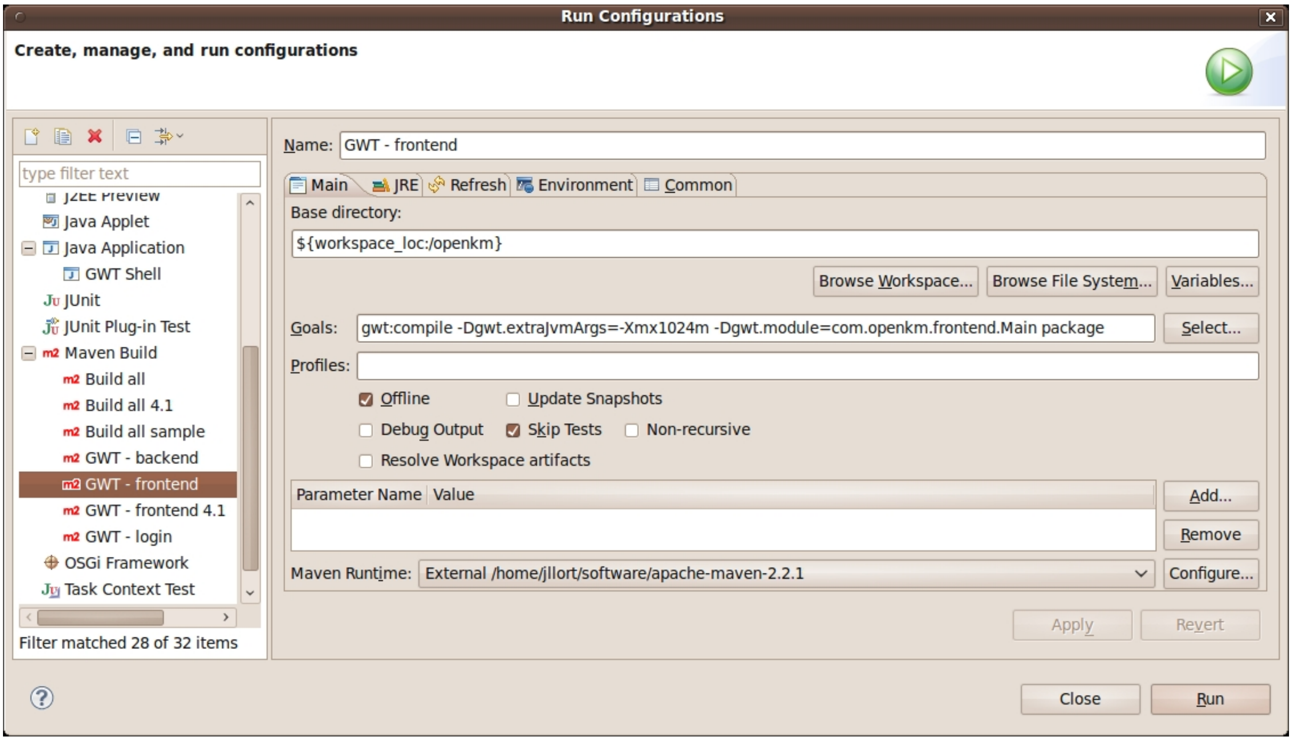
In case you're making strong changes in OpenKM UI (GWT) we recommend disabling pom compile directive

```
<plugin>
        <groupId>org.codehaus.mojo</groupId>
        <artifactId>gwt-maven-plugin</artifactId>
        <version>1.2</version>
        <executions>
          <execution>
            <goals>
              <goal>compile</goal>
              <!-- <goal>generateAsync</goal> -->
              <!-- <goal>test</goal> -->
            </goals>
          </execution>
        </executions>
        <configuration>
          <runTarget>com.openkm.frontend.Main/index.html</runTarget>
          <modules>
            <module>com.openkm.frontend.Main</module>
          </modules>
        </configuration>
      </plugin>
```

Is possible to compile GWT modules individually by this way:

```
$ mvn gwt:compile -Dgwt.module=com.openkm.frontend.Main
```

You might be interested in running frontend and backend GWT packaging from eclipse. Go to run configurations and add in maven build



Name: GWT - frontend

Base directory: ${workspace_loc:/openkm}

Goals: gwt:compile -Dgwt.extraJvmArgs=-Xmx1024m -Dgwt.module=com.openkm.frontend.Main package

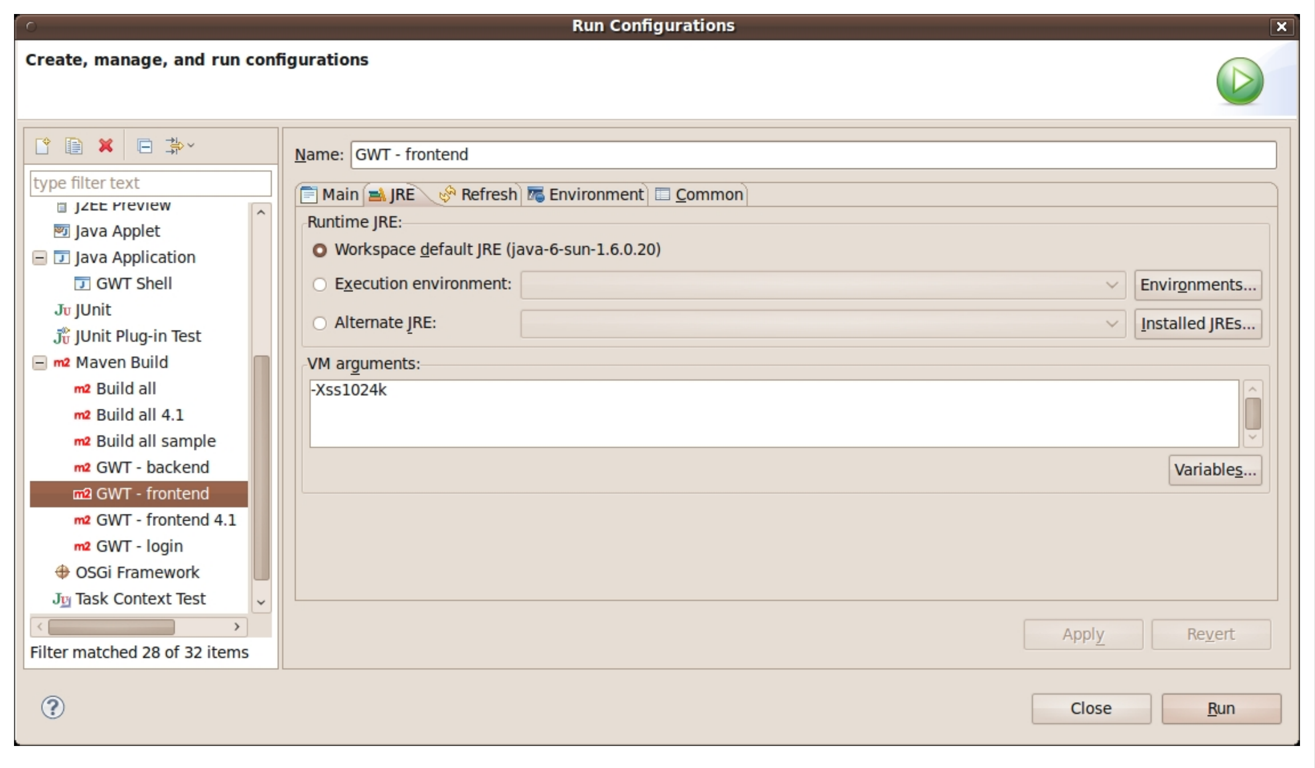
Profiles:

☑ Offline ☐ Update Snapshots

☐ Debug Output ☑ Skip Tests ☐ Non-recursive

☐ Resolve Workspace artifacts

Maven Runtime: External /home/jllort/software/apache-maven-2.2.1



Name: GWT - frontend

Runtime JRE:
● Workspace default JRE (java-6-sun-1.6.0.20)
○ Execution environment:
○ Alternate JRE:

VM arguments:
-Xss1024k

To achieve a faster compile we use to development Firefox, uncommenting gecko line on file **Main.gwt.xml** GWT compilation only for gecko browser runs fine. Don't forget at finish to comment it and try application with IE or other browsers.

```
<!-- Compile for Firefox only -->
<set-property name="user.agent" value="gecko"/>
```

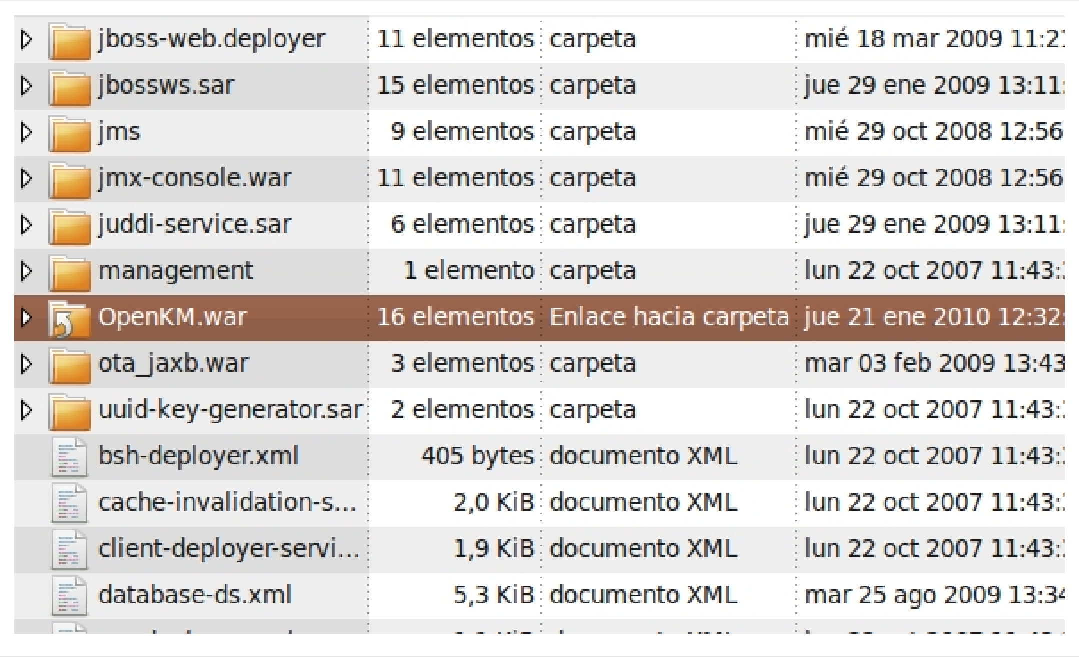
User contribution script

```
#!/bin/bash
# @author: Kenneth Walter

#Cleans and re-compiles the source files
mvn -Dmaven.test.skip=true clean gwt:compile install $*

#Set the TARGET_DIRECTORY to the path of your JBoss installation
TARGET_DIRECTORY=~/Downloads/jboss-4.2.3.GA/
FILE_TO_MOVE=target/OpenKM.war
#This will only attempt to replace the existing WAR if the new WAR exists
if [ -f $FILE_TO_MOVE ]
then
   echo 'Deploying WAR to JBoss Directory'
   cp -v $FILE_TO_MOVE $TARGET_DIRECTORY/server/default/deploy/.
   echo 'Done'
fi
```

More info about this script at Cannot create or use example extensions [2] forum thread.

