openCRX Installation Guide for JBoss 4

Version 2.1

www.opencrx.org

Table of Contents

List of Figures

List of Listings

1   About this Book

openCRX is the leading open source CRM enterprise suite. openCRX is based on the openMDX application framework, an open source application framework based on the OMG's model driven architecture (MDA) standards, delivering maximum openness, standards compliance and a state-of-the-art component-based architecture.

This book describes the installation of openCRX for the JBoss application server.

1.1   Who this book is for

The intended audience are openCRX administrators and application server system administrators.

1.2   What do you need to understand this book

This book describes the installation of openCRX for JBoss. The book assumes that you are familiar with JBoss deployment concepts and administration.

1.3   Tips, Warnings, etc.

We make use the following pictograms:

Information provided as a "Tip" might be helpful for various reasons: time savings, risk reduction, etc.

You should carefully read information marked with "Important". Ignoring such information is typically not a good idea.

Warnings should not be ignored (risk of data loss, etc.)

2 Prerequisites

As a first step select the openCRX version you want to install. Based on the published version compatibility information you can determine the appropriate versions of openMDX, JBoss, and Java JDK:

http://www.opencrx.org/faq.htm#versioncompatibility

Write down the version numbers of the software packages you have chosen to install – this may be helpful in the future in case you require support or want to file a bug report:

openCRX v_____
openMDX v_____
JBoss   v_____
JDK   v_____

2.1 JDK 5.0

Install either the Sun Java JDK 5.0 or the BEA JRockit JDK 5.0 appropriate for your platform. The JDKs are available from the following sites:

Sun JDK 5.0:
http://java.sun.com/javase/downloads/index_jdk5.jsp

Oracle/BEA JRockit JDK 5.0:
http://www.oracle.com/technology/software/products/jrockit/index.html

Unless you are a Sun Java VM expert **we recommend Oracle/BEA JRockit for production use**. The reason is the following one:

If you deploy openCRX on Sun's JVM you might run into a problem that is mind-boggling (but specific to the Sun JVM). It's the dreaded

PermGen OutOfMemoryError

Lots of people have been struggling with this error (which is by the way not related to openCRX – any application could trigger it) and as of today we have not come across a solution other than to avoid Sun's JVM (JVMs by Oracle/BEA and IBM are – by construction – not prone to this error). Do a Google search if you want to convince yourself or drop us a line if you have a solution. But be aware that increasing the memory allocated for the permanent generation is no solution – it will only postpone the problem:

- http://my.opera.com/karmazilla/blog/2007/03/13/good-riddance-permgen-outofmemoryerror
- http://my.opera.com/karmazilla/blog/2007/03/15/permgen-strikes-back
- http://crashingdaily.wordpress.com/2007/02/04/crashing-tomcat/

It is not sufficient to have a Java Runtime Environment (JRE) only. The full-blown JDK is required to run openCRX.

Don't forget to set the **environment variable** JAVA_HOME

2.2 JBoss

Download **JBoss** from http://www.jboss.org/jbossas/downloads/

2.3 openmdx-kernel.jar / slf4j-openmdx1.jar

Download and install **openCRX Server** – available from

http://www.opencrx.org/server.htm

Once installed, you will find **openmdx-kernel.jar** and **slf4j-openmdx1.jar** in the directory <Server_Install_Dir>**\apache-tomcat-6\lib**

2.4 openCRX EARs

Either you can build the EARs from scratch with the openCRX SDK, or you can get pre-built EARs as follows: Download and install **openCRX Server** – available from

http://www.opencrx.org/server.htm

Once installed, you will find the pre-built openCRX EARs in the directory <Server_Install_Dir>**\apache-tomcat-6\deployment-units**

openCRX/core EAR:
opencrx-core-CRX.ear

openCRX/groupware EAR:
opencrx-groupware-CRX.ear

openCRX/store EAR:
opencrx-store-CRX.ear

Only opencrx-core-CRX.ear is required to run openCRX; the other EARs are optional and only required if you want to activate the contained functionality. For an initial setup you should deploy opencrx-core-CRX.ear only!

