

SpagoBI eXo Tomcat Installation Manual

Authors Luca Fiscato
 Andrea Zoppello
 Davide Serbetto

Review Grazia Cazzin

Index

1	VERSION.....	3
2	DOCUMENT GOAL	3
3	REFERENCES	3
4	REQUIREMENTS	3
5	INSTALL SPAGOBI	3
5.1	INSTALL ADDITIONAL LIBRARIES	4
5.2	INSTALL THE METADATA DATABASE	4
5.2.1	<i>Install the database driver</i>	5
5.2.2	<i>Configuring Datasource as JNDI Resource</i>	5
5.3	INSTALL CMS REPOSITORY	5
5.4	INSTALL SPAGOBI PLATFORM.....	6
5.4.1	<i>Configure Hibernate</i>	6
5.5	INSTALL ENGINES	7
5.5.1	<i>Install SpagoBIJasperReportEngine.....</i>	7
5.5.2	<i>Install SpagoBIJPivotEngine</i>	7
5.6	INSTALL EXAMPLE PORTAL (OPTIONAL)	7

1 Version

Version/Release n° :	1.2	Data Version/Release :	Feb, 7th 2006
Update description:	eXo Tomcat 1.0 Installation steps		
Version/Release n° :	1.3	Data Version/Release :	May, 18th 2006
Update description:	Tomcat Server, with eXo Portal 1.1, installation steps		

2 Document goal

This document provides a step by step description for SpagoBI installation and configuration on a Tomcat Server 5.0.28 with eXo Portal 1.1 installed on. To do that, unzip the **eXoPortal-tomcat-Final.zip** downloaded from ObjectWeb forge (<http://forge.objectweb.org/projects/exoplatform>) to your file system.

3 References

Some concepts in this document refer to:

- SpagoBI business intelligence platform (available at <http://spagobi.eng.it/>)
- Exo Portal Platform (available at <http://www.exoplatform.com>)
- Spago framework (available at <http://spago.eng.it>)
- Tomcat server (available at <http://tomcat.apache.org/>)

4 Requirements

Using the eXo Portal v2, SpagoBI needs the JDK 1.5.x. You can verify that simply typing the *java -version* command from a command window.

We assume that you use the **java version 1.5.x** and that you have the eXo Portal 1.1 rightly installed on a Tomcat 5.0.28 server; in the following we'll refer to the exo-tomcat base directory as **EXO-HOME**.

5 Install SpagoBI

To test the eXo installation exec the command **EXO-HOME/bin/exo-run.bat** (in a windows environment) or **EXO-HOME/bin/exo-run.sh** (in a unix environment), wait until the command ends up with the message 'Server startup in xxxx ms' and then connect with a browser to the url <http://<<localhost>>:8080/portal>, the eXo home page should appear. Before proceed stop the server.

Connect to the SpagoBI page on the ObjectWeb community site (<http://forge.objectweb.org/projects/spagobi>), click on the 'File' tab, download the file called 'UtilityFiles-bin-1.8.zip', and then unzip it.

The unzip operation produces a folder 'SpagoBIUtilityFiles' which contains 'tomcat-server', 'libraries' and other folders. The folder 'tomcat-server' respects the tree-folders structure of a clean eXo-Tomcat installation; in the following we will refer to this folder as **EXO-INST-FILES**. The folder 'libraries' contains some necessary libraries; in the following we will refer to this folder as

LIBRARIES. There's also another folder 'example-portal' which contains a new portal war file useful to install a SpagoBI test portal (However the installation of this example portal is not mandatory); in the following we will refer to this folder as **EXAMPLE_PORTAL**

5.1 Install additional libraries

SpagoBI needs some additional libraries to manage metadata so copy the following libraries from **LIBRARIES** to **EXO-HOME/common/lib**:

- ehcache-1.1.jar
- concurrent-1.3.2.jar
- jackrabbit-0.16.4.1-dev.jar
- jcr-1.0.jar

5.2 Install the Metadata Database

SpagoBI metadata are stored in a database (for this release SpagoBI supports PostgreSQL, Oracle, and MySQL).

