

openCRX Installation Guide for WebLogic 8

Version 1.5.0

www.opencrx.org

openCRX Installation Guide for WebLogic 8: Version 1.5.0

by www.opencrx.org

The contents of this file are subject to a BSD license (the "License"); you may not use this file except in compliance with the License.

You may obtain a copy of the License *here* (<http://www.opencrx.org/license.htm>)

Table of Contents

1. About this Book	1
Who this book is for	1
What do you need to understand this book	1
2. Prerequisites	2
3. Installing openCRX for WebLogic	3
4. Installing Libraries	4
5. Configuring the Java Virtual Machine	5
6. Configuring the Datasource	6
7. Configuring Security	13
8. Deploying openCRX	16
9. Next Steps	19
A. Appendix	20
Bibliography	21

List of Figures

3-1. Launch the WebLogic Server Console.	3
6-1. Select the JDBC database vendor.	6
6-2. Select the JDBC driver.	7
6-3. Configure the connection pool.	8
6-4. Configure the connection pool.	8
6-5. Create and deploy the connection pool.	9
6-6. Create and configure the datasource.	10
6-7. Select the connection pool.	11
6-8. Select the deployment targets for the datasource.....	12
7-1. Create user groups.	13
7-2. Create global security roles.	14
7-3. Add member of the group condition to global role.....	15
7-4. Create user and make member of group.	15
8-1. Upload the opencrx-core-CRX-App application enterprise archive.	16
8-2. Deploy the opencrx-core-CRX-App.ear enterprise archive.	17
8-3. Select the deployment target and assign name.	18
8-4. Verify the installation.	18

List of Examples

5-1. openCRX-specific Java VM properties	5
--	---

Chapter 1. About this Book

openCRX is the leading open source CRM tool. *openCRX* is based on the *openMDX [02]* application framework, an open source application framework based on the OMG's model driven architecture (MDA) standards. This guarantees maximum openness, standards compliance and a state-of-the-art component-based architecture.

Who this book is for

This book describes the installation of *openCRX* for *WebLogic 8.x*. The intended audience are *openCRX* and application server system administrators.

What do you need to understand this book

The book assumes that you are familiar with *WebLogic* deployment concepts and administration.

Chapter 2. Prerequisites

As a first step select the *openCRX* version you want to install. Based on the published *version compatibility information* (<http://www.opencrx.org/faq.htm#versioncompatibility>) you can determine the appropriate versions of *openMDX* and *WebLogic*.

- Purchase/Install **WebLogic 8.x** (please note that we have tested openCRX on WebLogic 8.1.x)
- Download **openMDX** from *here* (http://sourceforge.net/project/showfiles.php?group_id=75132).
- Download **openCRX** from *here* (http://sourceforge.net/project/showfiles.php?group_id=95219). You must download the *opencrx-core* distribution for *jre-1.4* (e.g. *opencrx-1.5.0-core.CRX.jre-1.4.zip*).

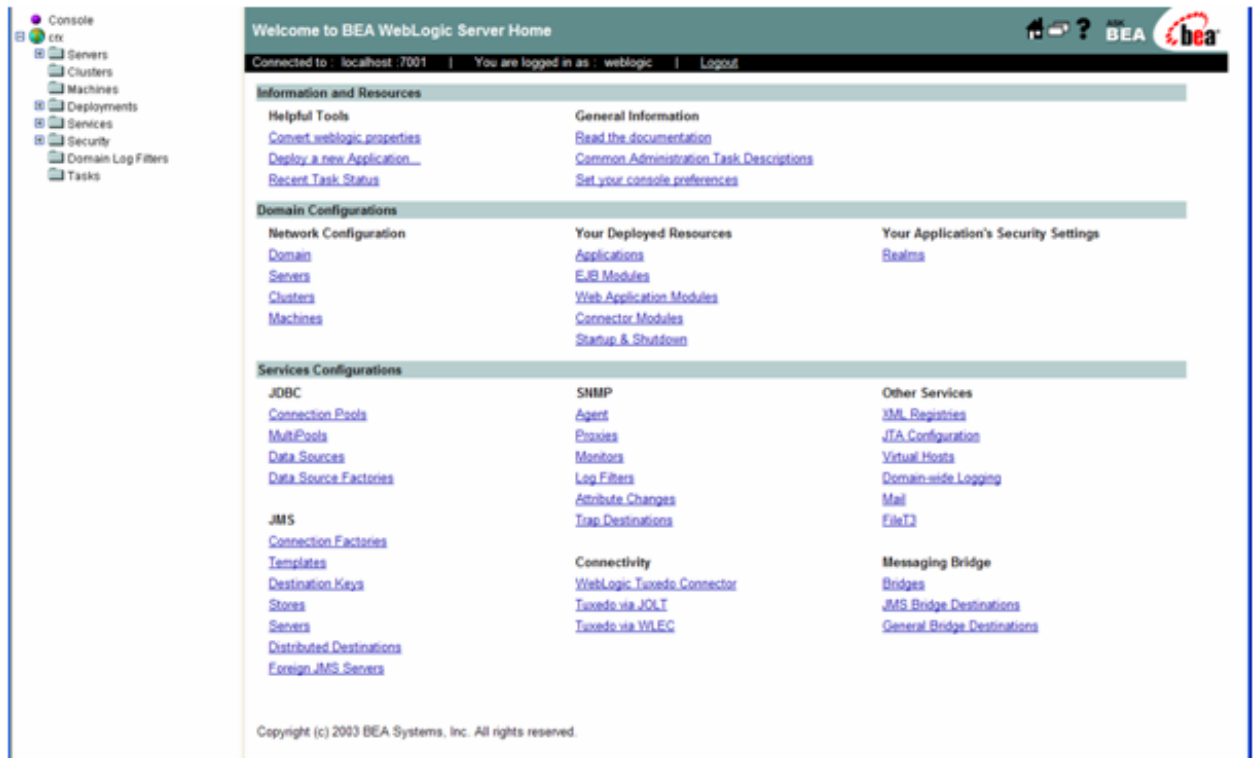


Important As a first step you must install the database as described in the database distribution. E.g. if you want to install *openCRX* for *MS SQL* you must first install *MS SQL* and the corresponding *openCRX* database schema. If you have successfully installed the database you are ready to continue with the *WebLogic* setup.

Chapter 3. Installing openCRX for WebLogic

In a first step you must install *WebLogic*. You should be able to start and stop *WebLogic* and launch the *WebLogic Server Console* as shown in *Figure 3-1*.

Figure 3-1. Launch the WebLogic Server Console.



The *openCRX* installation requires the following steps:

- Configure properties for the Java Virtual Machine required by *openCRX* and *openMDX*.
- Create and configure the datasource to access the *openCRX* database.
- Configure security, i.e. global roles, groups and principals.
- Deploy the *openCRX* application enterprise archives.
- Start and test *openCRX*

Chapter 4. Installing Libraries

As a first step you must copy the *openMDX* and Jdbc driver libraries to a place where *WebLogic* has access to it. You can do this as follows:

- Create a folder where you can store the libraries and *WebLogic* has access to it. E.g. create the directory `..\bea\user_projects\domains\mydomain\lib`.
- **openMDX**. Copy the library *openmdx-kernel.jar* to this newly created directory.
- **Database**. Copy the Jdbc libraries for your database to this newly created directory. E.g. for *Microsoft SQL Server 2000* the files are *msbase.jar*, *mssqlserver.jar*, *msutil.jar*

Chapter 5. Configuring the Java Virtual Machine

Second you must add some options to the Java Virtual Machine configuration. You can do this by adding the following settings to the *WebLogic* startup script, e.g. *startWebLogic.cmd* as follows:

Example 5-1. openCRX-specific Java VM properties

```
rem Setup openCRX-specific properties
set DOMAIN_HOME=c:\p9m\bea\user_projects\domains\mydomain
set CLASSPATH=%CLASSPATH%;%DOMAIN_HOME%\lib\openmdx-kernel.jar
set CLASSPATH=%CLASSPATH%;%DOMAIN_HOME%\lib\msutil.jar;%
set CLASSPATH=%CLASSPATH%;%DOMAIN_HOME%\lib\msbase.jar;
set CLASSPATH=%CLASSPATH%;%DOMAIN_HOME%\lib\mssqlserver.jar
set JAVA_OPTIONS=%JAVA_OPTIONS% -Dorg.openmdx.compatibility.base.application.j2ee.domain=apps
set JAVA_OPTIONS=%JAVA_OPTIONS% -Dorg.openmdx.compatibility.base.application.j2ee.server=server1
set JAVA_OPTIONS=%JAVA_OPTIONS% -Djava.protocol.handler.pkgs=org.openmdx.kernel.url.protocol
set JAVA_OPTIONS=%JAVA_OPTIONS% -Dorg.openmdx.log.config.filename=%DOMAIN_HOME%\server.log.properties
set JAVA_OPTS=%JAVA_OPTS% -Dmail.SSLSocketFactory.class=org.opencrx.kernel.mail.SendMailSSLSocketFactory

@REM Call WebLogic Server
...
```



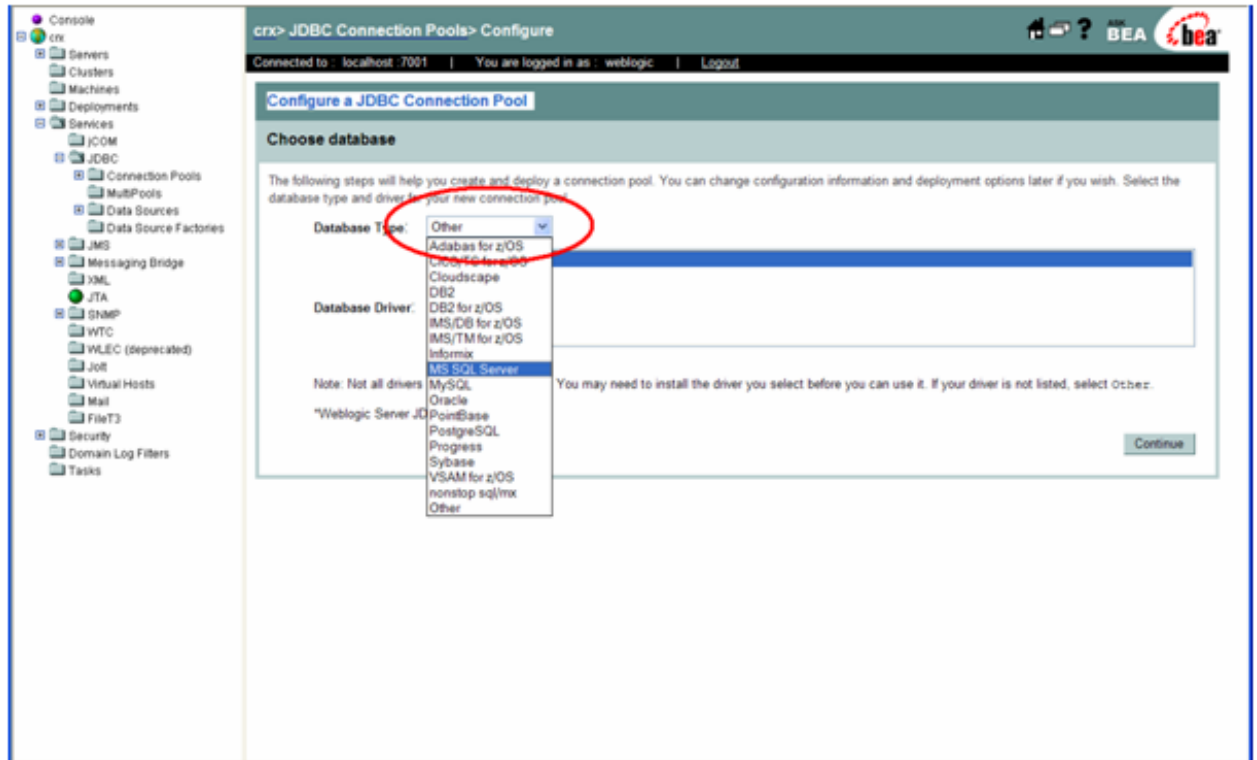
Important Before you continue you must restart (stop and start) *WebLogic*. The newly configured libraries and environment variables are not active until you restart *WebLogic*.

Chapter 6. Configuring the Datasource

The *openCRX* application requires the configuration of a JDBC datasource which connects to the *openCRX* database. You can do this as follows:

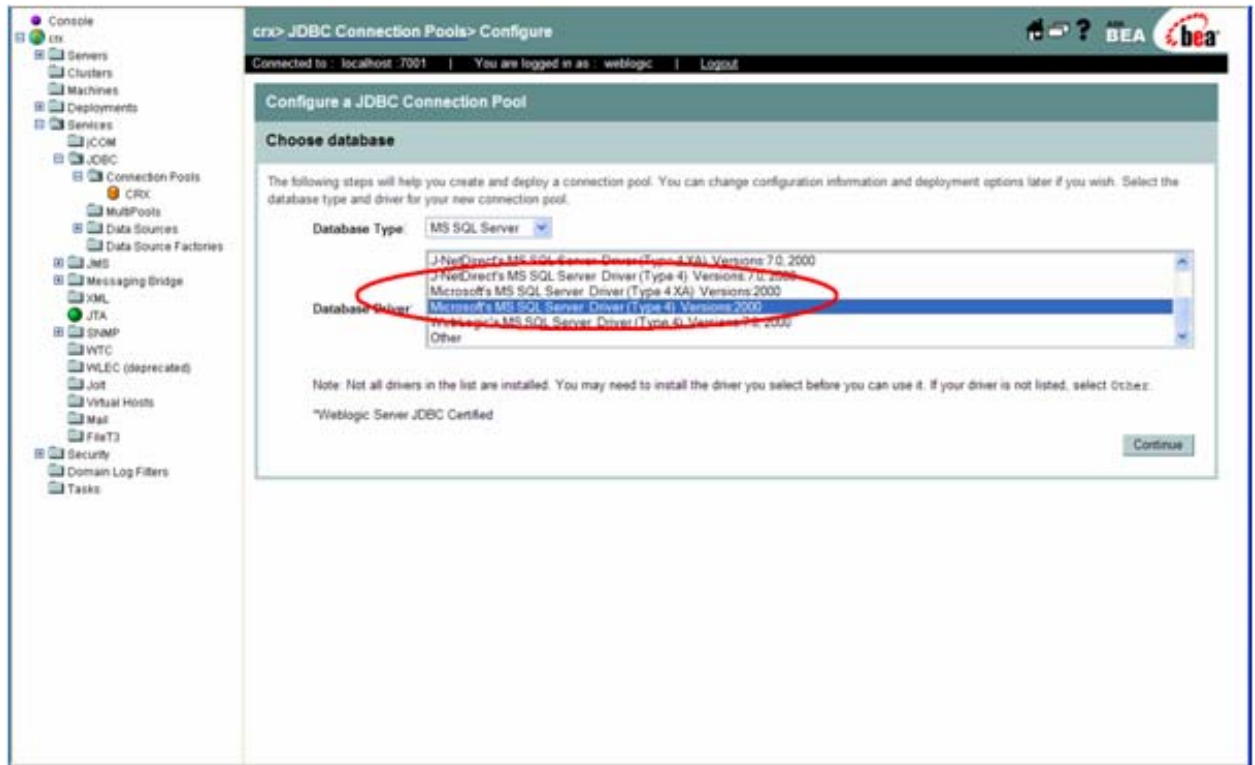
Navigate to *Services > JDBC > Connection Pools* and click on *Configure a new JDBC Connection Pool*. Select the database vendor from the drop down, e.g. *MS SQL Server* as shown in *Figure 6-1*.

Figure 6-1. Select the JDBC database vendor.



Then select the driver from the *Database Driver* drop down, e.g. *Microsoft's MS SQL Server Driver (Type) Version 2000* as shown in *Figure 6-2*. Then click continue.