2.5   Database

Please note that you **must set up the openCRX database** as described in the respective openCRX database installation guide **before you continue**. For example, if you want to install openCRX for MySQL you must first install MySQL and the matching openCRX database definitions.

Database installation guides for the supported database management systems are available from http://www.opencrx.org/documents.htm

You will also need a **datasource configuration file**. Sample data source configurations for various database management systems are contained in the openCRX SDK in the directory
<SDK_Install_Dir>**\opencrx-2.1.0\core\src\connector\jboss-3**

Please remember that you will also need a **JDBC driver** appropriate for your database management system (please refer to the relevant DB installation guide for additional information).

Once you have successfully installed the database you are ready to continue with the JBoss setup.

3   Installing openCRX for JBoss

In a first step, install JBoss by extracting the JBoss distribution to your program directory, e.g. d:\jboss-4.2.1.GA on Windows or /opt/jboss on Linux.

> On Windows, avoid paths that contain blanks (e.g. the famous C:\Program File\...) as there are all kinds of pitfalls down the road...

> Make sure that you add JAVA_HOME to your system environment variables as JBoss needs the JDK to compile JSPs, e.g. JAVA_HOME=D:\pgm\j2sdk1.5 on Windows or JAVA_HOME=/usr/java/j2sdk1.5 on Linux.

Next you must deploy openCRX to JBoss. You do this by copying several files to the JBoss server directory structure:

- Copy the files **openmdx-kernel.jar** and **slf4j-openmdx1.jar** to the directory d:\jboss-4.2.1.GA\server\default\lib on Windows or /opt/jboss/server/default/lib on Linux.

- Copy the appropriate database **JDBC driver** to the directory d:\jboss-4.2.1.GA\server\default\lib. The openCRX database installation manual describes how to download the drivers.

- Copy the file **opencrx-core-CRX.ear** to the JBoss deploy directory d:\jboss-4.2.1.GA\server\default\deploy on Windows or /opt/jboss/server/default/deploy on Linux.

    > **Do NOT deploy any other openCRX EARs (e.g. groupware) before having completed the openCRX QuickStart guide.**
    >
    > Optionally, you can also open **opencrx-core-CRX.ear** with a ZIP utility and extract the content to the directory d:\jboss-4.2.1.GA\server\default\deploy\opencrx-core-CRX-web.ear or /opt/jboss/server/default/deploy/opencrx-core-CRX-web-ear
    >
    > If you want to edit the content of the file **opencrx-core-CRX.war** without the zip/unzip roundtrip you can also extract the content with a ZIP utility.

- Install the **datasource configuration file** appropriate for your DB, e.g. copy the file **jdbc-opencrx-CRX-mysql-ds.xml** (if you use openCRX with MySQL) to the directory d:\jboss-4.2.1.GA\server\default\deploy on Windows or /opt/jboss/server/default/deploy on Linux.

    > Verify all the relevant information for correctness, e.g. ensure that the connection url – in particular the database name – the user name, and the password match with your installation.
    > A simple copy/paste of our sample files will typically not work, i.e. it is expected that you will have to adapt some of the parameters.

● Create the file **server.log.properties** with the following content in

d:\jboss-4.2.1.GA\server\default\on Windows or
/opt/jboss/server/default/ on Linux

Listing 1: The file server.log.properties

```
ApplicationId = openCRX
LogFileExtension = log
LogFilePath = D:/jboss-4.2.1.GA/server/default/log/
LogLevel = warning
java.LoggingMechanism = SharedDatedFileLoggingMechanism
```

Adapt D:/jboss-4.2.1.GA to your environment! If the directory log does not exist, you should create it now.

Next you must set a few Java VM options which are required for the openMDX application framework.

On Windows add the following lines to d:\jboss-4.2.1.GA\bin**run.bat** after the lines indicated below. Also uncomment and adapt the line

set JAVA_OPTS=%JAVA_OPTS% -Xms128m -Xmx800m

This gives more memory to the Java VM (and depending on your environment you may want to increase the value of the option Xmx to more than 800m).

