

SpagoBI JBoss Installation Manual 1.4.3

Authors

Fiscato Luca
Zerbetto Davide

Index

1 VERSION.....	3
2 DOCUMENT GOAL.....	3
3 REFERENCES.....	3
4 INSTALLING SPAGOBI CORE.....	4
4.1 INSTALL ADDITIONAL LIBRARIES	4
4.2 INSTALLING THE METADATA DATABASE.....	4
4.2.1 Install the database driver.....	5
4.2.2 Configuring Datasource.....	5
4.3 INSTALL SPAGOBI PLATFORM.....	5
4.3.1 Configure Hibernate.....	6
4.3.2 Configure Quartz.....	7
4.4 CMS CONFIGURATION.....	7
4.5 ENGINES.....	8
4.5.1 Install SpagoBIJasperReportEngine.....	8
4.5.2 Install SpagoBIJPivotEngine.....	8
4.5.3 Install SpagoBIQbeEngine.....	8
4.5.4 Install SpagoBIBirtReportEngine.....	9
4.5.5 Install SpagoBIWekaEngine.....	9
4.5.6 Install SpagoBIGeoEngine.....	10
4.5.7 Install SpagoBIJPXMLAEngine.....	10
4.6 INSTALL EXAMPLE PORTAL (OPTIONAL).....	11

1 Version

Version/Release n° :	1.2	Data Version/Release :	Feb, 3 rd 2006
Update description:	JBoss Installation steps		
Version/Release n° :	1.3	Data Version/Release :	April, 11 th 2006
Update description:	JBoss 4.0.3SP1 server, with eXo Platform Portal 1.1, installation steps		
Version/Release n° :	1.4	Data Version/Release :	July, 14 th 2006
Update description:	JBoss 4.0.3SP1 server, with eXo Platform Portal 1.1.2, installation steps		
Version/Release n° :	1.4.1	Data Version/Release :	October, 9 th 2006
Update description:	JBoss 4.0.3SP1 server, with eXo Platform Portal 1.1.2, installation steps		
Version/Release n° :	1.4.2	Data Version/Release :	January, 23 th 2007
Update description:	JBoss 4.0.3SP1 server, with eXo Platform Portal 1.1.2, installation steps		
Version/Release n° :	1.4.3	Data Version/Release :	February, 15 th 2007
Update description:	Added SpagoBIJPXMLAEngine Installation		

2 Document goal

This document provides a step by step description for SpagoBI installation and configuration on a JBoss 4.0.3SP1 server with eXo Platform Portal 1.1.2 installed. To get a working copy of a JBoss 4.0.3SP1 server with eXo Platform Portal 1.1.2 you can:

- download, from the JBoss site, the version JBoss 4.0.3SP1 and unzip it into your file system (jboss-home)
- download, from eXo ObjectWeb site (<http://forge.objectweb.org/projects/exoplatform>), the file eXoPortal-jboss-1.1.2.zip and unzip it into jboss-home/server/default/deploy
- remove all the libraries contained into jboss-home/lib/endorsed

3 References

Some of the concepts of this document refer to the following documentation:

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

4 Installing SpagoBI Core

We assume that you have correctly installed eXo Platform Portal 1.1.2 on a JBoss 4.0.3SP1 server; in the following we will refer to the exo-JBoss base directory as **EXO-HOME**.

Connect to the SpagoBI's page on the ObjectWeb community site (<http://forge.objectweb.org/projects/spagobi>), click on the 'File' tab, download the file called 'UtilityFiles-bin-1.9.2.zip', and then unzip it.

The unzip operation produces a folder 'SpagoBIUtilityFiles' which contains 'jboss-server', 'tomcat-server' and other folders. The folder 'jboss-server' respects the tree-folders structure of a clean eXo-JBoss installation; in the following we will refer to this folder as **EXO-INST-FILES**. There's also another folder 'example-portal' which contains some files useful to install a new SpagoBI test portal (the installation of this example portal is not mandatory); in the following we will refer to this folder as **EXAMPLE_PORTAL**.

4.1 Install additional libraries

SpagoBI needs some additional libraries to manage metadata, so, copy the following libraries from **EXO-INST-FILES**/server/default/deploy/exoplatform.sar to **EXO-HOME**/server/default/deploy/exoplatform.sar:

- ehcache-1.1.jar
- concurrent-1.3.4.jar
- jackrabbit-core-1.0.1.jar
- slf4j-log4j12.jar
- jcr-1.0.jar

4.2 Installing the Metadata Database

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

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-1.9.2.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-1.9.2.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 (listening on port 9002) with a 'spagobi' database already populated with the necessary data.