Figure 6-2. Select the JDBC driver.

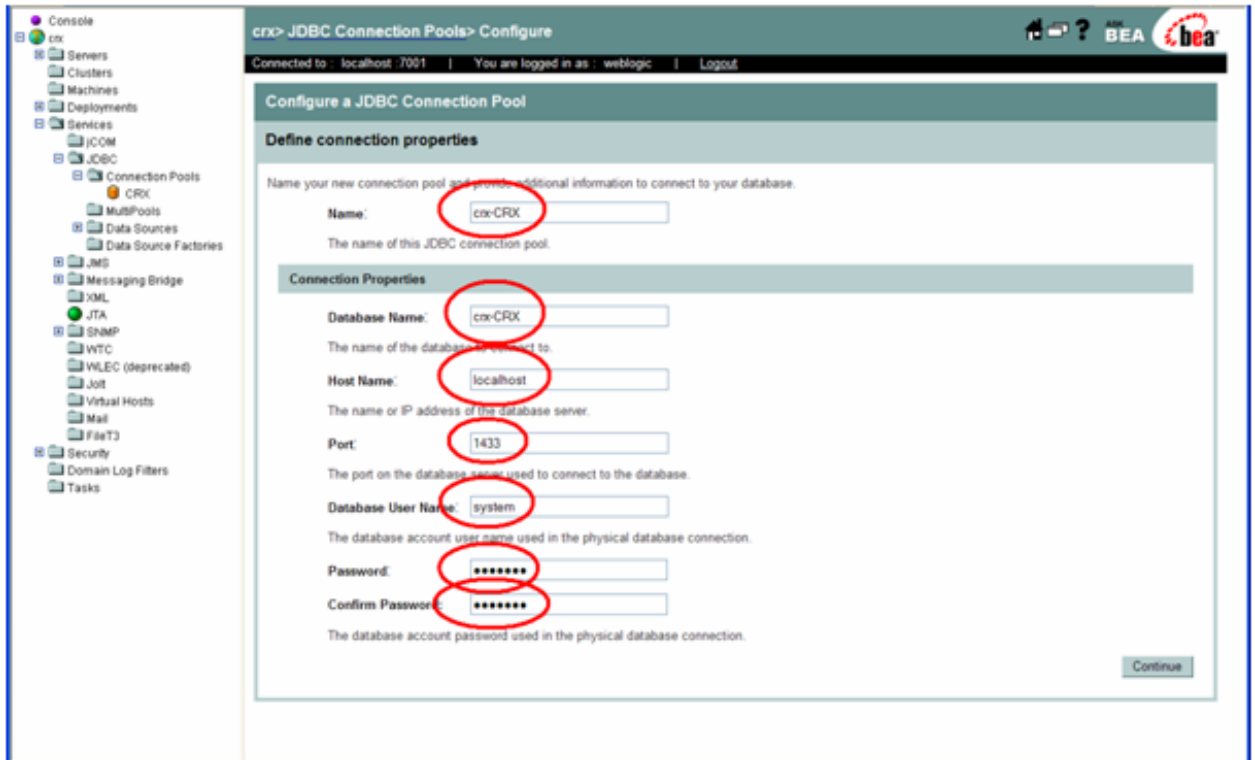


On the next form you must configure the connection pool. You must set the following properties:

- **Name.** Enter the name of the connection pool *crx-CRX*.
- **Database Name.** The name must match the name of your installed database, typically *crx-CRX*.
- **Host Name.** Enter the host name of your database server, e.g. *localhost*.
- **Port.** Enter the port number which is used for your database service, e.g. for *Microsoft SQL Server 2000* this is typically *1433*.
- **Database User Name / Password.** Enter the user name and password which is used as database login.

This is shown in *Figure 6-3*.

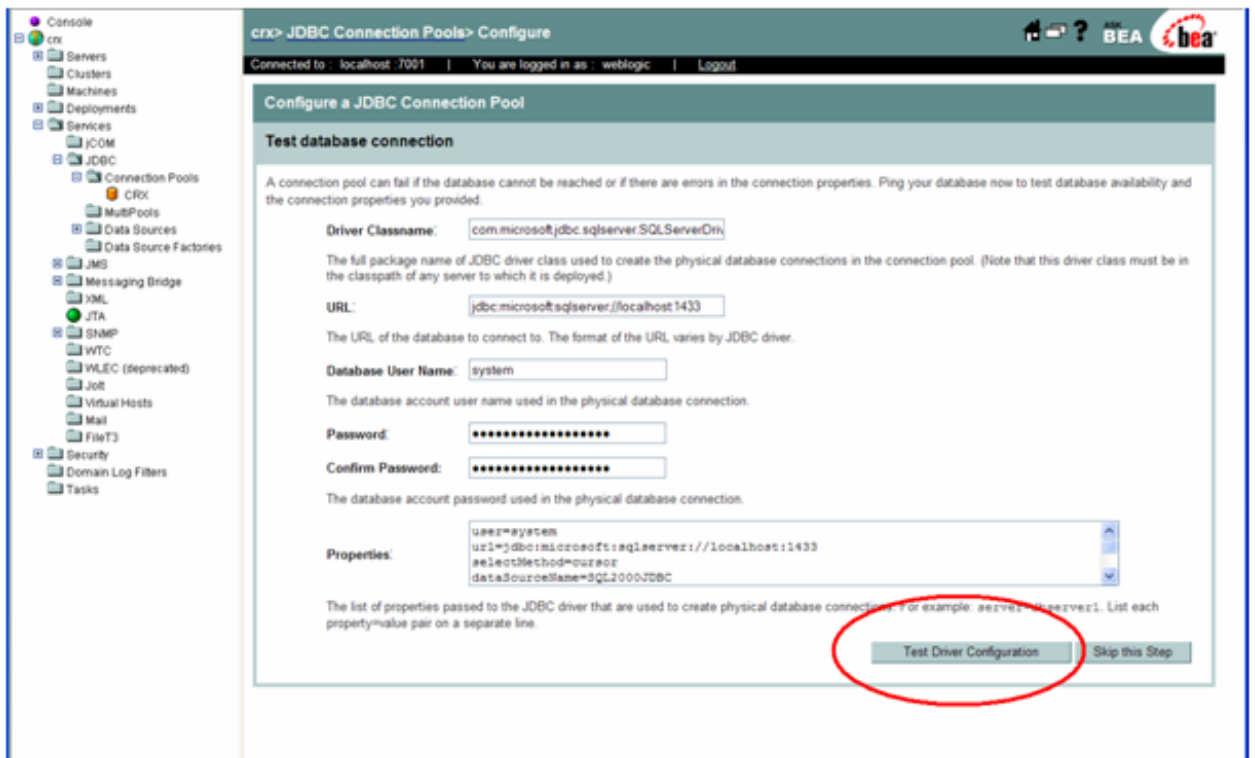
Figure 6-3. Configure the connection pool.



Then click continue.