Development tips OpenKM 4.0

We recommend downloading JBoss+OpenKM from SourceForge, remove *OpenKM.war* and develop with this JBoss server configuration. For better development we recommend not deploying *OpenKM.war* file it's better setting in *$JBOSS_HOME/server/default/deploy* some alias to *target/OpenKM* folder:



jboss-web.deployer	11 elementos	carpeta	mié 18 mar 2009 11:2
jbossws.sar	15 elementos	carpeta	jue 29 ene 2009 13:11
jms	9 elementos	carpeta	mié 29 oct 2008 12:56
jmx-console.war	11 elementos	carpeta	mié 29 oct 2008 12:56
juddi-service.sar	6 elementos	carpeta	jue 29 ene 2009 13:11
management	1 elemento	carpeta	lun 22 oct 2007 11:43
OpenKM.war	16 elementos	Enlace hacia carpeta	jue 21 ene 2010 12:32
ota_jaxb.war	3 elementos	carpeta	mar 03 feb 2009 13:43
uuid-key-generator.sar	2 elementos	carpeta	lun 22 oct 2007 11:43
bsh-deployer.xml	405 bytes	documento XML	lun 22 oct 2007 11:43
cache-invalidation-s...	2,0 KiB	documento XML	lun 22 oct 2007 11:43
client-deployer-servi...	1,9 KiB	documento XML	lun 22 oct 2007 11:43
database-ds.xml	5,3 KiB	documento XML	mar 25 ago 2009 13:3

To create symbolic links in windows use juntion.exe [1]

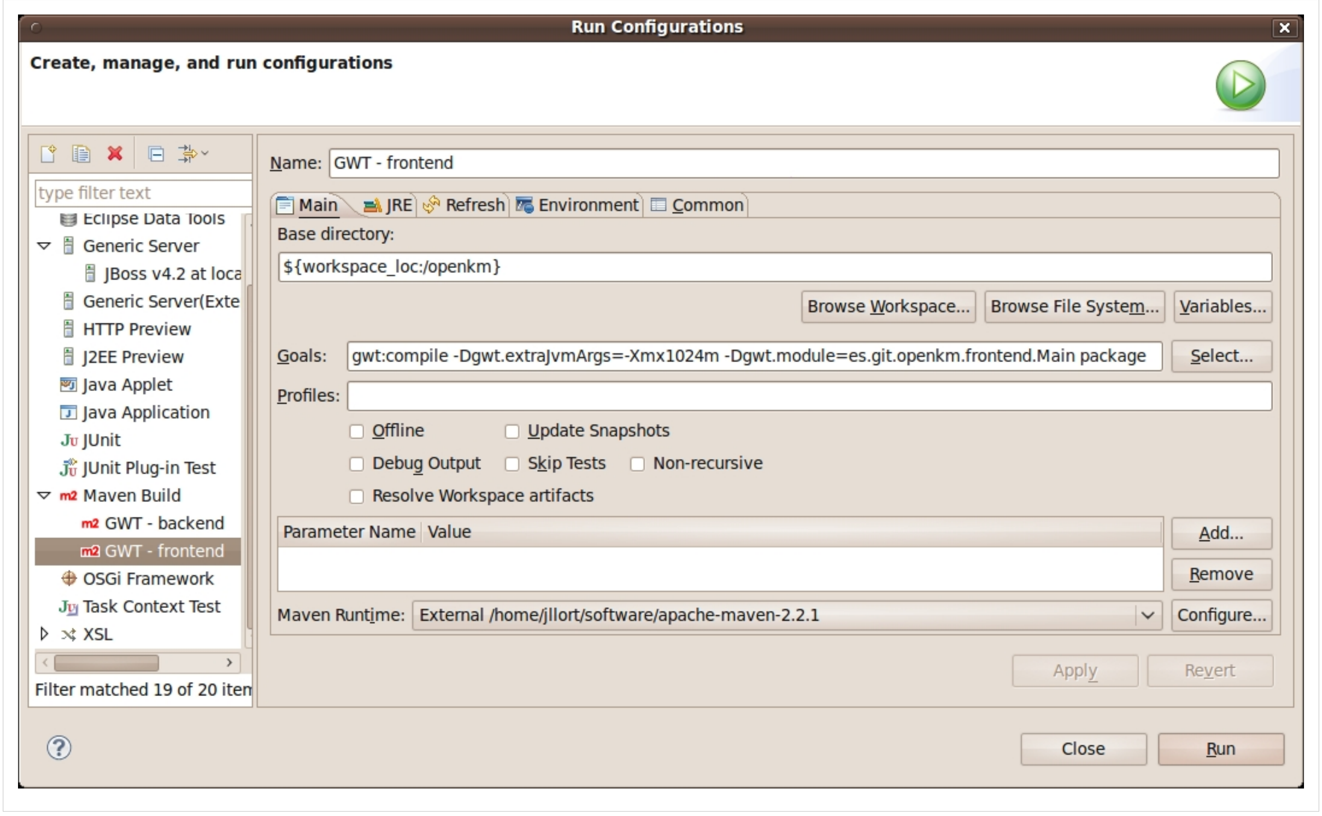
In case you're making strong changes in OpenKM UI (GWT) we recommend disabling pom compile directive

```
<plugin>
    <groupId>org.codehaus.mojo</groupId>
        <artifactId>gwt-maven-plugin</artifactId>
        <version>1.1</version>
        <configuration>
            <runTarget>es.git.openkm.frontend.Main/index.html</runTarget>
        </configuration>
        <executions>
          <execution>
            <goals>
               <!-- <goal>compile</goal> -->
              <!-- <goal>generateAsync</goal>  -->
              <goal>test</goal>
            </goals>
          </execution>
        </executions>
    </plugin>
```

Is possible to compile GWT modules individually by this way:

```
$ mvn gwt:compile -Dgwt.module=com.openkm.backend.Main
```

You might be interested in running frontend and backend GWT packaging from eclipse. Go to run configurations and add in maven build



Name: GWT - frontend

Base directory: ${workspace_loc:/openkm}

Goals: gwt:compile -Dgwt.extraJvmArgs=-Xmx1024m -Dgwt.module=es.git.openkm.frontend.Main package

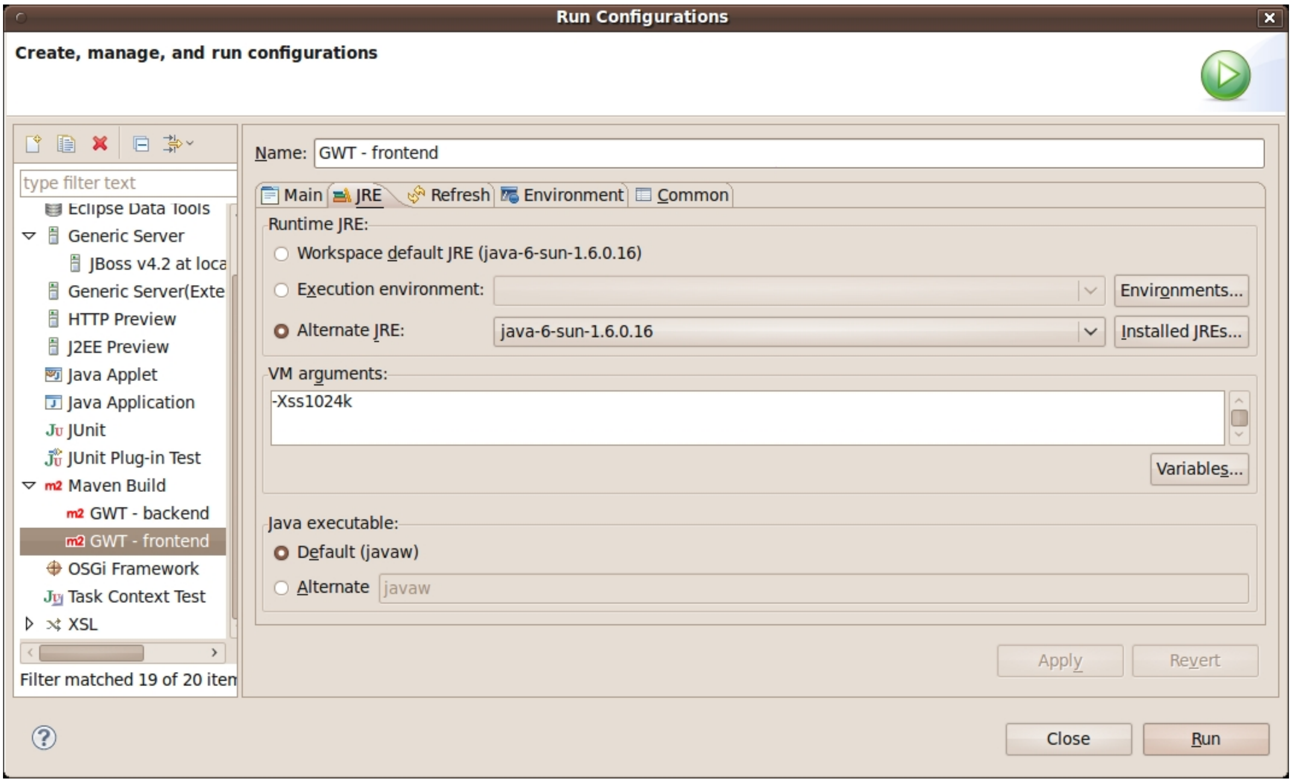
Profiles:

☐ Offline ☐ Update Snapshots

☐ Debug Output ☐ Skip Tests ☐ Non-recursive

☐ Resolve Workspace artifacts

Maven Runtime: External /home/jllort/software/apache-maven-2.2.1



Name: GWT - frontend

Runtime JRE:
○ Workspace default JRE (java-6-sun-1.6.0.16)
○ Execution environment:
● Alternate JRE: java-6-sun-1.6.0.16

VM arguments:
-Xss1024k

Java executable:
● Default (javaw)
○ Alternate javaw

To achieve a faster compile we use to development Firefox, uncommenting gecko line on file Main.gwt.xml GWT compilation only for gecko browser runs fine. Don't forget at finish to comment it and try application with IE or other browsers. Now are two files called Main.gwt.xml one for frontend and other to backend.

```
<!-- Compile for Firefox only -->
<set-property name="user.agent" value="gecko"/>
```

Debug with tomcat

You can debug your OpenKM installation using the Tomcat logging facility. This is an useful thing when you have problems with your configuration. Default OpenKM installation tries to log important events like errors and warnings. Is possible to change this configuration editing the file *$TOMCAT_HOME/conf/logging.properties*.

Default Tomcat log configuration can generate a lot of messages. These files are stored at *$TOMCAT_HOME/logs*.

Debugging with GWT

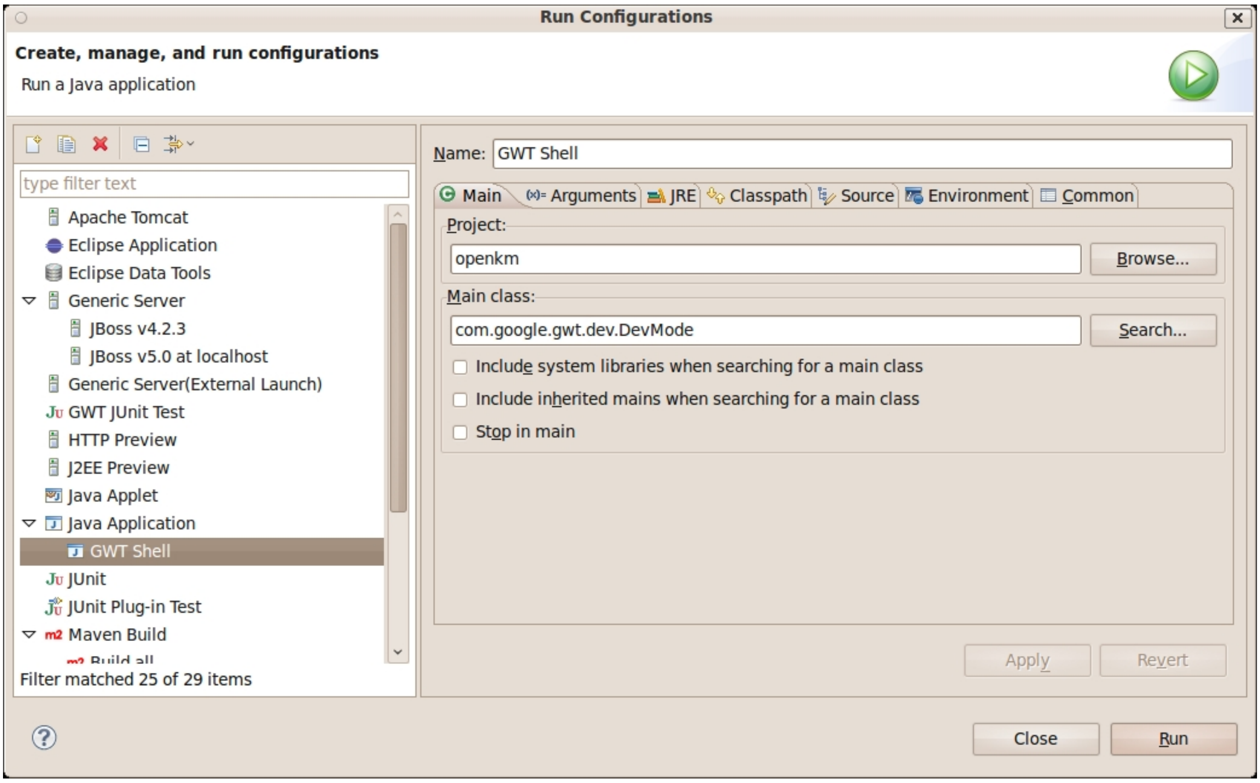
Go to **run configurations** and add new **java application** configuration as you can see in this screenshoot.