Listing 2: Java VM options required for openMDX on Windows

```
rem Sun JVM memory allocation pool parameters. Uncomment and modify as appropriate.
set JAVA_OPTS=%JAVA_OPTS% -Xms128m -Xmx800m

rem Setup openMDX-specific properties
set JAVA_OPTS=%JAVA_OPTS% -Dorg.openmdx.compatibility.base.application.j2ee.domain=apps
set JAVA_OPTS=%JAVA_OPTS% -Dorg.openmdx.compatibility.base.application.j2ee.server=server1
set JAVA_OPTS=%JAVA_OPTS% -Djava.protocol.handler.pkgs=org.openmdx.kernel.url.protocol
set JAVA_OPTS=%JAVA_OPTS% -Dorg.openmdx.log.config.filename=D:\jboss-4.2.1.GA\server\default\server.log.properties
```

Adapt D:\jboss-4.2.1.GA\server\default\server.log.properties to your environment and make sure that there are no line breaks in the set commands. Each -D options is of the form -Dname=value and must be on a single line.

On Linux add the following lines to /opt/jboss/bin/**run.conf** towards the end of the file.

Listing 3: Java VM options required for openMDX on Linux

```
rem Setup openMDX-specific properties
JAVA_OPTS="$JAVA_OPTS -Xms128m -Xmx800m"
JAVA_OPTS="$JAVA_OPTS -Dorg.openmdx.compatibility.base.application.j2ee.domain=apps"
JAVA_OPTS="$JAVA_OPTS -Dorg.openmdx.compatibility.base.application.j2ee.server=server1"
JAVA_OPTS="$JAVA_OPTS -Djava.protocol.handler.pkgs=org.openmdx.kernel.url.protocol"
JAVA_OPTS="$JAVA_OPTS -Dorg.openmdx.log.config.filename=/opt/jboss/server/default/server.log.properties"
```

Adapt /opt/jboss/server/default/server.log.properties to your environment and make sure that there are no line breaks in the set commands. Each -D options is of the form -Dname=value and must be on a single line.

4   Configuring Security

As a final step you must activate security for the openCRX application. You can configure the file-based *UsersRolesLoginModule* and/or the database-based *DatabaseServerLoginModule*.

4.1   Configuring LoginModule

openCRX stores security information in the database tables **OOCSE1_***, i.e.

OOCSE1_Authenticationcontext	**OOCSE1_Authenticationcontext_**
OOCSE1_Credential	**OOCSE1_Credential_**
OOCSE1_Permission	**OOCSE1_Permission_**
OOCSE1_Policy	**OOCSE1_Policy_**
OOCSE1_Principal	**OOCSE1_Principal_**
OOCSE1_Privilege	**OOCSE1_Privilege_**
OOCSE1_Realm	**OOCSE1_Realm_**
OOCSE1_Role	**OOCSE1_Role_**
OOCSE1_Segment	**OOCSE1_Segment_**
OOCSE1_Subject	**OOCSE1_Subject_**

JBoss can be enabled to access these tables by configuring a database login module. This way users can be managed in openCRX and are immediately available as JBoss logins.

It is strongly recommended that you stay with the file-based UsersRolesLoginModule for the user **admin-Root**. This simplifies the openCRX bootstrapping.

We recommend that you stay with the file-based authentication for all users until you have finished installing openCRX. You avoid situations where you have to trouble-shoot multiple issues at the same time...

Activate JAAS based authentication by adding the following configuration entries for the openCRX servlet to the JBoss configuration file

> d:\jboss-4.2.1.GA\server\default\conf**login_config.xml** (**login-config.xml** on Unix platforms)