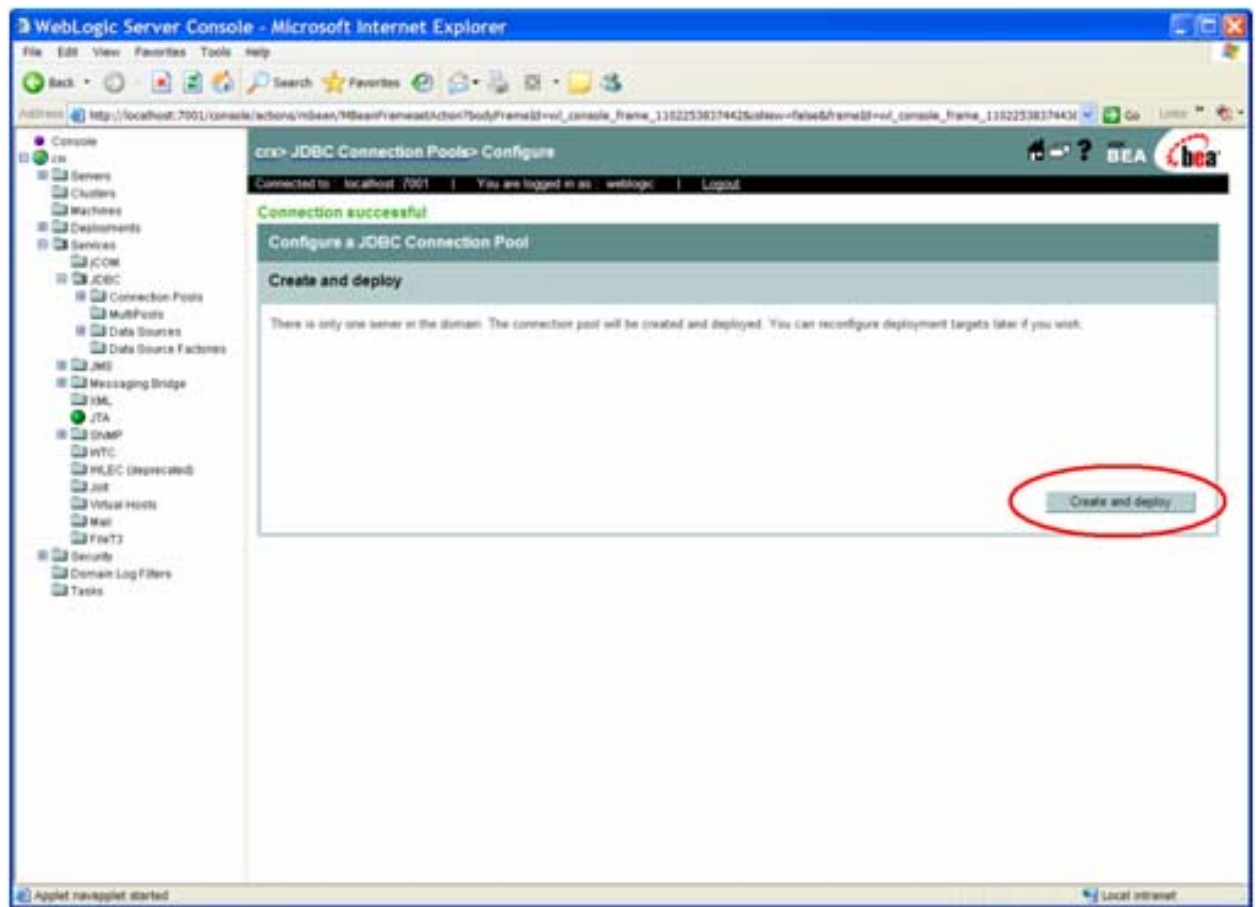
Verify the database connection properties on the next page as shown in *Figure 6-4* and then click *Test Driver Configuration*.

Figure 6-4. Configure the connection pool.



If everything works fine you should see the *Connection successful* message as shown in *Figure 6-5*. Click *Create and deploy*. This finally creates the connection pool.

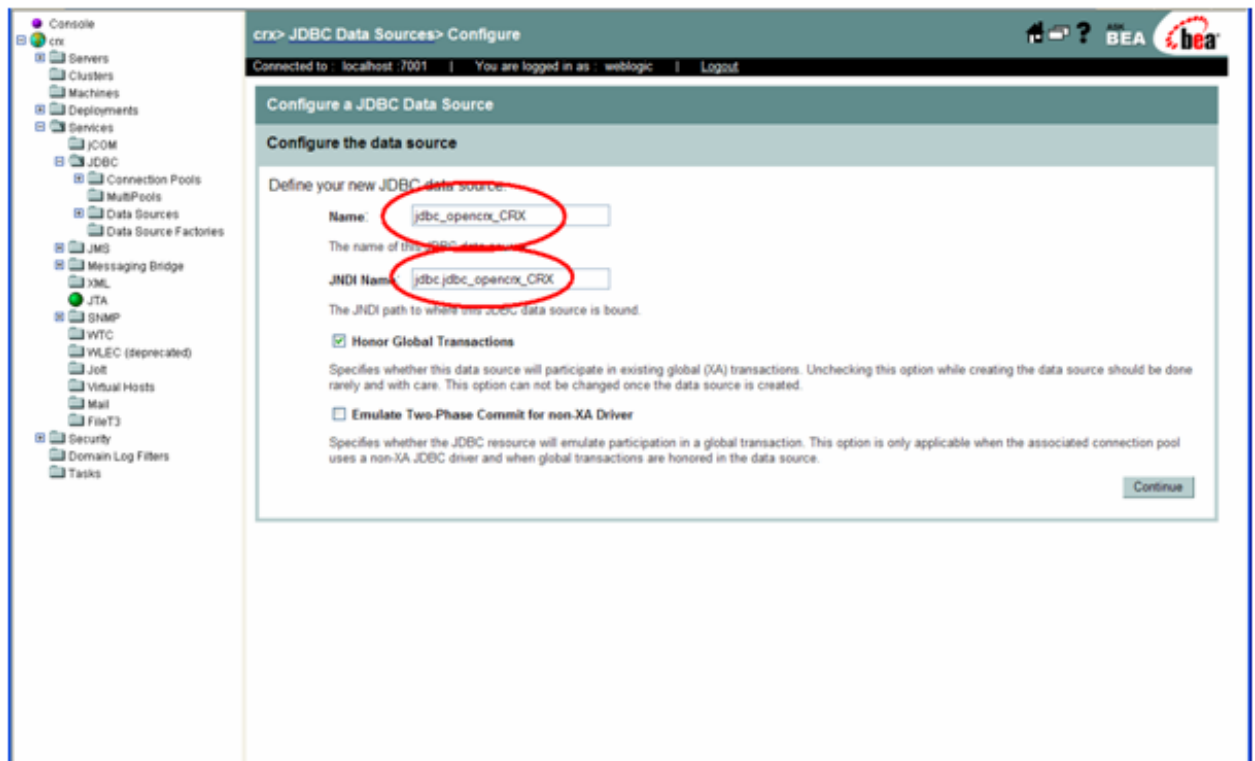
Figure 6-5. Create and deploy the connection pool.



As next step you must create a datasource using the connection pool you have just created. Navigate to *Services > JDBC > Data Sources* and click *Configure a new JDBC Data Source* as shown in Figure 6-6. You must set the properties as follows:

- **Name.** *jdbc_opencrx_CRX*.
- **JNDI Name.** *jdbc.jdbc_opencrx_CRX*. It is very important that you set the JNDI correctly. The datasource is referenced from the *openCRX* application by this name.