First of all, open the pom.xml file and uncomment this section:

```
<!-- Only for development -->
<dependency>
   <groupId>com.google.gwt</groupId>
   <artifactId>gwt-dev</artifactId>
   <version>${gwt.version}</version>
   <scope>provided</scope>
</dependency>
```

Set the Main class:

```
com.google.gwt.dev.DevMode
```



Run Configurations

Create, manage, and run configurations

Run a Java application

Name: GWT Shell

Project: openkm

Main class: com.google.gwt.dev.DevMode

☐ Include system libraries when searching for a main class

☐ Include inherited mains when searching for a main class

☐ Stop in main

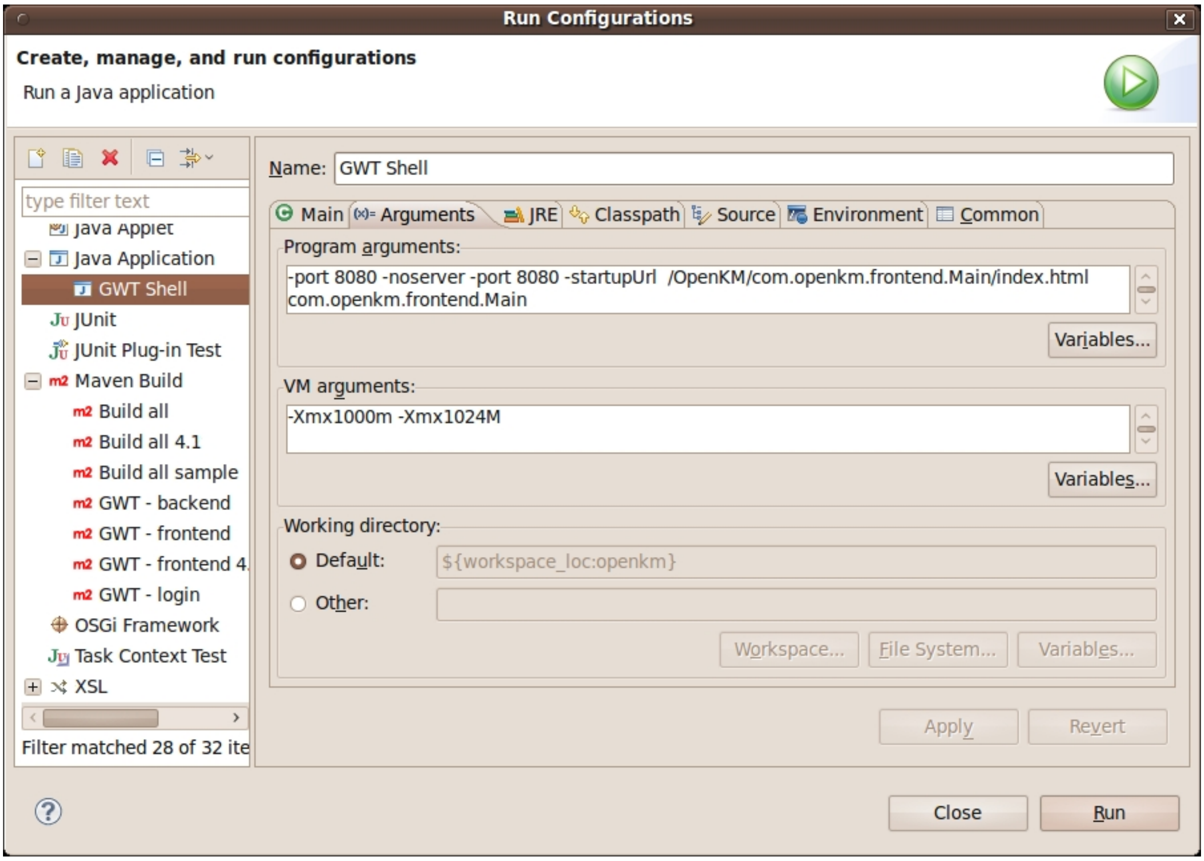
Filter matched 25 of 29 items

Set program arguments

```
-port 8080 -noserver -startupUrl  /OpenKM/frontend/index.jsp com.openkm.frontend.Main
```

and the VM arguments

```
-Xms1024m -Xmx1024M
```



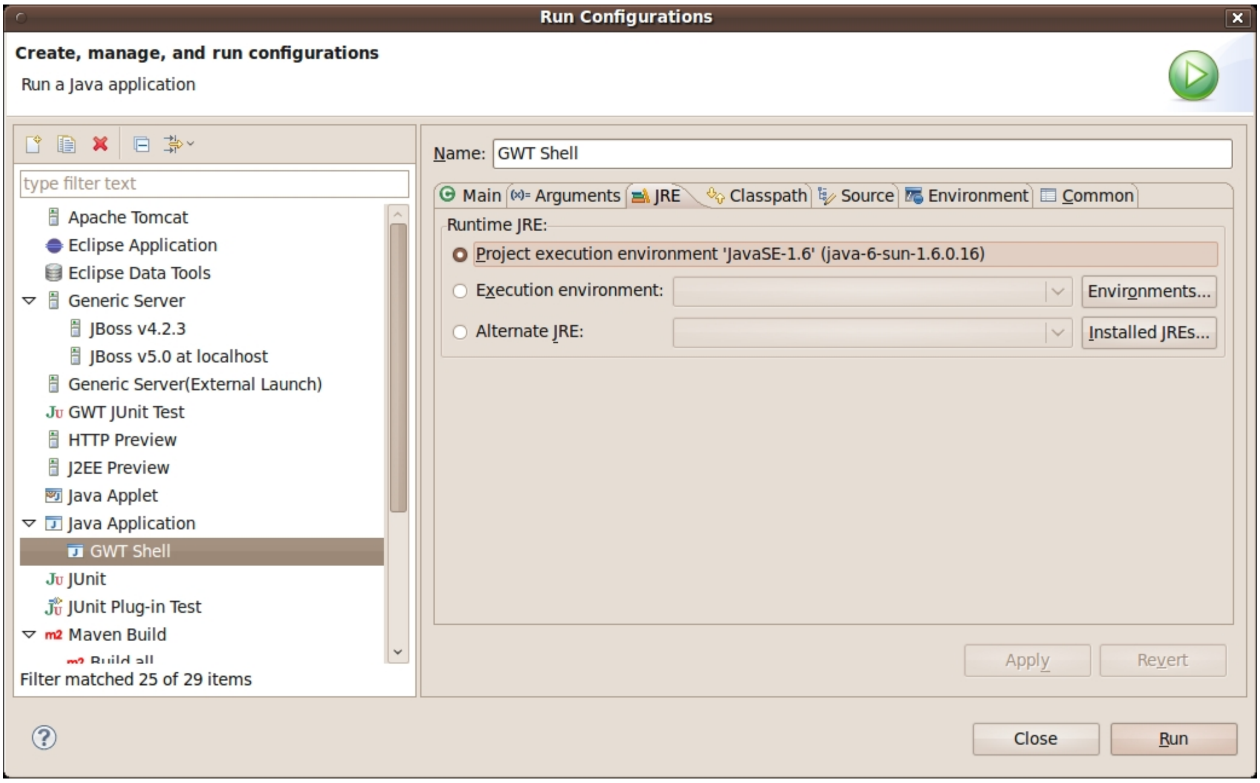
Name: GWT Shell

Program arguments:
-port 8080 -noserver -port 8080 -startupUrl /OpenKM/com.openkm.frontend.Main/index.html com.openkm.frontend.Main

VM arguments:
-Xmx1000m -Xmx1024M

Working directory:
● Default: ${workspace_loc:openkm}
○ Other:

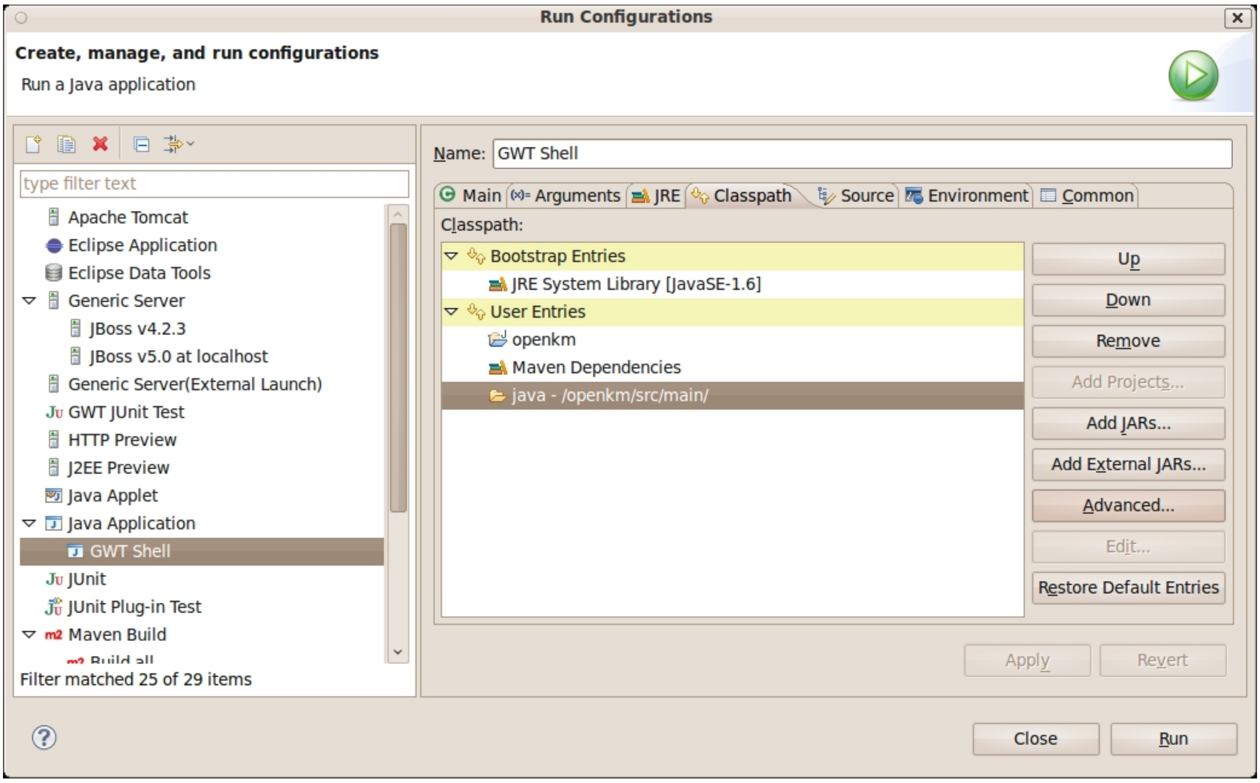
Set the jdk 1.6



Name: GWT Shell

Runtime JRE:
● Project execution environment 'JavaSE-1.6' (java-6-sun-1.6.0.16)
○ Execution environment:
○ Alternate JRE:

Set the class path. Pay special **attention** in **adding** folder **/openkm/src/main** (using button advanced and add folder option)