Listing 4: JBoss configuration for JAAS based authentication

```
<application-policy name="opencrx-core-CRX">
  <authentication>
    <login-module code="org.jboss.security.auth.spi.DatabaseServerLoginModule" flag="sufficient">
      <module-option name="dsJndiName">java:/jdbc_opencrx_CRX</module-option>
      <module-option name="principalsQuery">SELECT c.passwd FROM OOCSE1_PRINCIPAL p, OOCSE1_CREDENTIAL c WHERE
(p.object_id LIKE 'principal/CRX/Root/Default/%') AND (p.credential = c.object_id) AND (p.name = ?)</module-
option>
      <module-option name="rolesQuery">SELECT r.name, 'Roles' FROM OOCSE1_PRINCIPAL_ pg, OOCSE1_PRINCIPAL p,
OOCSE1_PRINCIPAL_ pn, OOCSE1_ROLE r WHERE  (p.object_id = pn.object_id) AND (pn.is_member_of = pg.object_id) AND
(pg.granted_role = r.object_id) AND (p.object_id LIKE 'principal/CRX/Root/Default/%') AND (p.name = ?)</module-
option>
      <module-option name="ignorePasswordCase">true</module-option>
      <module-option name="hashCharset">UTF-8</module-option>
      <module-option name="hashEncoding">base64</module-option>
      <module-option name="hashAlgorithm">MD5</module-option>
    </login-module>
    <login-module code="org.jboss.security.auth.spi.UsersRolesLoginModule" flag="sufficient" >
      <module-option name="usersProperties">users.properties</module-option>
      <module-option name="rolesProperties">roles.properties</module-option>
    </login-module>
  </authentication>
</application-policy>
```

The settings above enable both file-based authentication and database-based authentication.

Next you need to create the files **users.properties** and **roles.properties** in the directory d:\jboss-4.2.1.GA\server\default\conf (Windows) or /opt/jboss/server/default/conf (Linux).

*Listing 5: File **users.properties** with syntax user=password*

```
admin-Root=rootSecret
admin-Standard=adminSecret
guest=guest
```

*Listing 6: File **roles.properties** with syntax user.Roles=role1,role2*

```
admin-Root.Roles=OpenCrxRoot
admin-Standard.Roles=OpenCrxAdministrator
guest.Roles=OpenCrxUser
```

It is strongly recommended that you stay with the file-based UsersRolesLoginModule for the user **admin-Root**. This simplifies the openCRX bootstrapping.

5   Starting JBoss

You are now ready to start JBoss. Open a command shell and start
d:\jboss-4.2.1.GA\bin\run.bat. You should verify whether the start options
match the ones described earlier:

Listing 7:   JBoss console output

```
===============================================================================

  JBoss Bootstrap Environment

  JBOSS_HOME: D:\jboss-4.2.1.GA

  JAVA: D:\Java\jrockit1.5.0\bin\java

  JAVA_OPTS:
-Djava.library.path="D:\jboss-4.2.1.GA\bin\native;"
-Dprogram.name=run.bat -Xms128m -Xmx824m
-Dorg.openmdx.compatibility.base.application.j2ee.domain=apps
-Dorg.openmdx.compatibility.base.application.j2ee.server=server1
-Djava.protocol.handler.pkgs=org.openmdx.kernel.url.protocol
-Dorg.openmdx.log.config.filename=D:\jboss-4.2.1.GA\server\default\server.log.properties
-Dorg.opencrx.maildir=D:\jboss-4.2.1.GA\server\default\maildir
-Dsun.rmi.dgc.client.gcInterval=3600000
-Dsun.rmi.dgc.server.gcInterval=3600000

  CLASSPATH: D:\Java\jrockit1.5.0\lib\tools.jar;D:\jboss-4.2.1.GA\bin\run.jar

===============================================================================
...
...
...
12:28:34,984 INFO  [ConnectionFactoryBindingService] Bound ConnectionManager
                   'jboss.jca:service=DataSourceBinding,name=jdbc_opencrx_CRX' to JNDI name 'java:jdbc_opencrx_CRX'
12:28:35,062 INFO  [ConnectionFactoryBindingService] Bound ConnectionManager
                   'jboss.jca:service=ConnectionFactoryBinding,name=JmsXA' to JNDI name 'java:JmsXA'
12:28:35,078 INFO  [TomcatDeployer] deploy, ctxPath=/jmx-console, warUrl=.../deploy/jmx-console.war/
12:28:35,203 INFO  [EARDeployer] Init J2EE application: file:/D:/jboss-4.2.1.GA/server/default/deploy/opencrx-core-CRX.ear/
12:28:48,016 INFO  [EjbModule] Deploying opencrx_core_CRX_EntityManagerFactory
12:28:48,078 INFO  [EjbModule] Deploying opencrx_core_CRX_Gateway
12:28:48,219 INFO  [EjbModule] Deploying opencrx_core_CRX_kernel_mandatory
12:28:48,250 INFO  [EjbModule] Deploying opencrx_core_CRX_security_mandatory
12:28:48,344 INFO  [BaseLocalProxyFactory] Bound EJB LocalHome
                   'opencrx_core_CRX_EntityManagerFactory' to jndi 'org.opencrx.core.CRX.EntityMangerFactory'
12:28:48,344 INFO  [BaseLocalProxyFactory] Bound EJB LocalHome 'opencrx_core_CRX_Gateway' to jndi 'local/opencrx_core_CRX_Gateway@37895256'
12:28:48,359 INFO  [ProxyFactory] Bound EJB Home 'opencrx_core_CRX_Gateway' to jndi 'org.opencrx.core.CRX.Gateway'
12:28:48,359 INFO  [EJBDeployer] Deployed: file:/D:/jboss-4.2.1.GA/server/default/deploy/opencrx-core-CRX.ear/gateway.jar
12:28:48,406 INFO  [BaseLocalProxyFactory] Bound EJB LocalHome 'opencrx_core_CRX_kernel_mandatory' to jndi
                   'org.opencrx.core.CRX.local.mandatory.kernel'
12:28:48,406 INFO  [EJBDeployer] Deployed: file:/D:/jboss-4.2.1.GA/server/default/deploy/opencrx-core-CRX.ear/kernel.jar
12:28:48,422 INFO  [BaseLocalProxyFactory] Bound EJB LocalHome
                    'opencrx_core_CRX_security_mandatory' to jndi 'org.opencrx.core.CRX.local.mandatory.security'
12:28:48,422 INFO  [EJBDeployer] Deployed: file:/D:/jboss-4.2.1.GA/server/default/deploy/opencrx-core-CRX.ear/security.jar
12:28:48,438 INFO  [TomcatDeployer] deploy, ctxPath=/opencrx-core-CRX, warUrl=.../deploy/opencrx-core-CRX.ear/opencrx-core-CRX.war/
12:28:50,109 INFO  [[/opencrx-core-CRX]] org.tuckey.web.filters.urlrewrite.utils.Log ERROR: logLevelConf: WARN
12:28:50,141 INFO  [[/opencrx-core-CRX]]  [CompressingFilter/1.6.4] CompressingFilter has initialized
12:28:50,250 INFO  [EARDeployer] Started J2EE application: file:/D:/jboss-4.2.1.GA/server/default/deploy/opencrx-core-CRX.ear/
12:28:50,312 INFO  [Http11AprProtocol] Starting Coyote HTTP/1.1 on http-0.0.0.0-8080
12:28:50,344 INFO  [AjpAprProtocol] Starting Coyote AJP/1.3 on ajp-0.0.0.0-8009
12:28:50,344 INFO  [Server] JBoss (MX MicroKernel) [4.2.1.GA (build: SVNTag=JBoss_4_2_1_GA date=200707131605)] Started in 29s:594ms
```

Now you are ready to continue with the openCRX QuickStart Guide or you can
Install openCRX as Windows Service.

6   Additional JBoss Settings

6.1   Automatically delete the Directories tmp and work

To ensure a clean and consistent startup environment it is recommended to delete the directories

d:\jboss-4.2.1.GA\server\default\tmp and
d:\jboss-4.2.1.GA\server\default\work and
d:\jboss-4.2.1.GA\server\default\maildir (on Windows)

or

/opt/jboss/server/default/tmp and
/opt/jboss/server/default/work and
/opt/jboss/server/default/maildir (on Linux).

before you (re)start JBoss.

Add some statements to the file **run.bat** (Windows) or **run.sh** (Linux) to automatically delete these directories upon starting JBoss.

