BEA WebLogic Adapter for J.D. Edwards® OneWorld®

Installation and Configuration Guide

Release 7.0
Document Date: October 2002

Copyright

Copyright © 2002 BEA Systems, Inc. All Rights Reserved.

Copyright © 2002 iWay Software. All Rights Reserved.

Restricted Rights Legend

This software and documentation is subject to and made available only pursuant to the terms of the BEA Systems License Agreement and may be used or copied only in accordance with the terms of that agreement. It is against the law to copy the software except as specifically allowed in the agreement. This document may not, in whole or in part, be copied photocopied, reproduced, translated, or reduced to any electronic medium or machine readable form without prior consent, in writing, from BEA Systems, Inc.

Use, duplication or disclosure by the U.S. Government is subject to restrictions set forth in the BEA Systems License Agreement and in subparagraph (c)(1) of the Commercial Computer Software-Restricted Rights Clause at FAR 52.227-19; subparagraph (c)(1)(ii) of the Rights in Technical Data and Computer Software clause at DFARS 252.227-7013, subparagraph (d) of the Commercial Computer Software--Licensing clause at NASA FAR supplement 16-52.227-86; or their equivalent.

Information in this document is subject to change without notice and does not represent a commitment on the part of BEA Systems. THE SOFTWARE AND DOCUMENTATION ARE PROVIDED “AS IS” WITHOUT WARRANTY OF ANY KIND INCLUDING WITHOUT LIMITATION, ANY WARRANTY OF MERCHANTABILITY OR FITNESS FOR A PARTICULAR PURPOSE. FURTHER, BEA Systems DOES NOT WARRANT, GUARANTEE, OR MAKE ANY REPRESENTATIONS REGARDING THE USE, OR THE RESULTS OF THE USE, OF THE SOFTWARE OR WRITTEN MATERIAL IN TERMS OF CORRECTNESS, ACCURACY, RELIABILITY, OR OTHERWISE.

Trademarks or Service Marks

BEA, Jolt, Tuxedo, and WebLogic are registered trademarks of BEA Systems, Inc. BEA Builder, BEA Campaign Manager for WebLogic, BEA eLink, BEA Manager, BEA WebLogic Commerce Server, BEA WebLogic Enterprise, BEA WebLogic Enterprise Platform, BEA WebLogic Express, BEA WebLogic Integration, BEA WebLogic Personalization Server, BEA WebLogic Platform, BEA WebLogic Portal, BEA WebLogic Server, BEA WebLogic Workshop and How Business Becomes E-Business are trademarks of BEA Systems, Inc.

All other trademarks are the property of their respective companies.

BEA WebLogic Adapter for J.D. Edwards Installation and Configuration Guide

Part Number	Date
N/A	October 2002

Table of Contents

About This Document

This document explains how to install, configure, and deploy the BEA WebLogic Adapter for J.D. Edwards OneWorld to develop online connections to OneWorld applications using BEA WebLogic Integration.

This document is organized as follows:

- Chapter 1, "Installing the Adapter for WebLogic Integration 7.0," directs you to the information you will need before installing the adapter and describes how to install the adapter.
- Chapter 2, "Installing and Configuring the OneWorld Event Listener," describes how to install and configure the J.D. Edwards OneWorld Event Listener for use with specific business functions.

Audience

This document is written for system integrators with programming backgrounds and an understanding of the J.D. Edwards OneWorld product in an application space. Extensive knowledge of J.D. Edwards OneWorld is not required, but may be helpful in learning about the adapter.

Related Information

The following documents provide additional information for the associated software components:

- *BEA WebLogic Adapter for J.D. Edwards OneWorld User Guide*
- *BEA WebLogic Adapter for J.D. Edwards OneWorld Release Notes*
- *BEA Application Explorer Installation and Configuration Guide*
- BEA WebLogic Server installation and user documentation, which is available at the following URL:

  `http://edocs.bea.com/more_wls.html`
- BEA WebLogic Integration installation and user documentation, which is available at the following URL:

  `http://edocs.bea.com/more_wli.html`

Contact Us!

Your feedback on the BEA WebLogic Adapter for J.D. Edwards OneWorld documentation is important to us. Send us e-mail at docsupport@bea.com if you have questions or comments. Your comments will be reviewed directly by the BEA professionals who create and update the adapter documentation.

In your e-mail message, please indicate which version of the adapter documentation you are using.