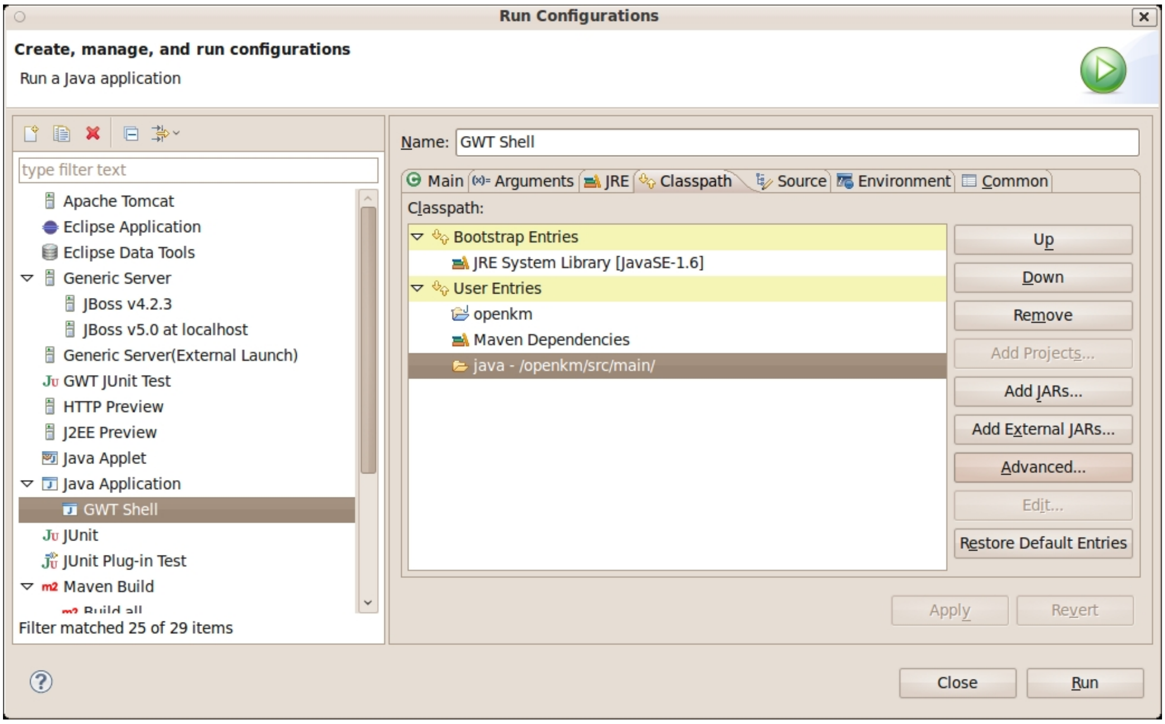


Name: GWT Shell

Classpath:

- Bootstrap Entries
  - JRE System Library [JavaSE-1.6]
- User Entries
  - openkm
  - Maven Dependencies
  - java - /openkm/src/main/

Now you can run (better if you run in debug mode, you can set breakpoints in your eclipse code). It'll appear some screen like this:



Name: GWT Shell

Classpath:

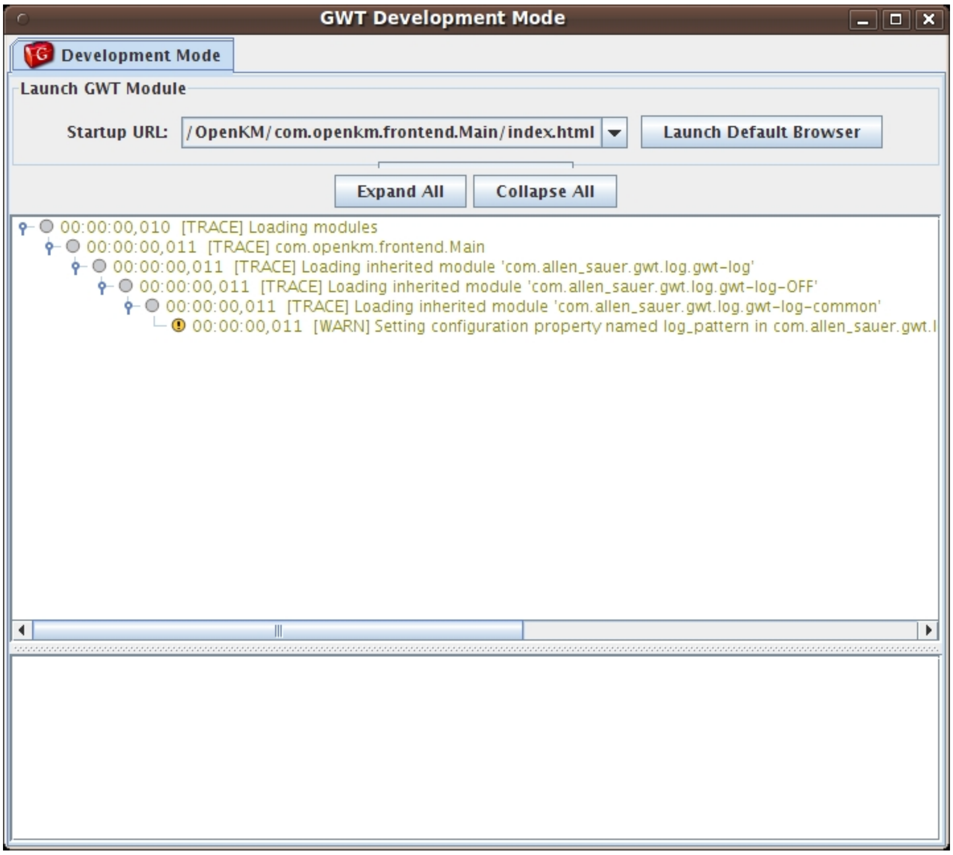
- Bootstrap Entries
  - JRE System Library [JavaSE-1.6]
- User Entries
  - openkm
  - Maven Dependencies
  - java - /openkm/src/main/

To **debug login** put in your browser

```
http://localhost:8080/OpenKM/frontend/index.jsp?gwt.codesvr=127.0.1.1:9997
```

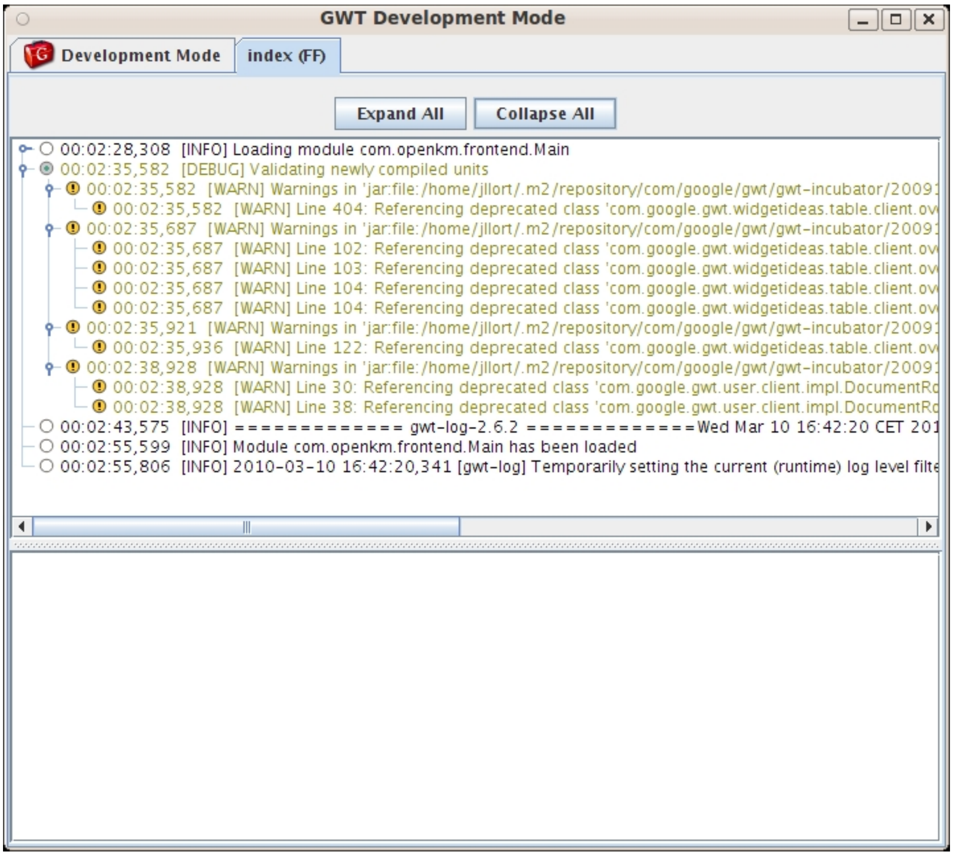
To **debug frontend** first authenticate and after it, put in your browser

```
http://localhost:8080/OpenKM/frontend/index.html?gwt.codesvr=127.0.1.1:9997
```



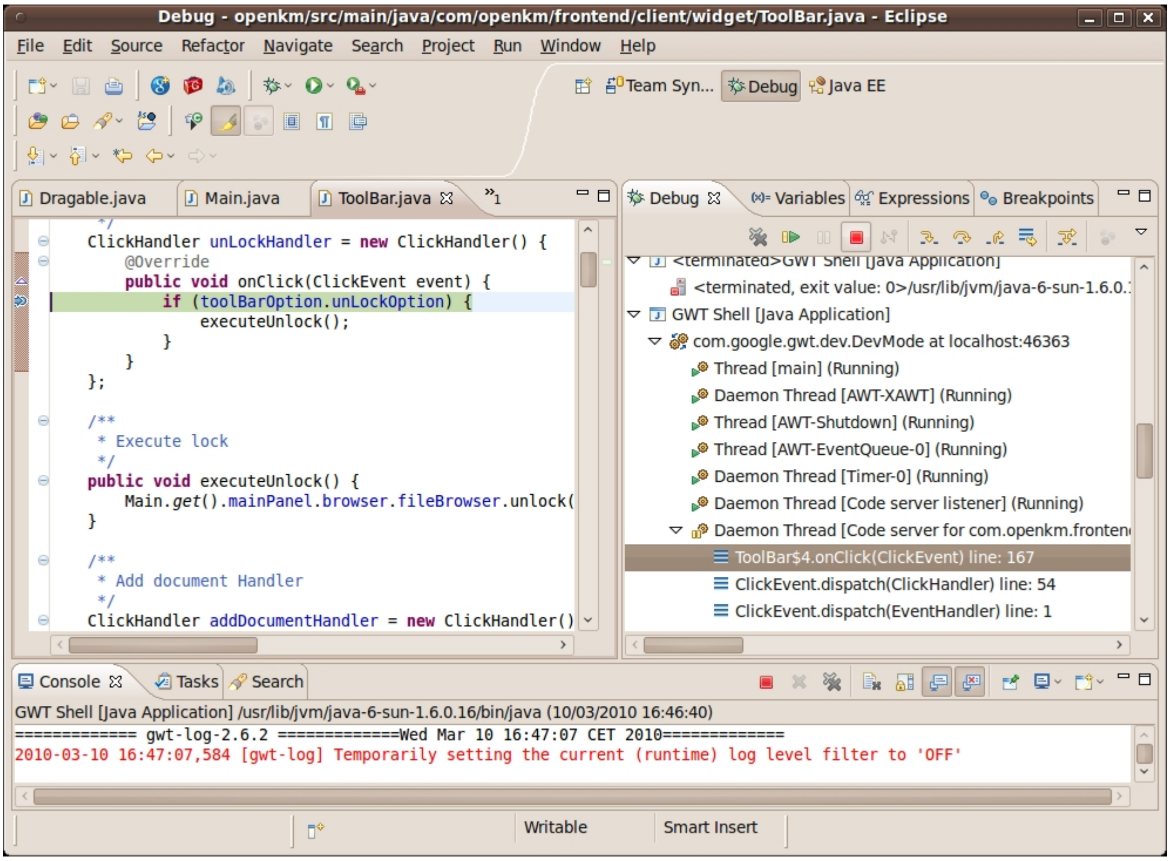
Startup URL: /OpenKM/com.openkm.frontend.Main/index.html

```
00:00:00,010  [TRACE] Loading modules
  00:00:00,011  [TRACE] com.openkm.frontend.Main
    00:00:00,011  [TRACE] Loading inherited module 'com.allen_sauer.gwt.log.gwt-log'
      00:00:00,011  [TRACE] Loading inherited module 'com.allen_sauer.gwt.log.gwt-log-OFF'
        00:00:00,011  [TRACE] Loading inherited module 'com.allen_sauer.gwt.log.gwt-log-common'
          00:00:00,011  [WARN] Setting configuration property named log_pattern in com.allen_sauer.gwt.l
```



```
00:02:28,308  [INFO] Loading module com.openkm.frontend.Main
00:02:35,582  [DEBUG] Validating newly compiled units
  00:02:35,582  [WARN] Warnings in 'jar:file:/home/jllort/.m2/repository/com/google/gwt/gwt-incubator/2009
    00:02:35,582  [WARN] Line 404: Referencing deprecated class 'com.google.gwt.widgetideas.table.client.ov
  00:02:35,687  [WARN] Warnings in 'jar:file:/home/jllort/.m2/repository/com/google/gwt/gwt-incubator/2009
    00:02:35,687  [WARN] Line 102: Referencing deprecated class 'com.google.gwt.widgetideas.table.client.ov
    00:02:35,687  [WARN] Line 103: Referencing deprecated class 'com.google.gwt.widgetideas.table.client.ov
    00:02:35,687  [WARN] Line 104: Referencing deprecated class 'com.google.gwt.widgetideas.table.client.ov
    00:02:35,687  [WARN] Line 104: Referencing deprecated class 'com.google.gwt.widgetideas.table.client.ov
  00:02:35,921  [WARN] Warnings in 'jar:file:/home/jllort/.m2/repository/com/google/gwt/gwt-incubator/2009
    00:02:35,936  [WARN] Line 122: Referencing deprecated class 'com.google.gwt.widgetideas.table.client.ov
  00:02:38,928  [WARN] Warnings in 'jar:file:/home/jllort/.m2/repository/com/google/gwt/gwt-incubator/2009
    00:02:38,928  [WARN] Line 30: Referencing deprecated class 'com.google.gwt.user.client.impl.DocumentR
    00:02:38,928  [WARN] Line 38: Referencing deprecated class 'com.google.gwt.user.client.impl.DocumentR
00:02:43,575  [INFO] ============= gwt-log-2.6.2 =============Wed Mar 10 16:42:20 CET 201
00:02:55,599  [INFO] Module com.openkm.frontend.Main has been loaded
00:02:55,806  [INFO] 2010-03-10 16:42:20,341 [gwt-log] Temporarily setting the current (runtime) log level filt
```

You debug into eclipse, see variables values etc...