On Windows, for example, you could add something along the lines of

RMDIR /S /Q D:\jboss-4.2.1.GA\server\default\tmp
RMDIR /S /Q D:\jboss-4.2.1.GA\server\default\work
RMDIR /S /Q D:\jboss-4.2.1.GA\server\default\maildir

to the beginning of **run.bat**

6.2   UTF-8 Support

For **full UTF-8 support** it is necessary to add/set the Tomcat option **URIEncoding="UTF-8"** in the file **server.xml** (details on the http connector reference page of the Apache-Jakarta-Project). Full UTF-8 support is for example required if you want to search for UTF-8 encoded characters.

6.3   Port Binding and Security

Newer versions of JBoss (4.2 and newer) bind to localhost only by default (it used to be global 0.0.0.0). You can get this same behavior by starting JBoss with **-b 0.0.0.0**.

If you look in the readme.html in your JBoss distribution there is a link with info on how to secure JBoss once you bind to 0.0.0.0. Alternatively, you can fiddle with the Tomcat options in the file **server.xml**.

7   openCRX as a Windows Service / Linux daemon

7.1   Install openCRX as a Windows Service

If you want to install JBoss / openCRX on a Windows platform as a Windows service you can do this by following instructions from the JBoss Wiki at http://wiki.jboss.org/wiki/RunJBossAsAServiceOnWindows:

- Download JBoss Web Server 2.0.1.GA from http://labs.jboss.com/jbossweb/downloads

- Unzip jboss-native-2.0.1xxx.zip to your JBoss Directory D:\jboss-4.2.1.GA\

- Open a DOS shell, navigate to D:\jboss-4.2.1.GA\bin and then execute the command

```
service.bat install
```

Adapt d:\jboss-4.2.1.GA to your environment!

7.2   Install openCRX as daemon on Linux

This section (provided by Seah Hong Yee) is devoted to the automatic start up of jboss services during the startup phase of a server. It also simplifies manual jboss startup with the use a of System V init script. The following configuration has been tested on Mandrake Linux 10.1 and SuSe Linux Enterprise Server 9. Based on the particular distribution at hand there might be some minor differences in init scripts and configuration, but the following guide should work with RHEL, CentOS, WhiteBox and Fedora.

In the directory $JBOSS_HOME/bin there should be two init scripts:

- jboss_init_redhat.sh
- jboss_init_suse.sh

If you are using **Mandrake/Mandriva**, **RHEL**, **CentOS**, **WhiteBox** or **Fedora**:

- Copy jboss_init_redhat.sh into /etc/init.d and rename it to jboss.

- Edit the script and adapt the parameters: JBOSS_HOME and JAVAPTH

- Although not strictly necessary, you might want to include an entry like:

Listing 8:   Declare location of jdk

```
export PATH=/usr/java/j2sdk1.5.0_06/bin
```

(/usr/java/j2sdk1.5.0_06/ being your jdk path, adapt it to your environment)

- At the top of the script there is an entry resembling the following one:

Listing 9:   Declare startup sequence

```
# chkconfig: 3 87 20
```

The second set of digits represents the order sequence of the service startup. Make sure the number is larger than your database startup. Typically postgresql starts with the sequence number of 85, so I have my jboss startup with the sequence of 87

- Type the commands

Listing 10:   Add JBoss to config

```
# chkconfig --add jboss
# chkconfig jboss on
```

From now on jboss should startup automatically after reboot, or you can do it manually with *service jboss restart*.

If you are using **Suse** Linux:

- Copy the jboss_init_suse.sh script into /etc/init.d and rename it to jboss.
- Edit the script and adapt the following parameters: JBOSS_HOME and JAVAPTH
- Although not strictly necessary, you might want to include an entry near the top of the script like:

Listing 11:   Declare location of jdk

```
export PATH=/usr/java/j2sdk1.5.0_06/bin
```

(/usr/java/j2sdk1.5.0_06/ being your jdk path, adapt it to your environment)

- Type the command "*inserv jboss*"
- Go into directory /usr/sbin and create a symbolic link with "*ln -s /etc/init.d/jboss rcjboss*"
- Execute the command "*chkconfig jboss on*"
- You should now be able to start jboss with the command "*rcjboss start*"

8 Next Steps

Now that you have successfully deployed openCRX on your application server you can continue with the openCRX QuickStart guide.