If you have any questions about this version of the adapter, or if you have problems installing and running it, contact BEA Customer Support through BEA WebSupport at www.bea.com. You can also contact Customer Support by using the contact information provided on the Customer Support Card, which is included in the product package.

When contacting Customer Support, be prepared to provide the following information:

- Your name, e-mail address, phone number, and fax number
- Your company name and company address
- Your machine type and authorization codes
- The name and version of the product you are using
- A description of the problem and the content of pertinent error messages

Documentation Conventions

The following conventions are used throughout this document.

Convention	Item
boldface text	Indicates terms defined in the glossary.
Ctrl+Tab	Indicates that you must press two or more keys simultaneously.
italics	Indicates emphasis or book titles.

`monospace text`	Indicates code samples, commands and their options, data structures and their members, data types, directories, and file names and their extensions. Monospace text also indicates text that you must enter from the keyboard.
	Examples:
	`#include <iostream.h> void main ( ) the pointer psz`
	`chmod u+w *`
	`\tux\data\ap`
	`.doc`
	`tux.doc`
	`BITMAP`
	`float`

Convention	Item
monospace boldface text	Identifies significant words in code. *Example*: `void` **`commit`** `( )`
monospace italic text	Identifies variables in code. *Example*: `String` *`expr`*
UPPERCASE TEXT	Indicates device names, environment variables, and logical operators. *Example*s: LPT1 SIGNON OR
{ }	Indicates a set of choices in a syntax line. The braces themselves should never be typed.

[ ]	Indicates optional items in a syntax line. The brackets themselves should never be typed.
	Example:
	`buildobjclient [-v] [-o name ] [-f` *`file-list`*`]...`
	`[-l` *`file-list`*`]...`

|	Separates mutually exclusive choices in a syntax line. The symbol itself should never be typed.

. . .	Indicates one of the following in a command line:
	■ That an argument can be repeated several times in a command line
	■ That the statement omits additional optional arguments
	■ That you can enter additional parameters, values, or other information
	The ellipsis itself should never be typed.
	Example:
	`buildobjclient [-v] [-o name ] [-f` *`file-list`*`]...`
	`[-l` *`file-list`*`]...`

. . .	Indicates the omission of items from a code example or from a syntax line. The vertical ellipsis itself should never be typed.

1 Installing the Adapter for WebLogic Integration 7.0

This section provides instructions for installing the BEA WebLogic Adapter for J.D. Edwards OneWorld with WebLogic Integration. It includes the following topics:

Before Installing the Adapter

Before you install the BEA WebLogic Adapter for J.D. Edwards OneWorld, you must review the BEA WebLogic Adapter for J.D. Edwards OneWorld *Release Notes* to ensure that you have the required prerequisite software installed. The BEA WebLogic Adapter for J.D. Edwards OneWorld *Release Notes* is available at the following URL:

`http://edocs.bea.com/wladapters/doc70/index.html`

Understanding the Representation of Paths

Because the location of files in the WebLogic Integration environment depends on options selected during installation and configuration, the conventions that follow are used throughout to represent paths.

- *BEA_HOME* represents the BEA Home directory specified for your WebLogic installation.

  For example, if you install the product in the default location on a Windows system, *BEA_HOME* represents `c:\bea`.

- *WLI_HOME* represents the root of your WebLogic Integration installation.

  For example:

  - If you install WebLogic Integration 7.0 in the default location on a Windows system, *WLI_HOME* represents `c:\bea\weblogic700\integration`.

- *domain* is used to indicate the name of a domain.

  - In WebLogic Integration 7.0, a new tool, the Configuration Wizard, is used to create custom user domains. When you use the Configuration Wizard to set up the domain configuration stored on the administration server, you are prompted to assign a domain name, *domain*, and to specify the location to which the *domain* directory will be installed. The files required by the administration server are installed in the *domain* directory under the specified location.

For additional information, see *Using the Configuration Wizard* which is available at the following URL:

`http://edocs.bea.com/platform/docs70/confgwiz/index.html`

For example, if you accept the Configuration Wizard default location, *BEA_HOME*\user_projects, the files required by the administration server are installed in the following directory:

BEA_HOME\user_projects*domain*

- *DOMAIN_HOME* represents the complete path to the root of a domain.

  For example:

Note: *WLI_HOME* and *BEA_HOME* (italicized) also represent the corresponding Windows and UNIX environment variables. For example, the literal interpretation of *WLI_HOME* is `%WLI_HOME%` for Windows and `$WLI_HOME` for UNIX.