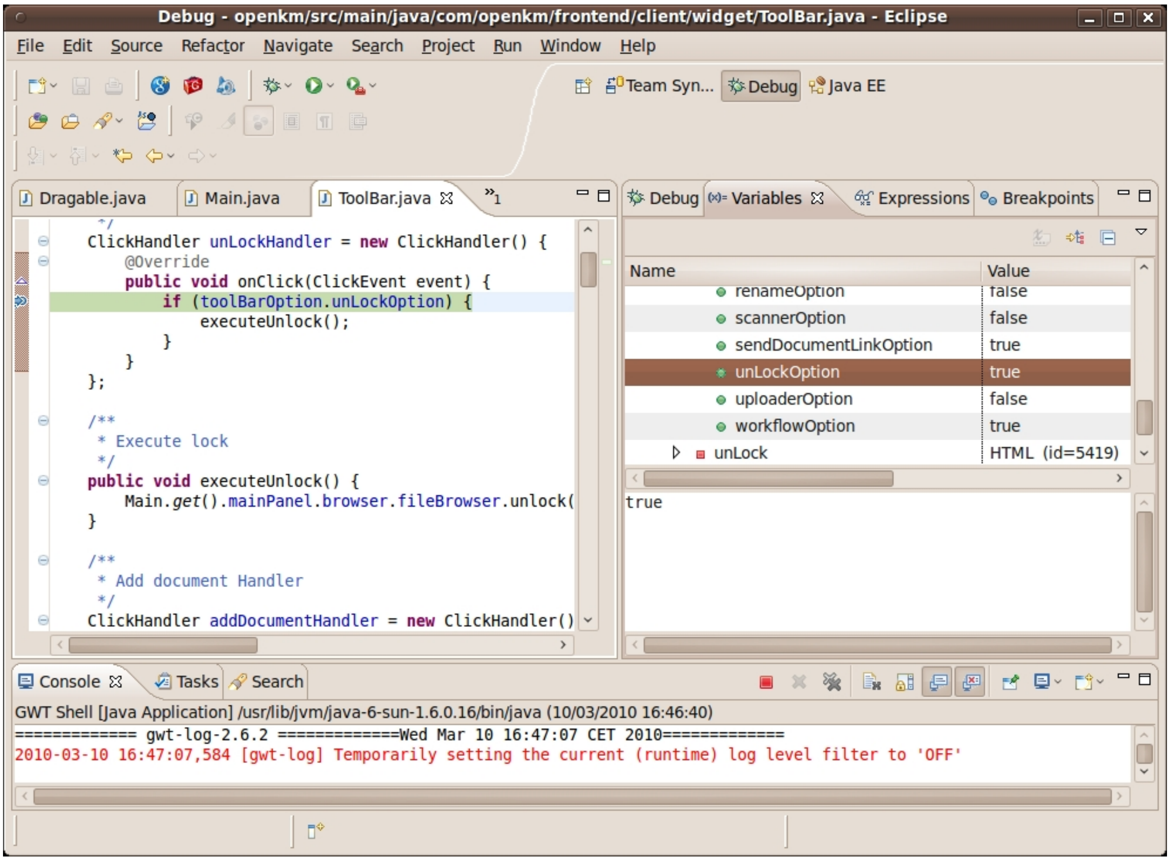


Debug - openkm/src/main/java/com/openkm/frontend/client/widget/ToolBar.java - Eclipse
File Edit Source Refactor Navigate Search Project Run Window Help
Team Syn... Debug Java EE
Dragable.java Main.java ToolBar.java
Debug Variables Expressions Breakpoints
Console Tasks Search
Writable Smart Insert

```
ClickHandler unLockHandler = new ClickHandler() {
    @Override
    public void onClick(ClickEvent event) {
        if (toolBarOption.unLockOption) {
            executeUnlock();
        }
    }
};

/**
 * Execute lock
 */
public void executeUnlock() {
    Main.get().mainPanel.browser.fileBrowser.unlock(
}

/**
 * Add document Handler
 */
ClickHandler addDocumentHandler = new ClickHandler()
```

```
<terminated>GWT Shell [Java Application]
    <terminated, exit value: 0>/usr/lib/jvm/java-6-sun-1.6.0.
GWT Shell [Java Application]
    com.google.gwt.dev.DevMode at localhost:46363
        Thread [main] (Running)
        Daemon Thread [AWT-XAWT] (Running)
        Thread [AWT-Shutdown] (Running)
        Thread [AWT-EventQueue-0] (Running)
        Daemon Thread [Timer-0] (Running)
        Daemon Thread [Code server listener] (Running)
        Daemon Thread [Code server for com.openkm.fronten
            ToolBar$4.onClick(ClickEvent) line: 167
            ClickEvent.dispatch(ClickHandler) line: 54
            ClickEvent.dispatch(EventHandler) line: 1
```

```
GWT Shell [Java Application] /usr/lib/jvm/java-6-sun-1.6.0.16/bin/java (10/03/2010 16:46:40)
============= gwt-log-2.6.2 =============Wed Mar 10 16:47:07 CET 2010=============
2010-03-10 16:47:07,584 [gwt-log] Temporarily setting the current (runtime) log level filter to 'OFF'
```



Debug - openkm/src/main/java/com/openkm/frontend/client/widget/ToolBar.java - Eclipse
File Edit Source Refactor Navigate Search Project Run Window Help
Team Syn... Debug Java EE
Dragable.java Main.java ToolBar.java
Debug Variables Expressions Breakpoints
Console Tasks Search

```
ClickHandler unLockHandler = new ClickHandler() {
    @Override
    public void onClick(ClickEvent event) {
        if (toolBarOption.unLockOption) {
            executeUnlock();
        }
    }
};

/**
 * Execute lock
 */
public void executeUnlock() {
    Main.get().mainPanel.browser.fileBrowser.unlock(
}

/**
 * Add document Handler
 */
ClickHandler addDocumentHandler = new ClickHandler()
```

Name	Value
renameOption	false
scannerOption	false
sendDocumentLinkOption	true
unLockOption	true
uploaderOption	false
workflowOption	true
unLock	HTML (id=5419)

```
true
```

```
GWT Shell [Java Application] /usr/lib/jvm/java-6-sun-1.6.0.16/bin/java (10/03/2010 16:46:40)
============= gwt-log-2.6.2 =============Wed Mar 10 16:47:07 CET 2010=============
2010-03-10 16:47:07,584 [gwt-log] Temporarily setting the current (runtime) log level filter to 'OFF'
```

OpenKM version 4.1 and older

Set the program arguments

```
-port 8080 -noserver -startupUrl  /OpenKM/com.openkm.frontend.Main/index.html com.openkm.frontend.Main
```

To **debug login** put in your browser

```
http://localhost:8080/OpenKM/com.openkm.login.Main/index.jsp?gwt.codesvr=127.0.1.1:9997
```

To **debug frontend** first authenticate and after it, put in your browser

```
http://localhost:8080/OpenKM/com.openkm.frontend.Main/index.html?gwt.codesvr=127.0.1.1:9997
```

Profiling OpenKM

You can profiling local and remote Java applications using Java VisualVM [1]. For local application is easy because it detect automatically them and show in the listing. For remote application you need to do a little work.

First of all add this line to **$TOMCAT_HOME/bin/setenv.sh** file (or **setenv.bat** if using Windows):

```
CATALINA_OPTS="$CATALINA_OPTS -Dcom.sun.management.jmxremote=true"
CATALINA_OPTS="$CATALINA_OPTS -Dcom.sun.management.jmxremote.port=9090"
CATALINA_OPTS="$CATALINA_OPTS -Dcom.sun.management.jmxremote.ssl=false"
CATALINA_OPTS="$CATALINA_OPTS
-Dcom.sun.management.jmxremote.authenticate=false"
CATALINA_OPTS="$CATALINA_OPTS -Djava.rmi.server.hostname=127.0.0.1"
```

The **java.rmi.server.hostname** should match the internal IP of the remote server, so replace **127.0.0.1** by the right IP.

In order to connect to this remote Tomcat instance, go to **File > Add JMX Connection...** and fill the hostname and port. In this case the port is 9090 but you can set another one using the **com.sun.management.jmxremote.port** property.

Authentication

To enhance security, enable authentication:

```
CATALINA_OPTS="$CATALINA_OPTS
-Dcom.sun.management.jmxremote.authenticate=true"
CATALINA_OPTS="$CATALINA_OPTS
-Dcom.sun.management.jmxremote.password.file=$CATALINA_HOME/conf/jmxremote.password"
CATALINA_OPTS="$CATALINA_OPTS
-Dcom.sun.management.jmxremote.access.file=$CATALINA_HOME/conf/jmxremote.access"
```

Edit the access allow file $CATALINA_HOME/conf/jmxremote.access:

```
monitorRole readonly
controlRole readwrite
```

To show remote threads you need the **readwrite** permission.

Edit the password file $CATALINA_BASE/conf/jmxremote.password :

```
monitorRole tomcat
controlRole tomcat
```

And finally, restrict access to these files:

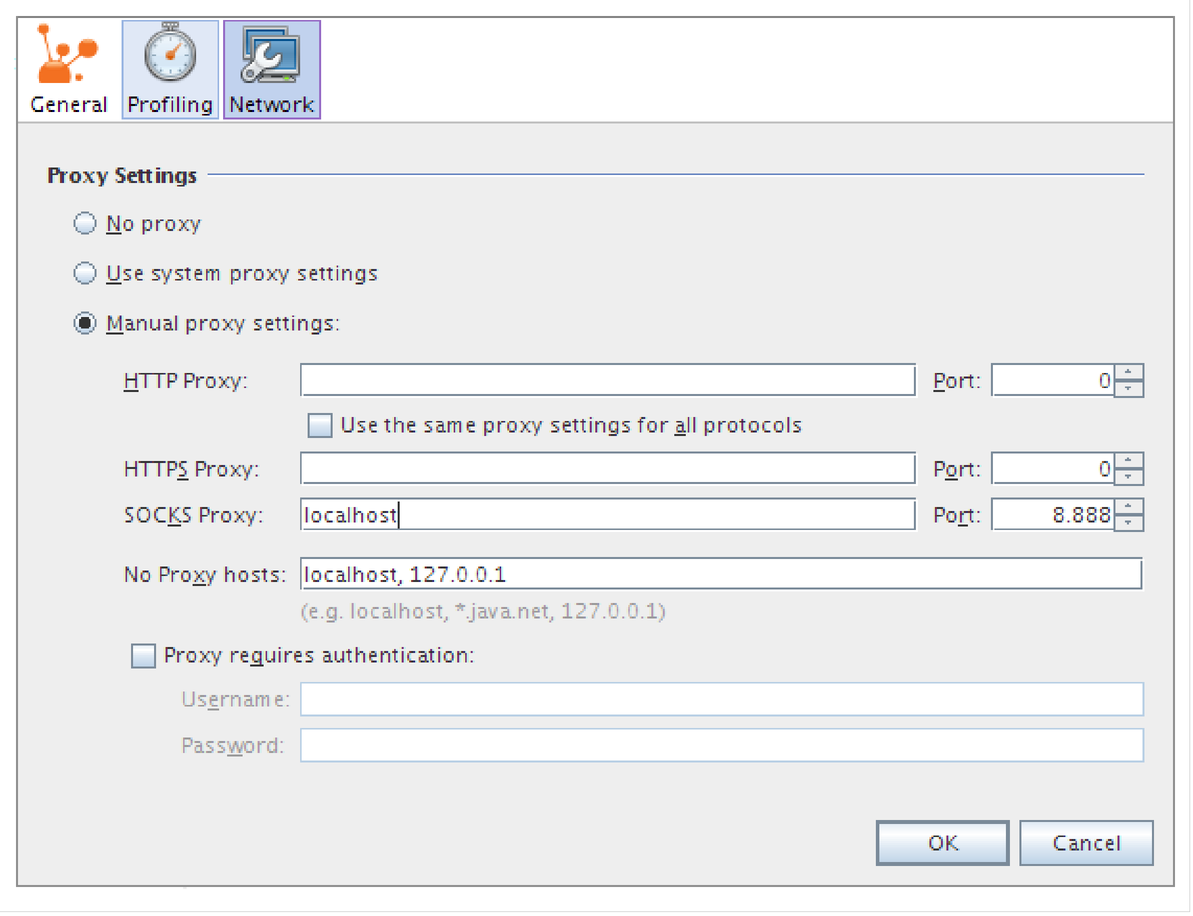
```
$ chmod 600 $CATALINA_HOME/conf/jmxremote.access
$ chmod 600 $CATALINA_HOME/conf/jmxremote.password
```

Monitoring Remote JVM Over SSH

The easiest way to connect to the remote JMX is to use a SOCKS proxy [2].

```
$ ssh user@external.server.ip -D 8888
```

You need to configure VisualVM to use this proxy. Go to **Tools** > **Options** > **Network** and edit **Manual Proxy Settings**. Configure SOCKS Proxy at *localhost* and port *8888*.