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

4.2.1 Install the database driver

Before to proceed with persistence configuration we must install the database drivers 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 vendors site 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/server/default/lib)

The driver jar of the corresponding database you use have to be put under **EXO-HOME/server/default/lib**.

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

4.2.2 Configuring Datasource

SpagoBI needs a JNDI datasource for the metadata database. To configure the JNDI resources do the following steps:

- Copy from EXO-INST-FILES/server/default/deploy/ the file spagobi-ds.xml into EXO-HOME/server/default/deploy/.
- Edit the spagobi-ds.xml file and change the properties connection-url, driver-class, username, password, type-mapping with the right value for the connection to your SpagoBI database. (The default value are suitable for hsqldb).

4.3 Install SpagoBI platform

- Connect to <http://forge.objectweb.org/projects/spagobi>, select the file tab and then download the 'SpagoBI-bin-1.9.2.zip'. Extract from the zip archive the file spagobi.war and unzip it into EXO-HOME/server/default/deploy/exoplatform.sar/spagobi.war folder. (To unzip a war

file you need first to rename its extension from war to zip). At the end you should have a 'spagobi.war' folder (into exoplatform.sar directory) which contains other subfolders 'jsp', 'img', 'WEB-INF', etc.

- Edit the file application.xml in EXO-HOME/server/default/deploy/exoplatform.sar/META-INF folder and add the following module (add only the <module> xml envelope into the existing <application> xml envelope):

```
<application>
```

```
...
```

```
<module>
  <web>
    <web-uri>spagobi.war</web-uri>
    <context-root>spagobi</context-root>
  </web>
</module>
```

```
...
```

```
</application>
```

- Delete the jar commons-logging-1.0.jar from EXO-HOME/server/default/deploy/exoplatform.sar/spagobi.war/WEB-INF/lib.
- Edit the file spagobi.xml contained into EXO-HOME/server/default/deploy/exoplatform.sar/spagobi.war/WEB-INF/conf/spagobi and:
 - Search the tag <SPAGOBI_CONTEXT_PATH> and replace the value with your SpagoBI installation URL (you have simply to change the server name and port). Example: if you have installed SpagoBI on a server call myhost and the server is listening on port 5000 the value must be <http://myhost:5000/spagobi>.
 - Search the tag <LANGUAGE_SUPPORTED> and set to true the 'default' attribute of the children LANGUAGE tag, corresponding to your language. Be sure that one and only one of the LANGUAGE tags have the 'default' attribute set to true.

4.3.1 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/server/default/deploy/exoplatform.sar/spagobi.war/WEB-INF/classes there are four hibernate configuration files, one for each database supported:

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

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

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

4.3.2 Configure Quartz

The SpagoBI scheduler feature is based on Quartz technology. Quartz is a library which can store its metadata into a database. For the SpagoBI installation the quartz metadata database can be the same as the SpagoBI one. In order to configure it do the following steps:

- edit the file EXO-HOME/server/default/deploy/exoplatform.sar/spagobi.war/WEB-INF/classes/quartz.properties
- search the string 'job store delegate class'. Under this split line there's the same property repeated four times (org.quartz.jobStore.driverDelegateClass) . Based on your SpagoBI database you have to uncomment the right one and obviously to comment the others (to comment a row just place a # at the beginning)
- search the string org.quartz.dataSource.quartz.jndiURL and comment it (type a # as the first character of the row)
- search the string org.quartz.dataSource.quartz.driver and change its value putting the driver class name for your SpagoBI metadata database
- search the string org.quartz.dataSource.quartz.URL and change its value putting the url of your SpagoBI metadata database
- search the string org.quartz.dataSource.quartz.user and org.quartz.dataSource.quartz.password and change their values putting the username and password for the connection to your SpagoBI metadata database

4.4 CMS configuration

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. Using JBoss server it's necessary to initialize the repository directly, so:

- Only if you haven't already done it, copy the folder EXO-INST-FILES/sbidata into EXO-HOME folder.
- open the file cms-jboss-jonas.xml in EXO-HOME/server/default/deploy/exoplatform.sar/spagobi.war/WEB-INF/conf and substitute \${SERVER_HOME} with the actual path of EXO-HOME in the definition of the parameters 'repository_path' (this property indicates the folder in which JackRabbit will store contents) and 'conf_file_path' (this property indicates the file for Jackrabbit configuration). Pay attention at the path form, also for Windows system it must contain only / separator and it must start with / (no c:\ for example); An example path could be '/Programs/exo-jboss'
- delete the file cms.xml (contained into EXO-HOME/server/default/deploy/exoplatform.sar/spagobi.war/WEB-INF/conf folder) or rename it into 'cms_tomcat.xml'
- rename the file cms-jboss-jonas.xml (contained into EXO-HOME/server/default/deploy/exoplatform.sar/spagobi.war/WEB-INF/conf folder) into cms.xml
- At last you have to substitute the file commons-collections.jar in EXO-HOME/server/default/lib with the commons-collections-3.1.jar you find in EXO-HOME/server/default/deploy/exoplatform.sar/spagobi.war/WEB-INF/lib

4.5 Engines

4.5.1 Install SpagoBIJasperReportEngine

- Connect to <http://forge.objectweb.org/projects/spagobi>, select the file tab and then download the file 'SpagoBIJasperReportEngine-bin-1.9.2.zip'. Extract from the zip archive the file SpagoBIJasperReportEngine.war and rename it as SpagoBIJasperReportEngine.zip. Create a folder with the name SpagoBIJasperReportEngine.war and unzip the file SpagoBIJasperReportEngine.zip inside it. Delete the SpagoBIJasperReportEngine.zip file and copy the parent SpagoBIJasperReportEngine.war directory into EXO-HOME/server/default/deploy folder.
- Remove the following jar files from EXO-HOME/server/default/deploy/SpagoBIJasperReportEngine.war/WEB-INF/lib
 - log4j-1.2.8.jar
- Using the SpagoBI administration portlet define a new External Engine with:
 - document type: Report
 - engine type: External
 - driver class: it.eng.spagobi.drivers.jasperreport.JasperReportDriver
 - url: `http://<server>:<port>/SpagoBIJasperReportEngine/JasperReportServlet`.

4.5.2 Install SpagoBIJPivotEngine

- Connect to <http://forge.objectweb.org/projects/spagobi>, select the file tab and then download the file 'SpagoBIJPivotEngine-bin-1.9.2.zip'. Extract from the zip archive the file SpagoBIJPivotEngine.war and rename it as SpagoBIJPivotEngine.zip. Create a new folder named SpagoBIJPivotEngine.war and unzip the file SpagoBIJPivotEngine.zip inside it. Delete the SpagoBIJPivotEngine.zip file and copy the parent SpagoBIJPivot.war directory into EXO-HOME/server/default/deploy folder.
- Remove the following jar files from EXO-HOME/server/default/deploy/SpagoBIJPivotEngine.war/WEB-INF
 - log4j-1.2.8.jar
- Using the SpagoBI administration portlet define a new External Engine with:
 - document type: On-line analytical processing
 - engine type: External
 - driver class: it.eng.spagobi.drivers.jpivot.JPivotDriver
 - url: `http://<server>:<port>/SpagoBIJPivotEngine/jpivotOlap.jsp`.

4.5.3 Install SpagoBIQbeEngine

- Connect to <http://forge.objectweb.org/projects/spagobi>, select the file tab and then download the file 'SpagoBIQbeEngine-bin-1.9.2.zip'. Extract from the zip archive the file

SpagoBIQbeEngine.war. Unzip the war file into a directory named SpagoBIQbeEngine.war. Copy the new directory into EXO-HOME/server/default/deploy folder.

- If you want (advised) change the password of the administrator user (look at the section 'How to configure SpagoBIQbeEngine administrator users' of the How To documentation file to learn how to do it)
- Using the SpagoBI administration portlet define a new External Engine with:
 - document type: Datamart Model
 - engine type: External
 - driver class: it.eng.spagobi.drivers.qbe.QbeDriver
 - url: `http://<server>:<port>/SpagoBIQbeEngine/servlet/AdapterHTTP?ACTION_NAME=SPAGO_BI_START_ACTION&NEW_SESSION=TRUE` (without any space)

4.5.4 Install SpagoBIBirtReportEngine

- Connect to <http://forge.objectweb.org/projects/spagobi>, select the file tab and then download the file 'SpagoBIBirtReportEngine-bin-1.9.2.zip'. Extract from the zip archive the file SpagoBIBirtReportEngine.war and rename it as SpagoBIBirtReportEngine.zip. Create a new folder named SpagoBIBirtReportEngine.war and unzip the file SpagoBIBirtReportEngine.zip inside it. Delete the SpagoBIBirtReportEngine.zip file and copy the parent SpagoBIBirtReportEngine.war directory into EXO-HOME/server/default/deploy folder.
- Remove the following jar files from EXO-HOME/server/default/deploy/SpagoBIBirtReportEngine.war/WEB-INF/lib
 - log4j-1.2.8.jar
- Using the SpagoBI administration portlet define a new External Engine with:
 - document type: Report
 - engine type: External
 - driver class: it.eng.spagobi.drivers.birt.BirtReportDriver
 - url: `http://<server>:<port>/SpagoBIBirtReportEngine/BirtReportServlet`.