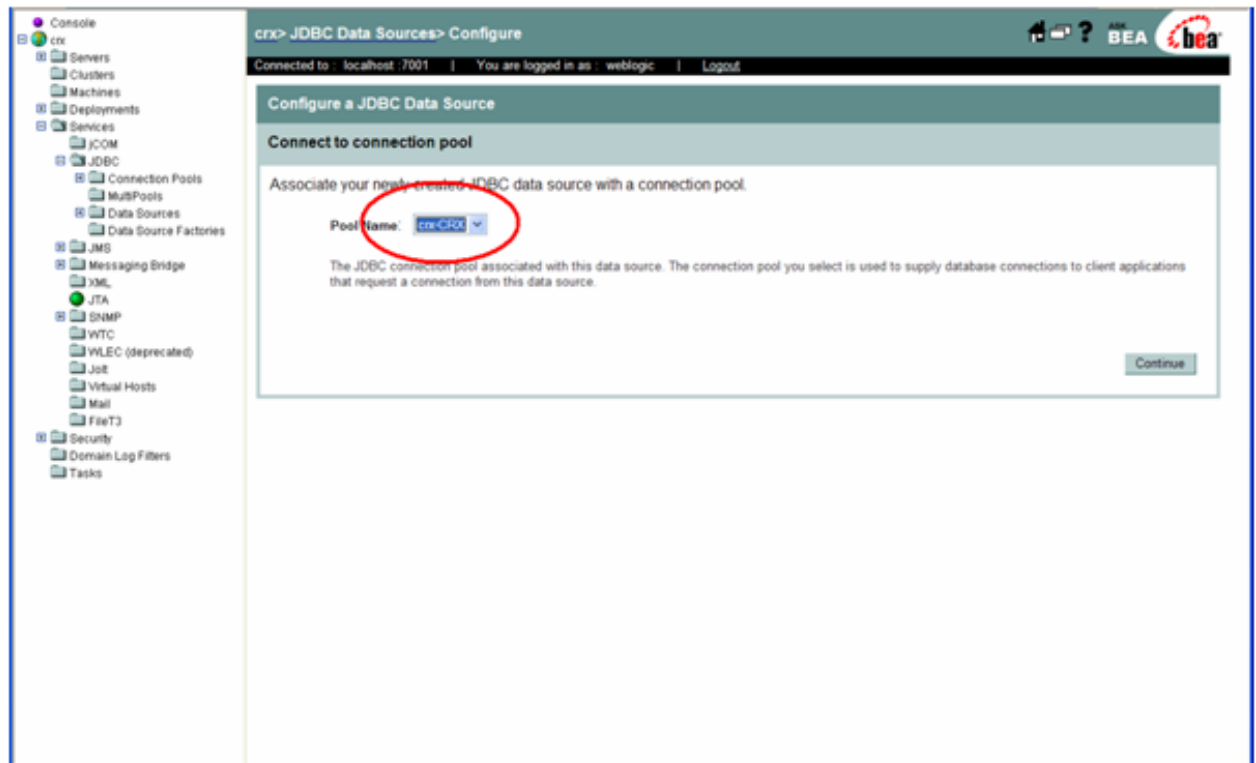
Figure 6-6. Create and configure the datasource.



Then click *Continue*.

On the next page you must select the database connection pool from the drop down. Select the pool you have just created, i.e. *crx-CRX* as shown in *Figure 6-7*.

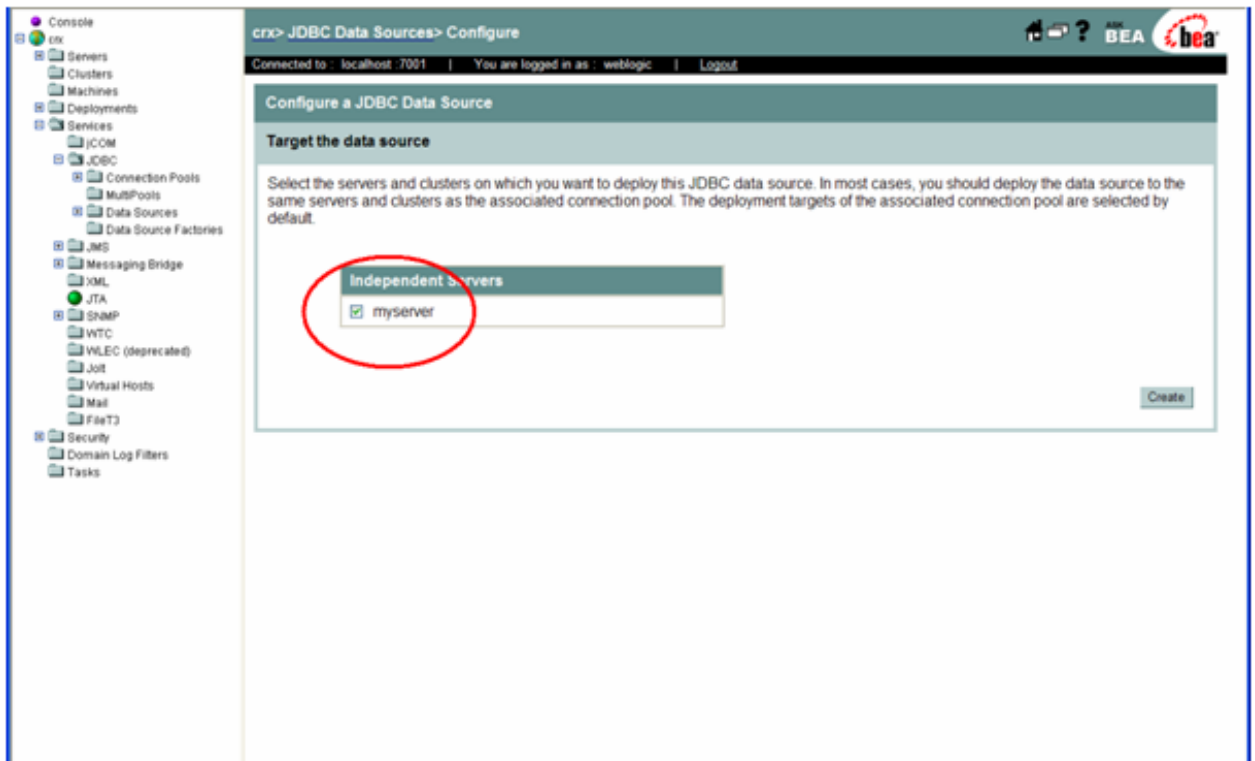
Figure 6-7. Select the connection pool.



Then click *Continue*.

On the next page you must select the servers where the new datasource should be deployed to. You must select the servers to which you plan to deploy the *openCRX* application. This is shown in *Figure 6-8*.

Figure 6-8. Select the deployment targets for the datasource.



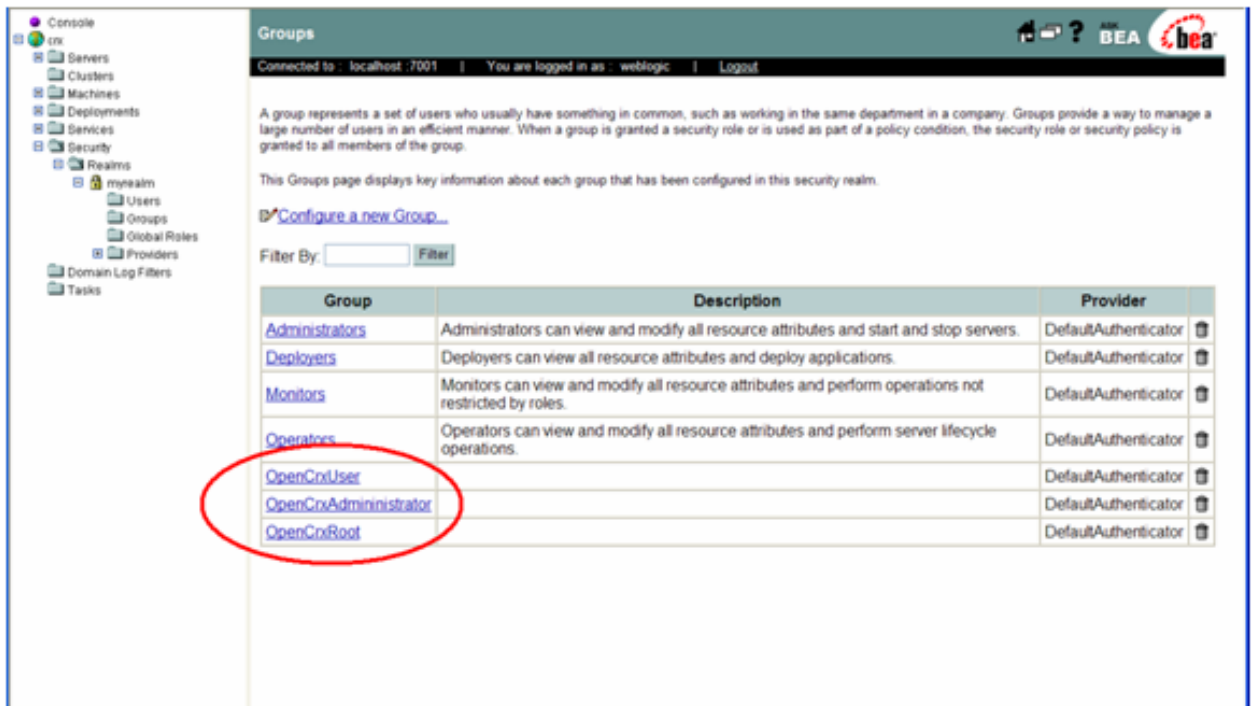
Then click *Create*. You are now finished creating the datasource for *openCRX*.