Proxy Settings

○ No proxy

○ Use system proxy settings

● Manual proxy settings:

HTTP Proxy: Port: 0

☐ Use the same proxy settings for all protocols

HTTPS Proxy: Port: 0

SOCKS Proxy: localhost Port: 8.888

No Proxy hosts: localhost, 127.0.0.1

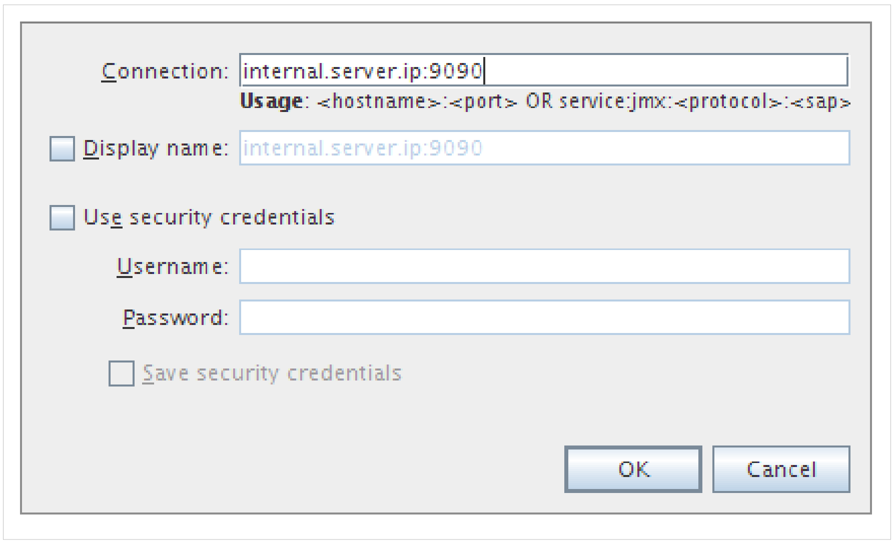
(e.g. localhost, *.java.net, 127.0.0.1)

☐ Proxy requires authentication:

Username:

Password:

Now you can connect VisualVM to the remote target. Go to **File** > **Add JMX Connection...** and type the IP or hostname of the remote machine and the configured JMX port (in our sample configuration is 9090).



Connection: internal.server.ip:9090

Usage: <hostname>:<port> OR service:jmx:<protocol>:<sap>

☐ Display name: internal.server.ip:9090

☐ Use security credentials

Username:

Password:

☐ Save security credentials

More info:

- Profiling With VisualVM, Part 1 [3]
- Profiling With VisualVM, Part 2 [4]
- Connecting Visual VM to Tomcat 7 [5]
- Monitoring and Managing Tomcat 7 [6]
- Remote Profiling of JBoss using VisualVM [7]
- Troubleshooting application performance with VisualVM [8]
- Monitoring of Tomcat with VisualVM and VisualGC [9]
- Using VisualVM to fix live Tomcat and JVM problems [10]

HPROF

Java includes HPROF, a profiler which collect application runtime information. HPROF is capable of presenting CPU usage, heap allocation statistics, and monitor contention profiles.

For example, can collect CPU usage information by sampling threads. Add this line to **$TOMCAT_HOME/bin/setenv.sh** file (or **setenv.bat** if using Windows):

```
CATALINA_OPTS="$CATALINA_OPTS -agentlib:hprof=cpu=samples"
```

When Tomcat starts you can see a file called **java.hprof.txt**. The CPU profiling info will be dumped to this file once Tomcat process is stopped.

More info:

- HPROF: A Heap/CPU Profiling Tool in J2SE 5.0 [11]
- HPjmeter [12]

References

[1] http://docs.oracle.com/javase/6/docs/technotes/guides/visualvm/index.html

[2] http://en.wikipedia.org/wiki/SOCKS

[3] https://blogs.oracle.com/nbprofiler/entry/profiling_with_visualvm_part_1

[4] https://blogs.oracle.com/nbprofiler/entry/profiling_with_visualvm_part_2

[5] http://blog.markwshead.com/1129/connecting-visual-vm-to-tomcat-7/

[6] http://tomcat.apache.org/tomcat-7.0-doc/monitoring.html

[7] http://hillert.blogspot.com.es/2010/01/remote-profiling-of-jboss-using.html

[8] http://www.skill-guru.com/blog/2010/11/11/troubleshooting-application-performance-with-visualvm/

[9] http://techblog.zabuchy.net/2012/monitoring-of-tomcat-with-visualvm-and-visualgc/

[10] http://blog.tty.nl/2010/09/03/using-visualvm-to-fix-live-tomcat-and-jvm-problems/

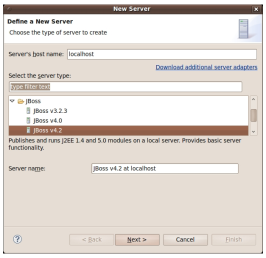
[11] http://java.sun.com/developer/technicalArticles/Programming/HPROF.html

[12] https://h20392.www2.hp.com/portal/swdepot/displayProductInfo.do?productNumber=HPJMETER

OpenKM 5.x and older

Configure JBoss server in Eclipse

Go to servers tab Using contextual menus select **Add / server**. Select JBoss server:



Define a New Server

Choose the type of server to create

Server's host name: localhost

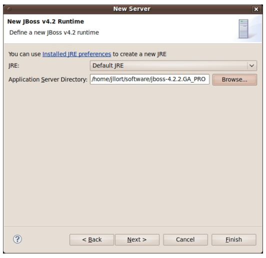
Download additional server adapters

Select the server type: type filter text

- JBoss
  - JBoss v3.2.3
  - JBoss v4.0
  - JBoss v4.2

Publishes and runs J2EE 1.4 and 5.0 modules on a local server. Provides basic server functionality.

Server name: JBoss v4.2 at localhost



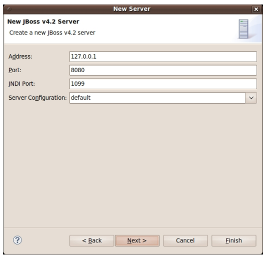
New JBoss v4.2 Runtime

Define a new JBoss v4.2 runtime

You can use Installed JRE preferences to create a new JRE

JRE: Default JRE

Application Server Directory: /home/jllort/software/jboss-4.2.2.GA_PRO



New JBoss v4.2 Server

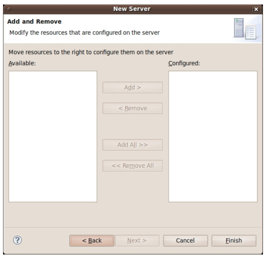
Create a new JBoss v4.2 server

Address: 127.0.0.1

Port: 8080

JNDI Port: 1099

Server Configuration: default



Add and Remove

Modify the resources that are configured on the server

Move resources to the right to configure them on the server

Debugging remote server

If you got problems on production environment this configuration is specially useful.

OpenKM 6.x - Tomcat

Add the following options when the JVM is started:

```
-Xdebug -Xrunjdwp:transport=dt_socket,address=8000,server=y,suspend=n
```

If you are using shell scripts to start Tomcat, start it with the following command:

```
$ $TOMCAT_HOME/bin/catalina.sh jpda start
```

It will start Tomcat so that a remote debugger can be connected to port 8000.

For more info read Tomcat developing: How do I configure Tomcat to support remote debugging? [1].

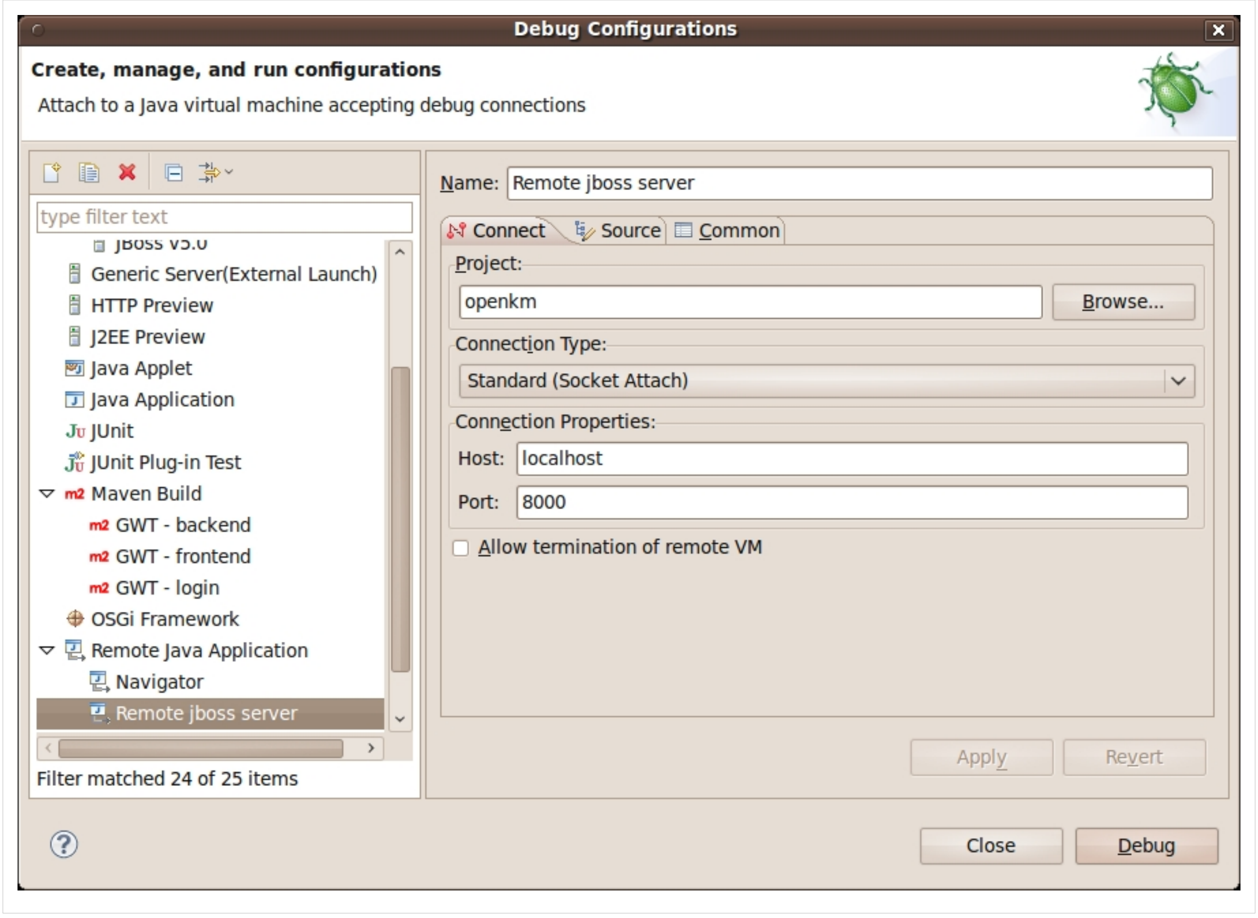
OpenKM 5.1 - JBoss

Add this line at the top of *$JBOSS_HOME/bin/run.sh* script:

```
JAVA_OPTS="-Xmx2000m -Djava.awt.headless=true -Xdebug -Xnoagent
-Xrunjdwp:transport=dt_socket,server=y,suspend=n,address=8000"
```

Configure Eclipse

Select the project and the host. Include in source your OpenKM java project. You can debug normally, setting breakpoints etc...