Unlike *WLI_HOME* and *BEA_HOME*, *DOMAIN_HOME* is not an environment variable that is set by default in the WebLogic Integration environment.

Step 1. Obtaining the BEA WebLogic Adapter for J.D. Edwards OneWorld

The BEA WebLogic Adapter for J.D. Edwards OneWorld software is packaged as an EAR file (`BEA_JDEDWARDSOW_1_0_70.ear`). You can obtain the software on CD or download it from `www.bea.com`.

Step 2. Configuring the Domain

The adapter can only be deployed in a domain that includes support for application integration functionality. If you have not already done so, use the Configuration Wizard to create the domain and select one of the following domain templates:

- Enterprise application integration (EAI) domain template
- WebLogic Integration (WLI) domain template
- Platform domain template

For the information you need to configure a fully functional domain based on the template, see the appropriate section of the *Configuration Wizard Template Reference*:

- For the EAI domain template, see the following URL:

  `http://edocs.bea.com/platform/docs70/template/eaidomain.html`

- For the WLI domain template, see the following URL:

  `http://edocs.bea.com/platform/docs70/template/wlidomain.html`

- For the platform domain template, see the following URL:

  `http://edocs.bea.com/platform/docs70/template/platjar.html`

For general information about using the Configuration Wizard, see *Using the Configuration Wizard* at the following URL:

`http://edocs.bea.com/platform/docs70/confgwiz/index.html`

When you use the Configuration Wizard to create a domain based on the WLI or EAI domain template, a domain-specific version of the Database Wizard is installed in the *DOMAIN_HOME* directory. You must run the Database Wizard for the domain to initialize the database repository with the required tables and system data. For additional information, see "Configuring the Database for a Domain" in *Starting, Stopping and Customizing BEA WebLogic Integration*, which is available at the following URL:

`http://edocs.bea.com/wli/docs70/config/index.htm`

Step 3. Adjusting Classpath

Set the classpath using the procedure appropriate for your system:

- Adjusting Classpath for Windows
- Adjusting Classpath for UNIX

Adjusting Classpath for Windows

To adjust the classpath on Windows, complete the following steps:

1. Go to the `\systems\classes` directory in your J.D. Edwards environment.
2. Copy the following files from the the `\systems\classes` directory to a directory of your choice (for example, *`BEA_HOME`*`\AdapterEars\jde\`):
   - `Connector.jar`
   - `Kernel.jar`
3. Go to the root directory for your domain:

   `cd` *`DOMAIN_HOME`*

   Note: You must choose a domain that supports application integration functionality.
4. Find the `Set`*`DomainType`*`Data.cmd` file.

   Here, *`DomainType`* is the type of the domain. For example, if your domain is configured to support the development of solutions that employ the full range of WebLogic Integration functionality, it contains the `SetwliDomainData.cmd` file.

5. Update the `SVRCP` environment variable settings in the `Set`*`DomainType`*`Data.cmd` file.

   Note: The `SVRCP` environment variable is used in the `Set`*`DomainType`*`Data` script to set the classpath for the java executable.

   After the following line:

   ```
   set SVRCP=%SVRCP%;%WLI_DOMAIN_HOME%\applications\DefaultWebApp_
   myserver\WEB-INF\classes
   ```

   add the following JAR files:

   ```
   set SVRCP=%SVRCP%;BEA_HOME\AdapterEars\jde\Connector.jar
   set SVRCP=%SVRCP%;BEA_HOME\AdapterEars\jde\Kernel.jar
   ```

Here, *`BEA_HOME\AdapterEars\jde`* is the directory specified in step 1.

Adjusting Classpath for UNIX

To adjust the classpath on UNIX, complete the following steps:

1. Go to the `/systems/classes` directory in your J.D. Edwards environment.
2. Copy the following files from the the `/systems/classes` directory to a directory of your choice (for example, *`BEA_HOME/AdapterEars/jde/`*):
   - `Connector.jar`
   - `Kernel.jar`
3. Go to the root directory for your domain:

   ```
   cd DOMAIN_HOME
   ```

   Note: You must choose a WebLogic Integration domain that supports application integration functionality.
4. Find the `Set`*`DomainType`*`Data.cmd` file.