Chapter 7. Configuring Security

openCRX requires that each user is properly authenticated. This allows *openCRX* to correlate a session to user-specific application data and to perform access control. *openCRX* does not support non-authenticated sessions. User authentication must be configured as follows:

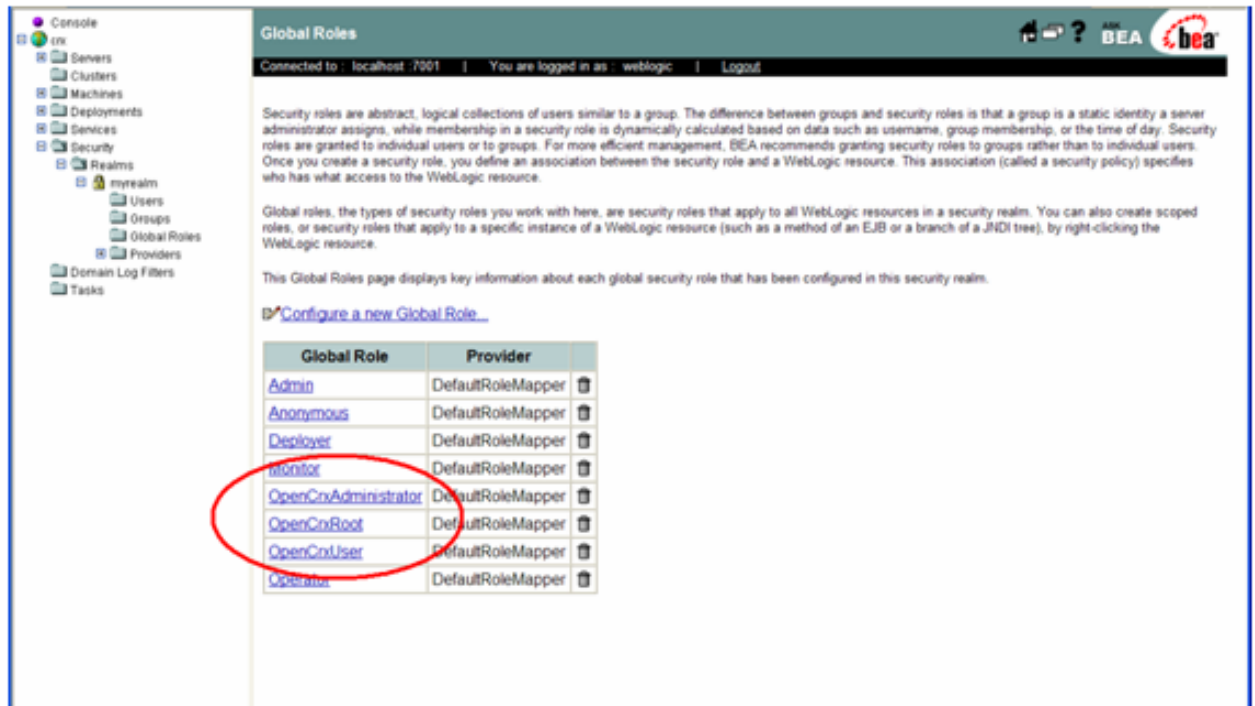
- Select *Security > Realms > myrealm*.
- Then click *Groups > Configure a new Group*. You must create the following groups: *OpenCrxUser*, *OpenCrxAdministrator*, *OpenCrxRoot* as shown in *Figure 7-1*.

Figure 7-1. Create user groups.



Then select *Global Roles > Configure a new Global Role*. Create the roles *OpenCrxRoot*, *OpenCrxAdministrator* and *OpenCrxUser* as shown in *Figure 7-2*.

Figure 7-2. Create global security roles.

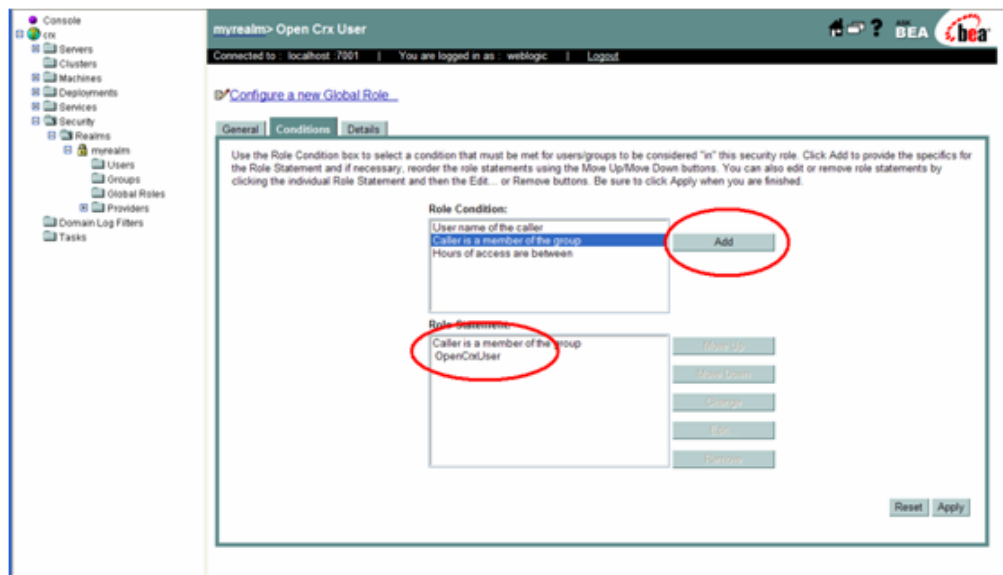


As a next step you must add the *Caller is a member of the group* condition to the newly created global roles as follows:

- Select the global role *OpenCrxUser* and select the tab *Conditions*. In the *Role Condition* pane select the entry *Caller is a member of the group* and then click *Add*. Enter the group name *OpenCrxUser* and save the modifications with *Apply*.
- Select the global role *OpenCrxAdministrator* and select the tab *Conditions*. In the *Role Condition* pane select the entry *Caller is a member of the group* and then click *Add*. Enter the group name *OpenCrxAdministrator* and save the modifications with *Apply*.
- Select the global role *OpenCrxRoot* and select the tab *Conditions*. In the *Role Condition* pane select the entry *Caller is a member of the group* and then click *Add*. Enter the group name *OpenCrxRoot* and save the modifications with *Apply*.

This is shown in *Figure 7-3*.

Figure 7-3. Add member of the group condition to global role.

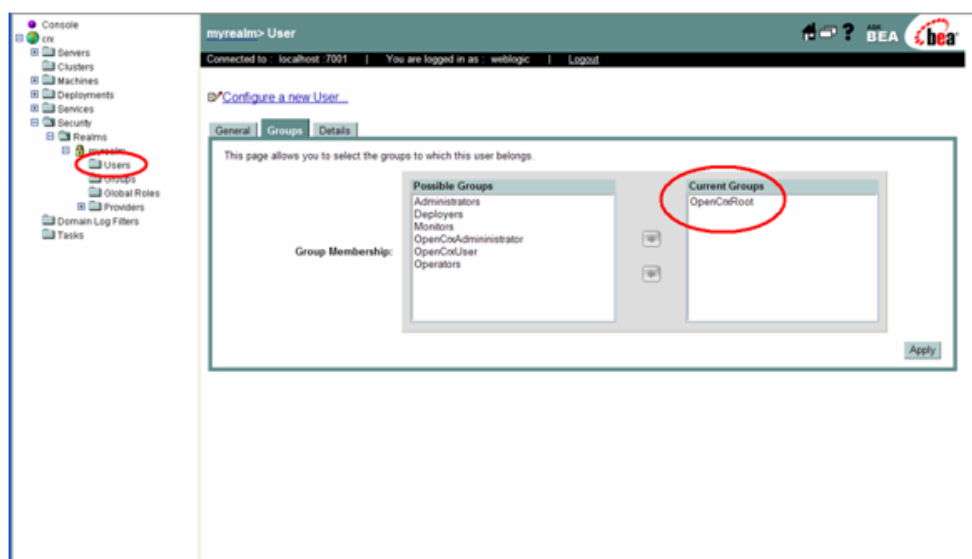


You now have successfully mapped the role names defined by the *openCRX* application to *WebLogic*-defined role names.