Debug Configurations

Create, manage, and run configurations

Attach to a Java virtual machine accepting debug connections

Name: Remote jboss server

Project: openkm

Connection Type: Standard (Socket Attach)

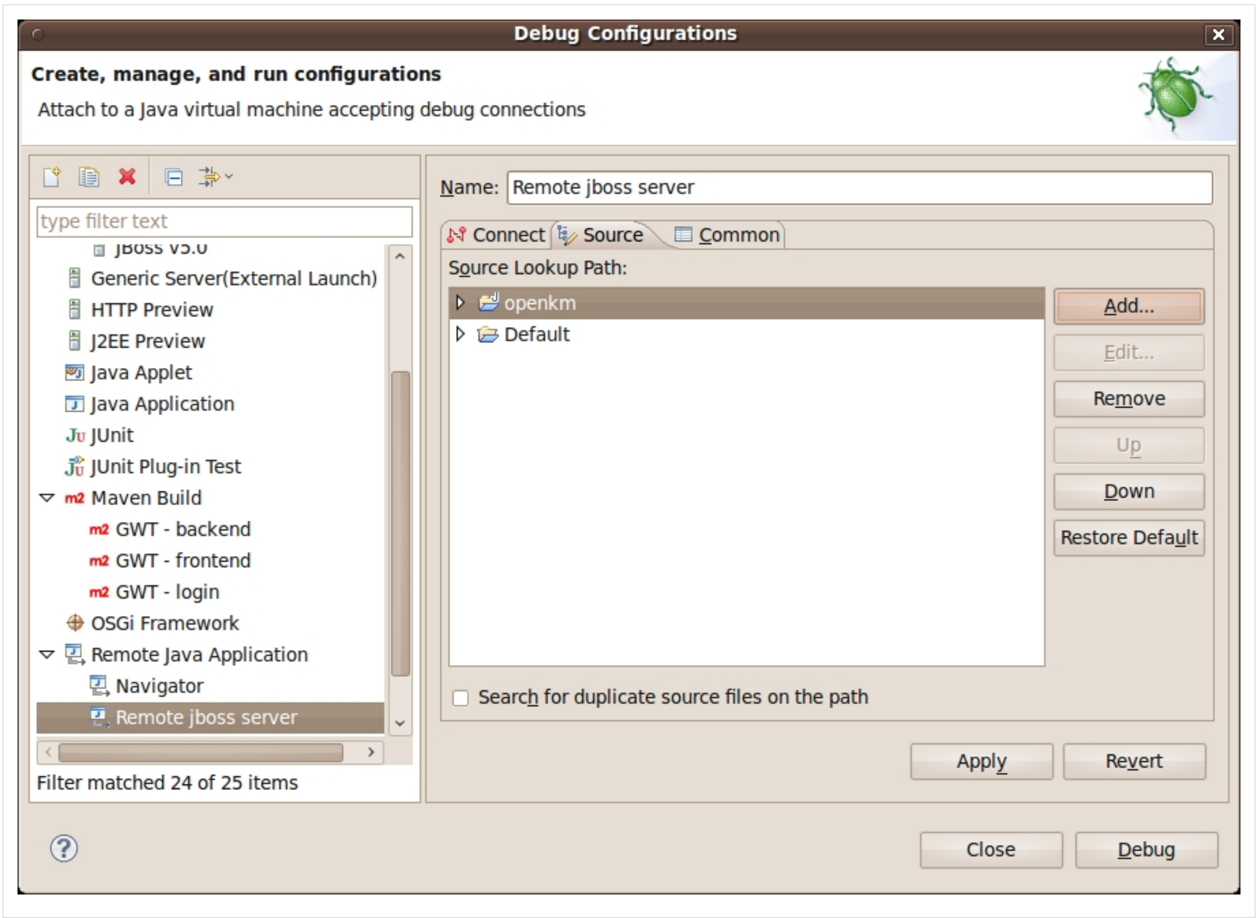
Connection Properties:

Host: localhost

Port: 8000

☐ Allow termination of remote VM

Filter matched 24 of 25 items



Debug Configurations

Create, manage, and run configurations

Attach to a Java virtual machine accepting debug connections

Name: Remote jboss server

Source Lookup Path:

- openkm
- Default

☐ Search for duplicate source files on the path

Filter matched 24 of 25 items

See also Como hacer debug remoto en Tomcat desde Eclipse [2].

References

[1] http://wiki.apache.org/tomcat/FAQ/Developing#Q1

[2] http://fherrerav.blogspot.com.es/2007/06/como-hacer-debug-remoto-en-tomcat-desde.html

Browsing embeded database

OpenKM comes with FOUR embeded databases that are defined in poll openkm-ds.xml into $JBOSS_HOME/server/default/deploy directory

- OKMActivity (used for log info)
- OKMAuth (user for authentication purpose)
- OKMDashboardStats (used for dashboard info)
- OKMWorkflow (used to store workflow info)

A simply way to browse it

Check the JAVA_OPTS in bin/run.sh. java.awt.headless has to be false to use the embedded database browser.

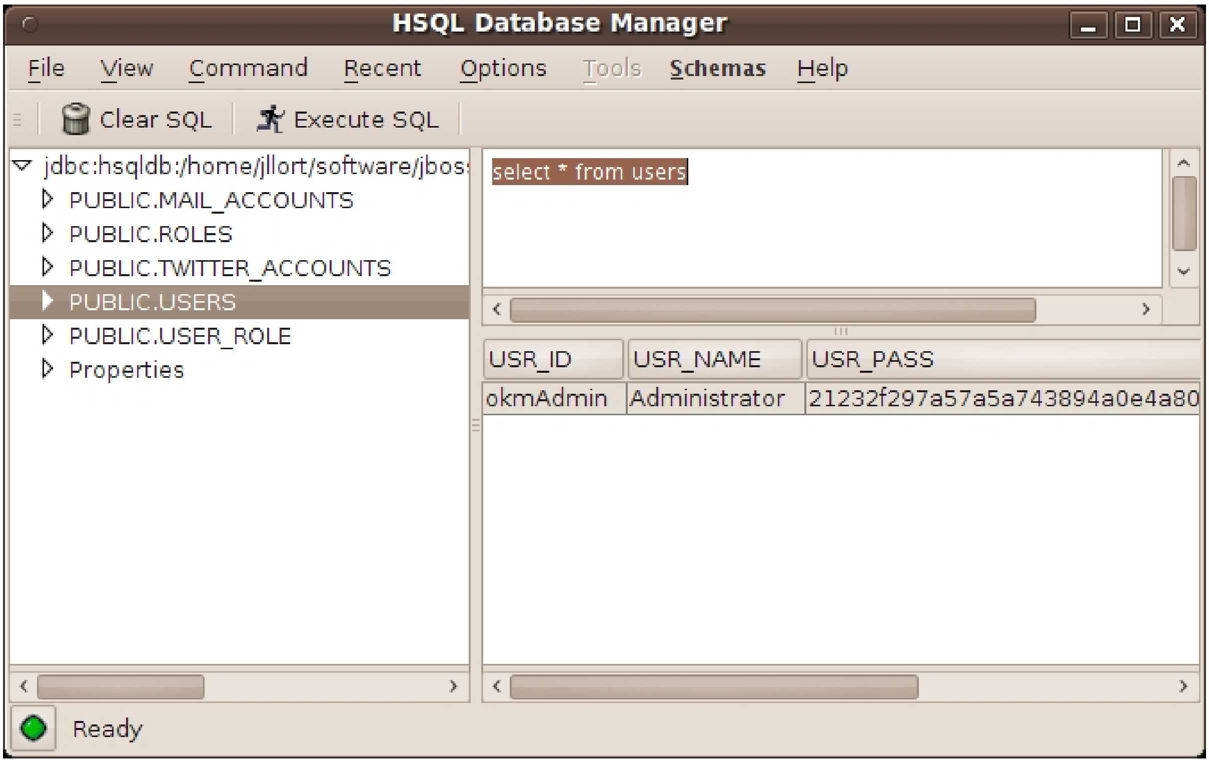
An example JAVA_OPTS could be:

```
JAVA_OPTS="-Xms256m -Xmx1024m -XX:PermSize=64m -XX:MaxPermSize=128m -Djava.awt.headless=false"
```

Open the url from server http://localhost:8080/jmx-console/. You'll see there three services defined under JBoss:

```
database=OKMActivity,service=Hypersonic
database=OKMAuth,service=Hypersonic
database=OKMDashboardStats,service=Hypersonic
database=OKMWorkflow,service=Hypersonic
database=localDB,service=Hypersonic
```

Select one datasource, then look for **void startDatabaseManager()** and press the **Invoke** button. You will see an screen like this:



```
select * from users
```

USR_ID	USR_NAME	USR_PASS
okmAdmin	Administrator	21232f297a57a5a743894a0e4a80

More info at Starting hsqldb manager on JBoss [1].

References

[1] http://docs.jboss.org/jbpm/v3/userguide/thejbpmdatabase.html#d0e2677

Debugging OpenKM with Jboss

You can debug your OpenKM installation using the JBoss logging facility. This is an useful thing when you have problems with your configuration. Default OpenKM installation tries to log important events like errors and warnings. Is possible to change this configuration editing the file *$JBOSS_HOME/server/default/conf/jboss-log4j.xml*.

Default JBoss log configuration can generate a lot of messages. These files are stored at *$JBOSS_HOME/server/default/log*. It is configured to use the DailyRollingFileAppender. This appender create a new log file for every day. This is better than have a unique huge log file, os course. The rollover is performed at midnight each day, but you can configure it to make the rollover every hour (uncomment the proper line).

```
<!-- A time/date based rolling appender -->
<appender name="FILE"
          class="org.jboss.logging.appender.DailyRollingFileAppender">
   <errorHandler class="org.jboss.logging.util.OnlyOnceErrorHandler"/>
   <param name="File" value="${jboss.server.log.dir}/server.log"/>
   <param name="Append" value="false"/>
   <!-- Rollover at midnight each day -->
   <param name="DatePattern" value="'.'yyyy-MM-dd"/>
   <!-- Rollover at the top of each hour
   <param name="DatePattern" value="'.'yyyy-MM-dd-HH"/>
   -->
   <layout class="org.apache.log4j.PatternLayout">
      <!-- The default pattern: Date Priority [Category] Message\n -->
      <param name="ConversionPattern" value="%d %-5p [%c] %m%n"/>
      <!-- The full pattern: Date MS Priority [Category] (Thread:NDC) Message\n
      <param name="ConversionPattern" value="%d %-5r %-5p [%c] (%t:%x) %m%n"/>
      -->
   </layout>
</appender>
```

You can reduce the amount of log messages produced by OpenKM, or can increase them. In this example we limit the log messages produced by the class OKMAccessManager for those of type ERROR.

```
<category name="com.openkm.core.OKMAccessManager">
   <priority value="ERROR" />
</category>
```

If you create this configuration:

```
<category name="com.openkm">
   <priority value="DEBUG" />
</category>
```

All the log messages generated by OpenKM will be shown. As you can see, you can increase debug messages in some parts of OpenKM to check a determinate behavior.