   Here, *`DomainType`* is the type of the domain. For example, if your domain is configured to support the development of solutions that employ the full range of WebLogic Integration functionality, it contains the `SetwliDomainData.cmd` file.

5. Update the SVRCP environment variable settings in the Set*DomainType*Data.cmd file.

   Note: The SVRCP environment variable is used in the Set*DomainType*Data script to set the classpath for the java executable.

   After the following line:

   ```
   SVRCP=$SVRCP:$WLI_DOMAIN_HOME/applications/DefaultWebApp_myserver/
   WEB-INF/classes
   ```

   add the following JAR files:

   ```
   SVRCP=$SVRCP:BEA_HOME/AdapterEars/jde/Connector.jar
   SVRCP=$SVRCP:BEA_HOME/AdapterEars/jde/Kernel.jar
   ```

Here, *BEA_HOME/AdapterEars/jde* is the directory specified in step 1.

Step 4. Updating the BEA License

The BEA WebLogic Adapter for J.D. Edwards OneWorld cannot be used without a valid software license. If you have downloaded the adapter for evaluation, you must obtain an evaluation license as described on the adapter download page. If you have purchased a license for the adapter, the license file is typically sent to you as an e-mail attachment.

When you have obtained a valid license for the adapter, update your license.bea file by completing the following steps:

1. Save the license file that you obtained with a name other than license.bea, in the *BEA_HOME* directory. For example, save the file as jdedwardsow_adapter_license.bea. Use this file as the *license_update_file* in step 4 of this procedure.

   Warning: Do not overwrite or change the name of the existing license.bea file.

2. Perform the step appropriate for your platform:

   - On a Windows system, open an MS-DOS session and go to the *BEA_HOME* directory.
   - On a UNIX system, go to the *BEA_HOME* directory.

3. If it is not already included, add the JDK to your `PATH` variable by executing the command appropriate to your system:
   - On a Windows system:

     ```
     set PATH=BEA_HOME\jdk131_03\bin;%PATH%
     ```

   - On a UNIX system:

     ```
     PATH=BEA_HOME/jdk131_03/bin:$PATH
     export PATH
     ```

4. Merge the license update file into your existing license by executing the command appropriate to your system:
   - On a Windows system:

     ```
     UpdateLicense license_update_file
     ```

   - On a UNIX system:

     ```
     sh UpdateLicense.sh license_update_file
     ```

   Here, *`license_update_file`* is the name to which you saved the license update file in step 1.

5. Save a copy of your updated `license.bea` file in a safe place outside the WebLogic Integration and application installation directories.

Step 5. Deploying the Adapter Using the WebLogic Server Console

After the BEA WebLogic Adapter for J.D. Edwards OneWorld is installed, it must be deployed to your domain. To configure and deploy an adapter using the WebLogic Server Administration Console, complete the following steps:

1. Start WebLogic Server.
2. Start the WebLogic Server Administration Console in a browser using the following URL:

   ```
   http://host:port/console/
   ```

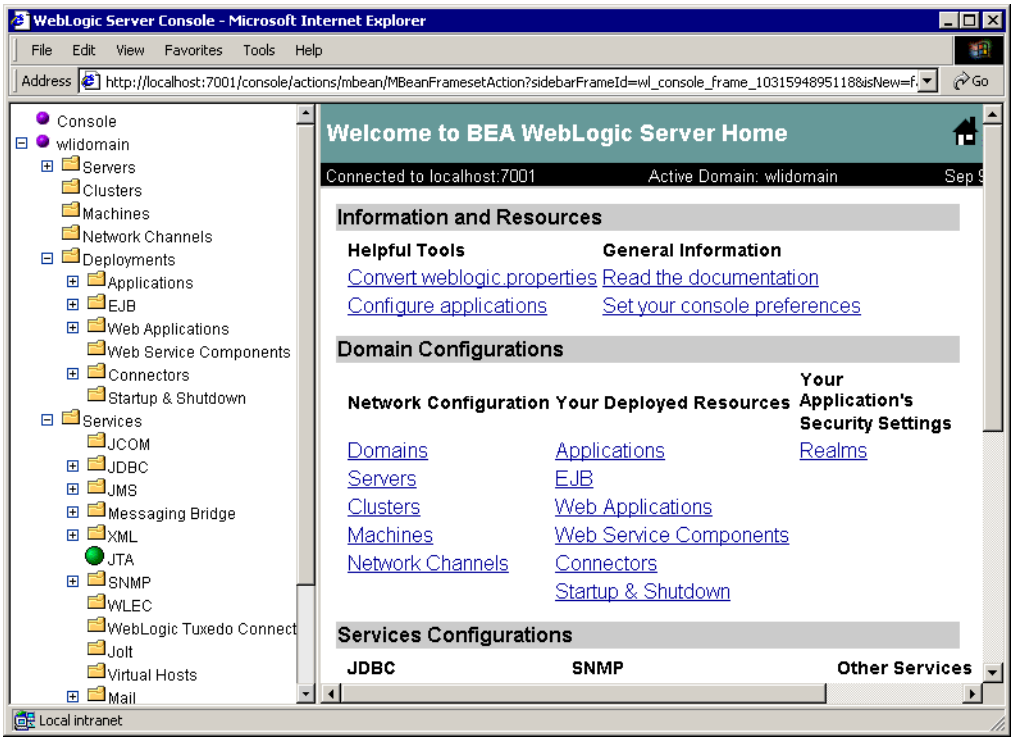
Here, *host* represents the machine on which WebLogic Server is running and *port* represents the listening port.