If you don't have anyone of these database servers supported you need to install one of them. Once you have a functional database server you must create a new database for the metadata (spagobi is the database name suggested).

Once completed the operation above it's possible to proceed with the creation and initial population of the metadata database launching the right script for your database server. For each database server supported you need to download from the SpagoBI Repository a zip archive containing the sql script to create the schema, the comments of the table and finally to populate the schema with initial data. (**These scripts don't contain the examples data as the hsqldb**). So, connect to <http://forge.objectweb.org/projects/spagobi>, select the file tab and then download the file called <<name of you database>>-dbscript-1.4.zip. Into the zip file there are some sql script files, you must execute the creation and insertion script with a client for your database server. In every archive exists also a drop script but this one is useful only if you need to clean your database deleting all the spagobi metadata tables. *Example: if you have a postgresql database server you need to download the relative archive (postgres-dbscript-1.4.zip) and run in the following order the scripts PG_create.sql and PG_insert.sql*

SpagoBI can use also an HSQLDB database, which is very useful for test purpose but is very weak in a production environment. Anyway, if you want to use HSQLDB, just copy the folder **EXO-INST-FILES/sbidata** into **EXO-HOME** and then exec the command `exo-home/sbidata/database/start.bat` (on a windows platform) or `exo-home/sbidata/database/start.sh` (on a unix platform). The command starts an HSQLDB server with a 'spagobi' databases already populated with the necessary data.

Every time you start the exo server for work with SpagoBI the database server must be running.

5.2.1 Install the database driver

Before proceed with persistence configuration we must install the database driver packages in the application server. Because SpagoBI can be configured to connect to different database servers, one for the metadata and one for datawarehouse for example, you must obtain the specific drivers for every database server used by SpagoBI. The drivers package can be obtained from database vendor sites and for the current SpagoBI release we test the following versions:

- Postgresql : postgresql-8.0-311.jdbc2.jar
- Oracle: ojdbc14.jar
- MySQL: mysql-connector-java-3.1.10-bin.jar
- HSQLDB: hsqldb1_8_0_2.jar (contained into **EXO-INST-FILES/common/lib**)

The driver jars, of the corresponding database you use, have to be put under **EXO-HOME/common/lib**.

Pay attention: If you decide to use the SpagoBI hsqldb database add into **EXO-HOME/common/lib** the hsqldb1_8_0_2.jar library and remember to remove the existing hsqldb1.8.0.1.jar library in the same folder in order to avoid conflicts.

5.2.2 Configuring Datasource as JNDI Resource

SpagoBI needs a connection to the metadata database. This connection can be configured as a direct jdbc connection or as link to a server jndi datasource. The second option is surely better and it's covered into this manual. To configure the JNDI global datasource resource do the following steps:

- Edit the file **EXO-INST-FILES/conf/server.xml**.
- Search for the string "<Resource name="jdbc/spagobi"");
- Copy entirely the <Resource> tag and the subsequent <ResourceParams> tag
- Edit the file **EXO-HOME/conf/server.xml**
- Inside tag named 'GlobalNamingResources' paste the piece of xml copied previously

The xml copied configures a new jndi datasource for the metadata database. Some of default settings are suitable for an hsql database and you need to change them if you are using another database server (the default values are for hsql db). Based on your database change the value of:

- DriverClassName
- url
- username
- password

5.3 Install Cms Repository

SpagoBI needs a connection to a content management system (cms) compliant to the jsr 170 specification in order to store and version the BI documents. The connection is represented by a jsr 170 'Repository' object which allows to open working session into the cms. SpagoBI can be configured to initialize directly the repository or to get it as a jndi resource. The second option is

surely better and it's covered into this manual. To configure the JNDI global repository resource do the following steps:

- o Edit the file **EXO-INST-FILES/conf/server.xml**.
- o Search for the string "<Resource name="cms/spagobicms"";
- o Copy entirely the <Resource> tag and the subsequent <ResourceParams> tag
- o Edit the file **EXO-HOME/conf/server.xml**
- o Inside tag named 'GlobalNamingResources' paste the piece of xml previously copied