You are now ready to create users and make them member of the groups *OpenCrxRoot*, *OpenCrxAdministrator* or *OpenCrxUser*. A user must be member of *OpenCrxRoot* if he/she wants to access the servlet *opencrx-core-CRX-Root*. The servlet *opencrx-core-core-CRX* requires to be member of the groups *OpenCrxUser* or *OpenCrxAdministrator*. Standard users are member of the group *OpenCrxUser*. Segment administrators are member of the group *OpenCrxAdministrator*. This is shown in Figure 7-4. As a start you can add the following users:

- **admin-Root:** root user which is allowed to access the Root servlet. Make it member of the group *OpenCrxRoot*.
- **admin-Standard:** administrator for segment standard. Make the user member of the group *OpenCrxAdministrator*.
- **user1:** test user. Make it member of *OpenCrxUser*.

Figure 7-4. Create user and make member of group.



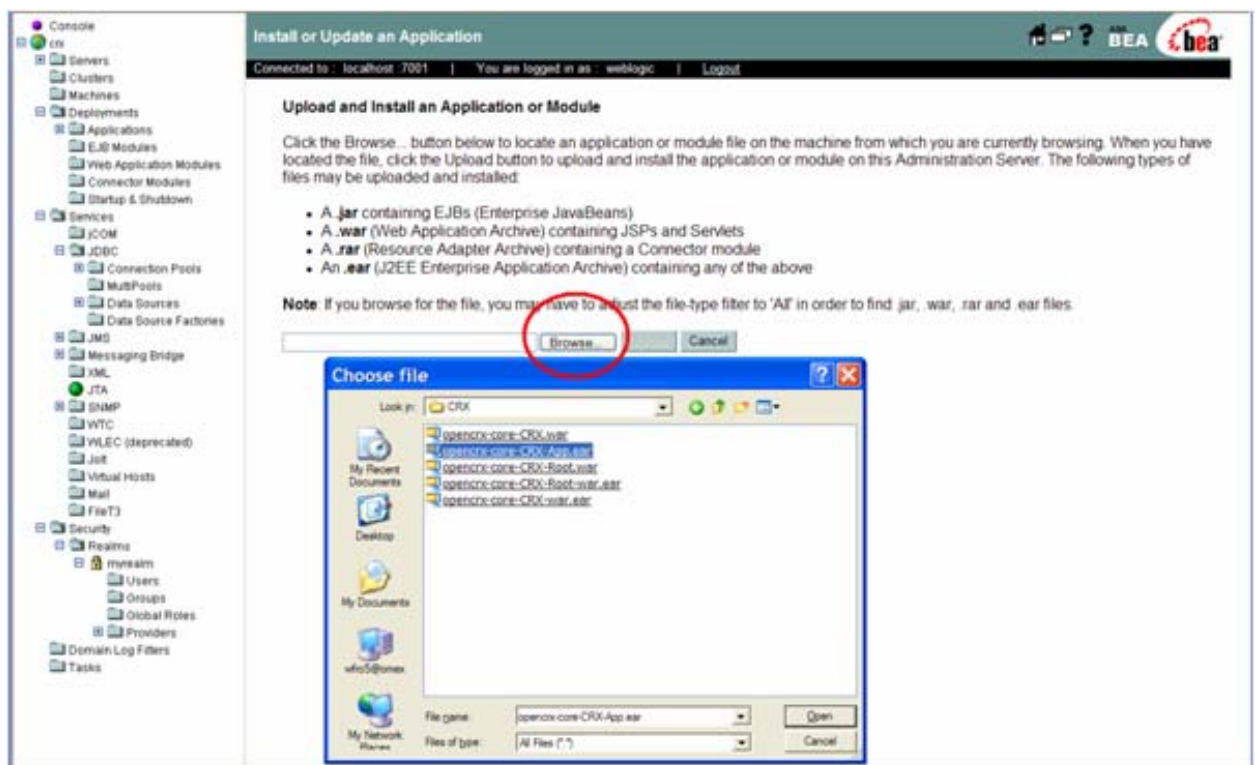
Chapter 8. Deploying openCRX

openCRX comes with three enterprise application archives (EAR):

- **opencrx-core-CRX-App.ear**. contains the *openCRX* server components, i.e. Enterprise Java Beans.
- **opencrx-core-CRX-web.ear**. contains the web application for standard *openCRX* users.
- **opencrx-core-CRX-Root-web.ear**. Contains the web application for root *openCRX* users.

In a first step you deploy *opencrx-core-CRX-App.ear*. Select *Deployments > Applications > Deploy a new Application > Upload your files*. Click *Browse* and select the file *opencrx-core-CRX-App.ear* as shown in *Figure 8-1*.

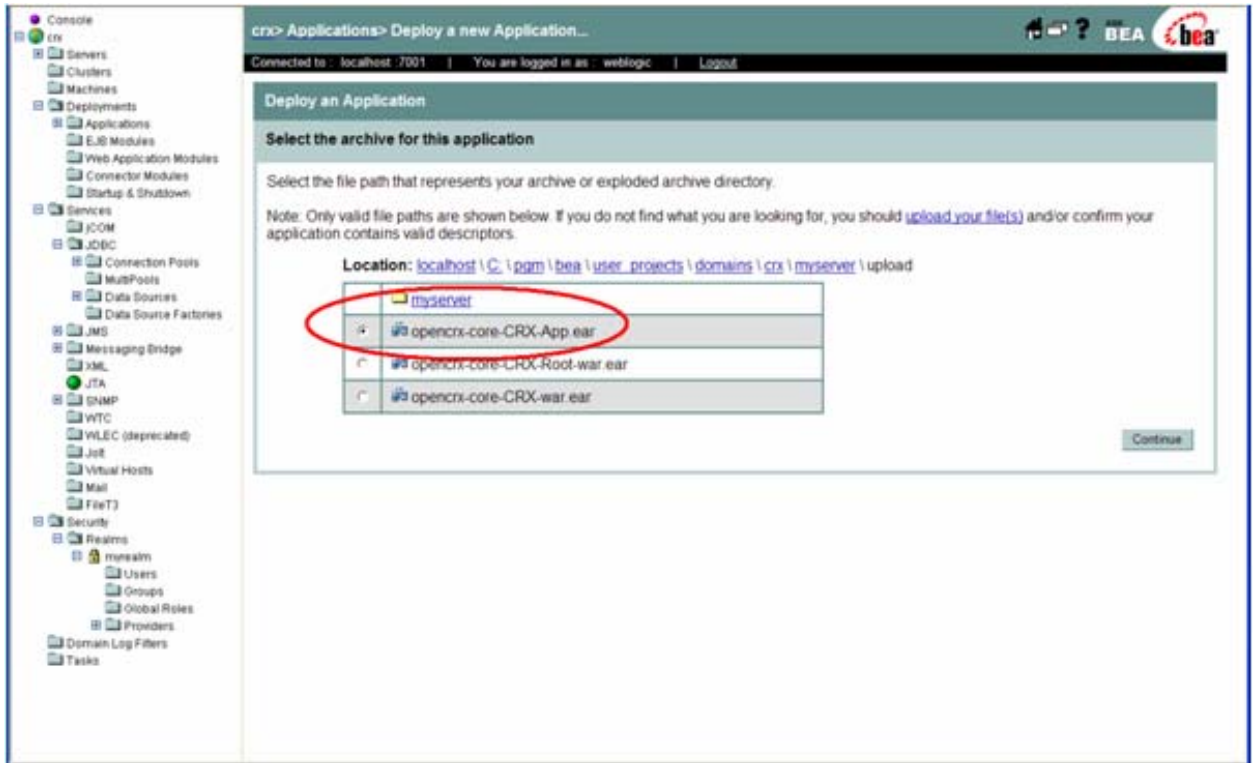
Figure 8-1. Upload the *opencrx-core-CRX-App* application enterprise archive.