For example, `http://localhost:7001/console/`

3. When prompted, enter the user name and password for the server.

   The WebLogic Server Administration Console opens.

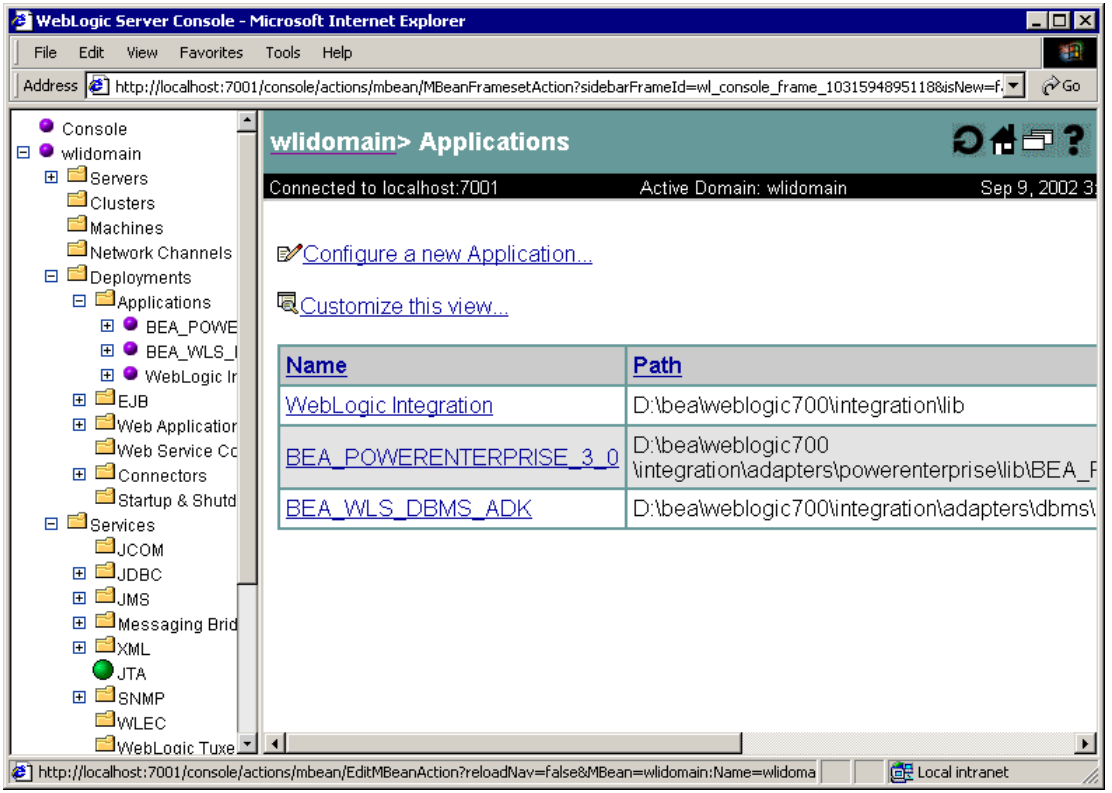
Figure 1-1 WebLogic Server Console



4. In the left pane, choose Deployments and then Applications from the navigation tree.

The console displays the Applications window.