Debugging JAAS configuration

If you are trying to setup another authentication source different from the default provided by OpenKM, you can afford some problems. The JBoss **login-config.xml** is supposed to have the right configuration, but you can't log into the application. The most common case is a bad or missing JAAS configuration. So if you need to debug the JAAS, you can add to the **$JBOSS_HOME/server/default/conf/jboss-log4j.xml** file the following:

```
<category name="org.jboss.security">
   <priority value="TRACE" class="org.jboss.logging.XLevel"/>
   <appender-ref ref="SECURITY_F"/>
</category>
<appender name="SECURITY_F" class='org.jboss.logging.appender.DailyRollingFileAppender'>
   <param name="Append" value="true"/>
   <param name="DatePattern" value="'.'yyyy-MM-dd"/>
   <param name="File" value="${jboss.server.home.dir}/log/jboss.security.log"/>
   <layout class="org.apache.log4j.PatternLayout">
      <param name="ConversionPattern" value="%d{ABSOLUTE} %-5p [%c] %m%n"/>
   </layout>
</appender>
```

This is more or less in the middle of the file, just where the <category-name> section begins. And you should look at this new log file:

```
$ tailf -f $JBOSS_HOME/server/default/log/jboss.security.log
```

Email error notification

Always is good idea to be notified when things goes wrong. There are some log appenders that can help you. The SMTPAppender will mail you log messages with threshold ERROR by default. You can lower this threshold, but you will got lots of useless mail messages. Here you must configure some properties:

- **To**: The mail account where the messages will arrive.
- **From**: You can set it simply as noreply@your-domain.com.
- **Subject**: Here you can specify the subject of the mail. If you have several OpenKM installations, you can create a filter in your mail client using this value.
- **SMTPHost**: The mail server server. Can be localhost if there is a mail server installed in this computer.

```
<!-- EMail events to an administrator -->
<appender name="SMTP" class="org.apache.log4j.net.SMTPAppender">
  <errorHandler class="org.jboss.logging.util.OnlyOnceErrorHandler"/>
  <param name="Threshold" value="ERROR"/>
  <param name="To" value="admin@myhost.domain.com"/>
  <param name="From" value="nobody@myhost.domain.com"/>
  <param name="Subject" value="JBoss Sever Errors"/>
  <param name="SMTPHost" value="localhost"/>
  <param name="BufferSize" value="10"/>
  <layout class="org.apache.log4j.PatternLayout">
     <param name="ConversionPattern" value="[%d{ABSOLUTE},%c{1}] %m%n"/>
  </layout>
</appender>
```

In Unix / Linux systems there is a centralized log manager called syslog. You can configure Log4J to use this system using the SyslogAppender:

```
<!-- Syslog events -->
<appender name="SYSLOG" class="org.apache.log4j.net.SyslogAppender">
  <errorHandler class="org.jboss.logging.util.OnlyOnceErrorHandler"/>
  <param name="Facility" value="LOCAL7"/>
  <param name="FacilityPrinting" value="true"/>
  <param name="SyslogHost" value="localhost"/>
  <layout class="org.apache.log4j.PatternLayout">
     <param name="ConversionPattern" value="[%d{ABSOLUTE},%c{1}] %m%n"/>
  </layout>
</appender>
```

- **SyslogHost**: This configuration parameters allows you to specify the local syslog or a remote syslog server used to centralize the network log management.

For more info, visit:

- http://jboss.org/community/docs/DOC-11280
- http://jboss.org/community/docs/DOC-9372
- http://jakarta.apache.org/log4j
- http://primalcortex.wordpress.com/2007/11/28/jboss-and-jaas-debug
- can I get log4j to delete old rotating log files? [1]

References

[1] http://stackoverflow.com/questions/1050256/how-can-i-get-log4j-to-delete-old-rotating-log-files|How

Article Sources and Contributors

Developer Guide *Source*: http://wiki.openkm.com/index.php?oldid=7578 *Contributors*: Anonymous, Jllort, Pavila

Configure Java in Eclipse *Source*: http://wiki.openkm.com/index.php?oldid=1027 *Contributors*: Pavila

Configure tomcat server in Eclipse *Source*: http://wiki.openkm.com/index.php?oldid=7512 *Contributors*: Jllort

Maven installation *Source*: http://wiki.openkm.com/index.php?oldid=2241 *Contributors*: Jllort, Pavila

Checkout source from Subversion *Source*: http://wiki.openkm.com/index.php?oldid=7466 *Contributors*: Anonymous, Jllort, Pavila

JBoss Developer Studio *Source*: http://wiki.openkm.com/index.php?oldid=6170 *Contributors*: Pavila

Development tips OpenKM 6.0 *Source*: http://wiki.openkm.com/index.php?oldid=7520 *Contributors*: Jllort

Development tips OpenKM 5.1 *Source*: http://wiki.openkm.com/index.php?oldid=5666 *Contributors*: Jllort

Development tips OpenKM 5.0 *Source*: http://wiki.openkm.com/index.php?oldid=5661 *Contributors*: Jllort, Pavila

Development tips OpenKM 4.0 *Source*: http://wiki.openkm.com/index.php?oldid=5667 *Contributors*: Jllort

Debug with tomcat *Source*: http://wiki.openkm.com/index.php?oldid=7580 *Contributors*: Jllort

Debugging with GWT *Source*: http://wiki.openkm.com/index.php?oldid=5673 *Contributors*: Jllort, Pavila, Wangmj

Profiling OpenKM *Source*: http://wiki.openkm.com/index.php?oldid=7387 *Contributors*: Pavila

Configure JBoss server in Eclipse *Source*: http://wiki.openkm.com/index.php?oldid=928 *Contributors*: Pavila

Debugging remote server *Source*: http://wiki.openkm.com/index.php?oldid=7099 *Contributors*: Jllort, Pavila

Browsing embeded database *Source*: http://wiki.openkm.com/index.php?oldid=2443 *Contributors*: Christoph.xmt, Jllort, Pavila

Debugging OpenKM with Jboss *Source*: http://wiki.openkm.com/index.php?oldid=7522 *Contributors*: Jllort, Pavila

Image Sources, Licenses and Contributors

File:Nota clasica.png *Source*: http://wiki.openkm.com/index.php?title=File:Nota_clasica.png *License*: unknown *Contributors*: Pavila

File:Nota idea.png *Source*: http://wiki.openkm.com/index.php?title=File:Nota_idea.png *License*: unknown *Contributors*: Pavila

File:Padlock.gif *Source*: http://wiki.openkm.com/index.php?title=File:Padlock.gif *License*: unknown *Contributors*: Pavila

File:Okm_dev_guide_004.jpeg *Source*: http://wiki.openkm.com/index.php?title=File:Okm_dev_guide_004.jpeg *License*: unknown *Contributors*: Jllort

File:Okm_dev_guide_005.jpeg *Source*: http://wiki.openkm.com/index.php?title=File:Okm_dev_guide_005.jpeg *License*: unknown *Contributors*: Jllort

File:Okm_dev_guide_035.png *Source*: http://wiki.openkm.com/index.php?title=File:Okm_dev_guide_035.png *License*: unknown *Contributors*: Jllort

File:Okm_dev_guide_036.png *Source*: http://wiki.openkm.com/index.php?title=File:Okm_dev_guide_036.png *License*: unknown *Contributors*: Jllort

File:Okm_dev_guide_037.png *Source*: http://wiki.openkm.com/index.php?title=File:Okm_dev_guide_037.png *License*: unknown *Contributors*: Jllort

File:Okm_dev_guide_038.png *Source*: http://wiki.openkm.com/index.php?title=File:Okm_dev_guide_038.png *License*: unknown *Contributors*: Jllort

File:Okm_dev_guide_039.png *Source*: http://wiki.openkm.com/index.php?title=File:Okm_dev_guide_039.png *License*: unknown *Contributors*: Jllort

File:Okm_dev_guide_040.png *Source*: http://wiki.openkm.com/index.php?title=File:Okm_dev_guide_040.png *License*: unknown *Contributors*: Jllort

File:Okm_dev_guide_001.jpeg *Source*: http://wiki.openkm.com/index.php?title=File:Okm_dev_guide_001.jpeg *License*: unknown *Contributors*: Jllort

File:Okm_dev_guide_002.jpeg *Source*: http://wiki.openkm.com/index.php?title=File:Okm_dev_guide_002.jpeg *License*: unknown *Contributors*: Jllort

File:Okm_dev_guide_003.jpeg *Source*: http://wiki.openkm.com/index.php?title=File:Okm_dev_guide_003.jpeg *License*: unknown *Contributors*: Jllort

File:Okm_dev_guide_006.jpeg *Source*: http://wiki.openkm.com/index.php?title=File:Okm_dev_guide_006.jpeg *License*: unknown *Contributors*: Jllort

File:JBoss Developer Subversion Import.png *Source*: http://wiki.openkm.com/index.php?title=File:JBoss_Developer_Subversion_Import.png *License*: unknown *Contributors*: Pavila

File:Okm_dev_guide_041.png *Source*: http://wiki.openkm.com/index.php?title=File:Okm_dev_guide_041.png *License*: unknown *Contributors*: Jllort

File:Okm_dev_guide_031.jpeg *Source*: http://wiki.openkm.com/index.php?title=File:Okm_dev_guide_031.jpeg *License*: unknown *Contributors*: Jllort

File:Okm_dev_guide_032.jpeg *Source*: http://wiki.openkm.com/index.php?title=File:Okm_dev_guide_032.jpeg *License*: unknown *Contributors*: Jllort

File:Okm_dev_guide_011.jpeg *Source*: http://wiki.openkm.com/index.php?title=File:Okm_dev_guide_011.jpeg *License*: unknown *Contributors*: Jllort

File:Okm_dev_guide_012.jpeg *Source*: http://wiki.openkm.com/index.php?title=File:Okm_dev_guide_012.jpeg *License*: unknown *Contributors*: Jllort

File:Okm_dev_guide_013.jpeg *Source*: http://wiki.openkm.com/index.php?title=File:Okm_dev_guide_013.jpeg *License*: unknown *Contributors*: Jllort

File:Okm_dev_guide_014.jpeg *Source*: http://wiki.openkm.com/index.php?title=File:Okm_dev_guide_014.jpeg *License*: unknown *Contributors*: Jllort

File:Okm_dev_guide_034.jpeg *Source*: http://wiki.openkm.com/index.php?title=File:Okm_dev_guide_034.jpeg *License*: unknown *Contributors*: Jllort

File:Okm_dev_guide_016.jpeg *Source*: http://wiki.openkm.com/index.php?title=File:Okm_dev_guide_016.jpeg *License*: unknown *Contributors*: Jllort

File:Okm_dev_guide_017.jpeg *Source*: http://wiki.openkm.com/index.php?title=File:Okm_dev_guide_017.jpeg *License*: unknown *Contributors*: Jllort

File:Okm_dev_guide_018.jpeg *Source*: http://wiki.openkm.com/index.php?title=File:Okm_dev_guide_018.jpeg *License*: unknown *Contributors*: Jllort

File:Okm_dev_guide_019.jpeg *Source*: http://wiki.openkm.com/index.php?title=File:Okm_dev_guide_019.jpeg *License*: unknown *Contributors*: Jllort

File:Okm_dev_guide_020.jpeg *Source*: http://wiki.openkm.com/index.php?title=File:Okm_dev_guide_020.jpeg *License*: unknown *Contributors*: Jllort

File:Okm_dev_guide_021.jpeg *Source*: http://wiki.openkm.com/index.php?title=File:Okm_dev_guide_021.jpeg *License*: unknown *Contributors*: Jllort

File:Visualvm proxy.png *Source*: http://wiki.openkm.com/index.php?title=File:Visualvm_proxy.png *License*: unknown *Contributors*: Pavila

File:Visualvm jmx.png *Source*: http://wiki.openkm.com/index.php?title=File:Visualvm_jmx.png *License*: unknown *Contributors*: Pavila

File:Okm_dev_guide_007.jpeg *Source*: http://wiki.openkm.com/index.php?title=File:Okm_dev_guide_007.jpeg *License*: unknown *Contributors*: Jllort

File:Okm_dev_guide_008.jpeg *Source*: http://wiki.openkm.com/index.php?title=File:Okm_dev_guide_008.jpeg *License*: unknown *Contributors*: Jllort

File:Okm_dev_guide_009.jpeg *Source*: http://wiki.openkm.com/index.php?title=File:Okm_dev_guide_009.jpeg *License*: unknown *Contributors*: Jllort

File:Okm_dev_guide_010.jpeg *Source*: http://wiki.openkm.com/index.php?title=File:Okm_dev_guide_010.jpeg *License*: unknown *Contributors*: Jllort

File:Okm_user_guide_073.jpeg *Source*: http://wiki.openkm.com/index.php?title=File:Okm_user_guide_073.jpeg *License*: unknown *Contributors*: Jllort

File:Okm_user_guide_074.jpeg *Source*: http://wiki.openkm.com/index.php?title=File:Okm_user_guide_074.jpeg *License*: unknown *Contributors*: Jllort

File:Okm_005.jpeg *Source*: http://wiki.openkm.com/index.php?title=File:Okm_005.jpeg *License*: unknown *Contributors*: Jllort