Repeat this step for the files *opencrx-core-CRX-Root-web.ear* and *opencrx-core-CRX-web.ear*. The files are now ready for deployment.

Navigate now to *Deployments > Applications > Deploy a new Application* and then to the upload directory of your administration server, e.g. *domains/mydomain/myserver/upload*. In this directory you should find the files you have just uploaded. Select the file *opencrx-core-CRX-App.ear* as shown in *Figure 8-2*.

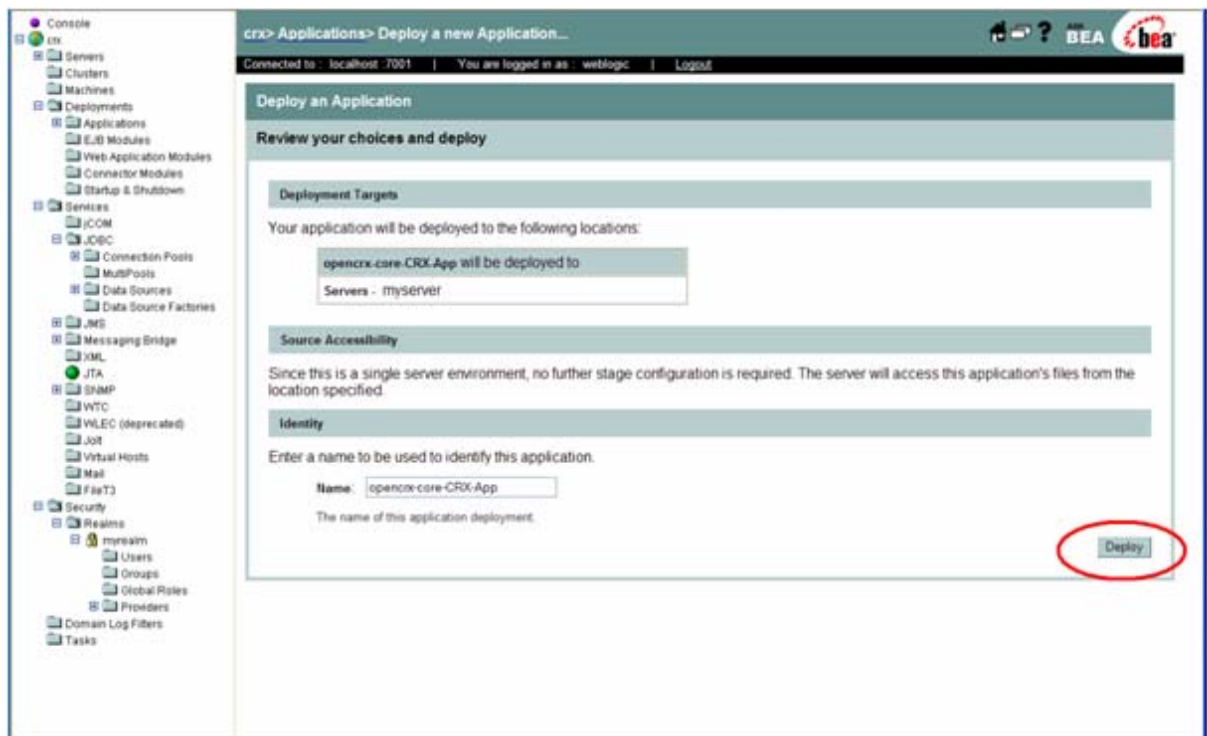
Figure 8-2. Deploy the *opencrx-core-CRX-App.ear* enterprise archive.



Then click *Continue*.

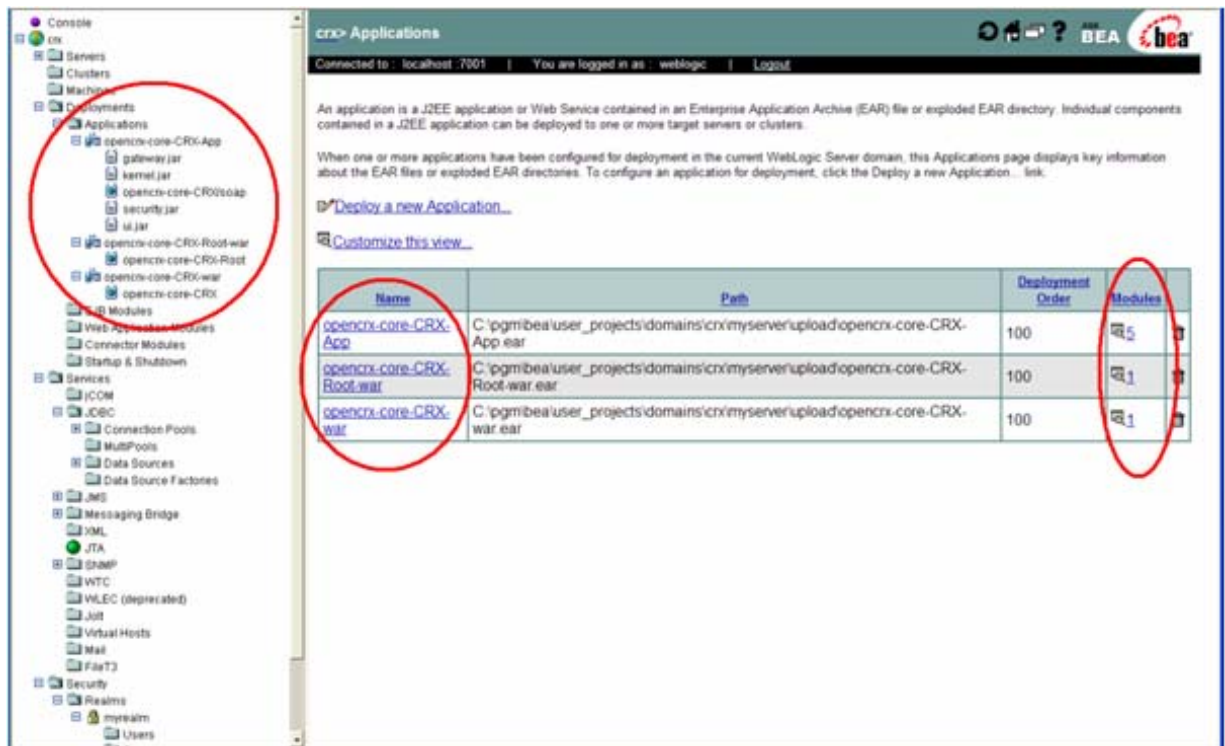
On the next screen you must specify the deployment targets and the application name. Leave the default values unchanged and then click *Deploy* as shown in *Figure 8-3*.

Figure 8-3. Select the deployment target and assign name.



Repeat the deployment step for the archives *opencrx-core-CRX-Root-web.ear* and *opencrx-core-CRX-web.ear*. Finally, the tree of the deployed applications should look as shown in Figure 8-4.

Figure 8-4. Verify the installation.



Chapter 9. Next Steps

Before you proceed to the *openCRX QuickStart guide* make sure that you have deployed and started all applications.

The application is initialized the first time a user calls the login page. If the startup fails you should consult the following log files:

- **WebLogic console and access logs.** Located in *bea\user_projects\domains\mydomain\myserver* and contains the console and the access log.
- **openCRX opencrx-server1...log.** Located in *bea\user_projects\domains\mydomain\log* and contains the *openCRX* application log files.

Appendix A. Appendix

Bibliography

[01] *openCRX - the leading open source CRM solution*, [opencrx.org](http://www.opencrx.org).

@ <http://www.opencrx.org> (<http://www.opencrx.org>)

[02] *openMDX - The leading open source MDA platform*, [openmdx.org](http://www.openmdx.org).

@ <http://www.openmdx.org> (<http://www.openmdx.org>)