The xml copied contains two parameters which have to be configured

- o ConfigFilePath (path of the jackrabbit configuration file)
- o RepHomeDir (path to the folder which will be the root of the repository)

You have simply to replace the string 'EXO-HOME' with the path of your exo tomcat installation directory. Example: if your exo tomcat is installed into a directoy C:\Programs\exo-tomcat you have to replace 'EXO-HOME' with '/Programs/exo-tomcat'

5.4 Install SpagoBI platform

Connect to <http://forge.objectweb.org/projects/spagobi>, select the file tab and then download the 'SpagoBI-bin-1.8.zip'. Extract from the zip archive the file **SpagoBI/spagobi.war** and copy it into **EXO-HOME/webapps** folder.

Copy the file **spagobi.xml** **EXO-INST-FILES/conf/Catalina/localhost** to **EXO-HOME/conf/Catalina/localhost**.

Start the server (with **exo-home/bin/exo-run.bat** or **.sh**) and after stop it (this operation will explode the **spagobi.war** just copied)

5.4.1 Configure Hibernate

SpagoBI uses Hibernate to manage metadata. Since SpagoBI can use different databases, hibernate must be configured in order to use the correct settings and sql dialect. Into the folder **exo-home/webapps/spagobi/WEB-INF/classes** there are four hibernate configuration files, one for each database supported:

- o **hibernate.cfg.xml** (postgres database)
- o **hibernate.cfg.ora.xml** (oracle database)
- o **hibernate.cfg.mysql.xml** (mysql database)
- o **hibernate.cfg.hsql.xml** (hsql database)

You must define which file must be used based on your database:

- o edit the file **exo-home/webapps/spagobi/WEB-INF/conf/spagobi/spagobi.xml**
- o search for the tag <HIBERNATE-CFGFILE>
- o change its value putting the name of the correct hibernate configuration file (the default is for hsqldb)

5.5 Install Engines

5.5.1 Install SpagoBIJasperReportEngine

- Connect to <http://forge.objectweb.org/projects/spagobi>, select the file tab and then download the file 'SpagoBIJasperReport-bin-1.8.zip'. Extract from the zip archive the file SpagoBI/SpagoBIJasperReportEngine.war and copy it into the EXO_HOME/webapps folder;
- copy the file SpagoBIJasperReportEngine.xml from EXO-INST-FILES/conf/Catalina/localhost into EXO-HOME/conf/Catalina/localhost.

5.5.2 Install SpagoBIJPivotEngine

- Connect to <http://forge.objectweb.org/projects/spagobi>, select the file tab and then download the file 'SpagoBIJPivotEngine-bin-1.8.zip'. Extract from the zip archive the file SpagoBI/SpagoBIJPivotEngine.war and copy it into the EXO-HOME/webapps folder.
- Copy the file SpagoBIJPivotEngine.xml from EXO-INST-FILES/conf/Catalina/localhost into EXO-HOME/conf/Catalina/localhost

5.6 Install Example Portal (Optional)

After the execution of the previous steps you should be able to connect to exo portal, imports SpagoBI portlets and define your own portal pages. (look at exo documentation for information and instructions). However our distribution contains also a simple portal, which can be installed over exo platform, useful to test SpagoBI portlets. To install the test portal:

- copy the EXAMPLE-PORTAL/sbiportal.war into EXO-HOME/webapps folder
- copy the EXAMPLE-PORTAL/sbiportal.xml into EXO-HOME/conf/Catalina/localhost folder
- start the server

At the end you should be able to connect to the url <http://localhost:8080/sbiportal>. The example portal defines four users:

- biadmin (password=biadmin): his pages contains the SpagoBI administration portlet
- bidev (password=bidev): his pages contains the SpagoBI development portlet
- bitest (password=bitest): his page contains the SpagoBI execution portlet with the possibility to view documents in test state
- biuser (password=biuser): his page contains the SpagoBI execution portlet