4.5.5 Install SpagoBIWekaEngine

- Connect to <http://forge.objectweb.org/projects/spagobi>, select the file tab and then download the file 'SpagoBIWekaEngine-bin-1.9.2-Alpha.zip'. Extract from the zip archive the file SpagoBIWekaEngine.war and rename it as SpagoBIWekaEngine.zip. Create a new folder named SpagoBIWekaEngine.war and unzip the file SpagoBIWekaEngine.zip inside it. Delete the SpagoBIWekaEngine.zip file and copy the parent SpagoBIWekaEngine.war directory into EXO-HOME/server/default/deploy folder.
- Remove the following jar files from EXO-HOME/server/default/deploy/SpagoBIWekaEngine.war/WEB-INF/lib
 - log4j-1.2.8.jar
- Using the SpagoBI administration portlet define a new External Engine with:
 - document type: Data mining model
 - engine type: External

o driver class: it.eng.spagobi.drivers.weka.WekaDriver
o url: http://<server>:<port>/SpagoBIWekaEngine/WekaServlet.

4.5.6 Install SpagoBIGeoEngine

- Connect to <http://forge.objectweb.org/projects/spagobi>, select the file tab and then download the file 'SpagoBIGeoEngine-bin-1.9.2-Alpha.zip'. Extract from the zip archive the file SpagoBIGeoEngine.war and rename it as SpagoBIGeoEngine.zip. Create a new folder named SpagoBIGeoEngine.war and unzip the file SpagoBIGeoEngine.zip inside it. Delete the SpagoBIGeoEngine.zip file and copy the parent SpagoBIGeoEngine.war directory into EXO-HOME/server/default/deploy folder.
- Remove the following jar files from EXO-HOME/server/default/deploy/SpagoBIGeoEngine.war/WEB-INF/lib
 - o xalan-2.4.0.jar
 - o xercesImpl.jar
 - o log4j-1.2.8.jar
- Using the SpagoBI administration portlet define a new External Engine with:
 - o document type: Map
 - o engine type: External
 - o driver class: it.eng.spagobi.drivers.geo.GeoDriver
 - o url: http://<server>:<port>/SpagoBIGeoEngine/servlet/AdapterHTTP?ACTION_NAME=GEO_ACTION&NEW_SESSION=TRUE.

4.5.7 Install SpagoBIJPXMLAEngine

- Connect to <http://forge.objectweb.org/projects/spagobi>, select the file tab and then download the file 'SpagoBIJPXMLAEngine-bin-1.9.2.zip'. Extract from the zip archive the file SpagoBIJPXMLAEngine.war and rename it as SpagoBIJPXMLAEngine.zip. Create a new folder named SpagoBIJPXMLAEngine.war and unzip the file SpagoBIJPXMLAEngine.zip inside it. Delete the SpagoBIJPXMLAEngine.zip file and copy the parent SpagoBIJPXMLAEngine.war directory into EXO-HOME/server/default/deploy folder.
- Remove the following jar files from EXO-HOME/server/default/deploy/SpagoBIJPXMLAEngine.war/WEB-INF
 - o log4j-1.2.8.jar
- Connect to <http://forge.objectweb.org/projects/spagobi>, select the file tab and then download the file 'SpagoBIJPXMLADriver-bin-1.9.2.zip'. Extract from the zip archive the file sbi.drivers.jpxmla-1.9.2.jar and copy it inside EXO-HOME/server/default/deploy/exoplatform.sar/spagobi.war/WEB-INF/lib
- Using the SpagoBI administration portlet define a new External Engine with:
 - o document type: On-line analytical processing
 - o engine type: External
 - o driver class: it.eng.spagobi.drivers.jpxmla.JPXMLADriver
 - o url: http://<server>:<port>/SpagoBIJPXMLAEngine/JPIVotServlet

4.6 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/jboss/sbiportal.war into EXO-HOME/server/default/deploy/exoplatform.sar folder;
- copy the file 'sbiportal.script' and 'sbiportal.properties' from EXAMPLE-PORTAL/jboss/db to EXO-HOME/temp/data (if the folder does not exist, create it);
- start the server.

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

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