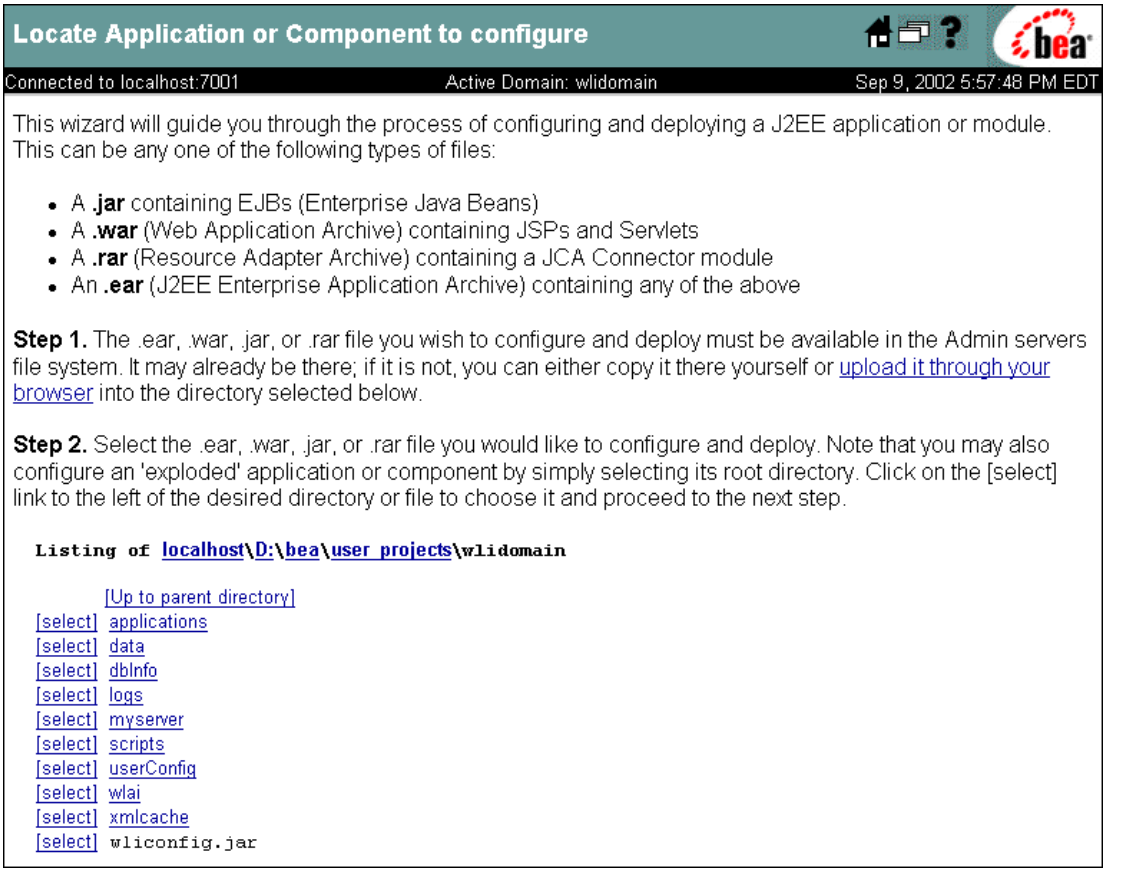
Figure 1-2 Applications Window



5. Click the Configure a new Application link.

   The console displays the Locate Application or Component to configure window.

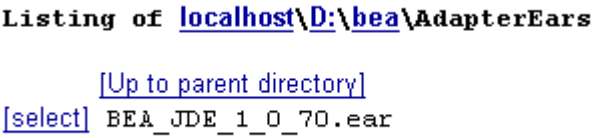
Figure 1-3 Locate Application or Component to Configure Window



6. Do one of the following:

   If you copied the `BEA_JDEDWARDSOW_1_0_70.ear` file to a directory on the administrative server, navigate to that directory. For example, if you copied the file to the `D:\bea\AdapterEars` directory, select the directory as shown in the following figure.

Figure 1-4 Locating the Adapter EAR File



If you have not copied the `BEA_JDEDWARDSOW_1_0_70.ear` file to a directory on the administrative server, do the following:

a. Navigate to the directory to which the file will be uploaded.

   For example, to upload the adapter from your local machine to the `D:\bea\AdapterEars` directory on the administrative server, select the directory as shown in the following figure.

   Figure 1-5 Selecting the Target Directory

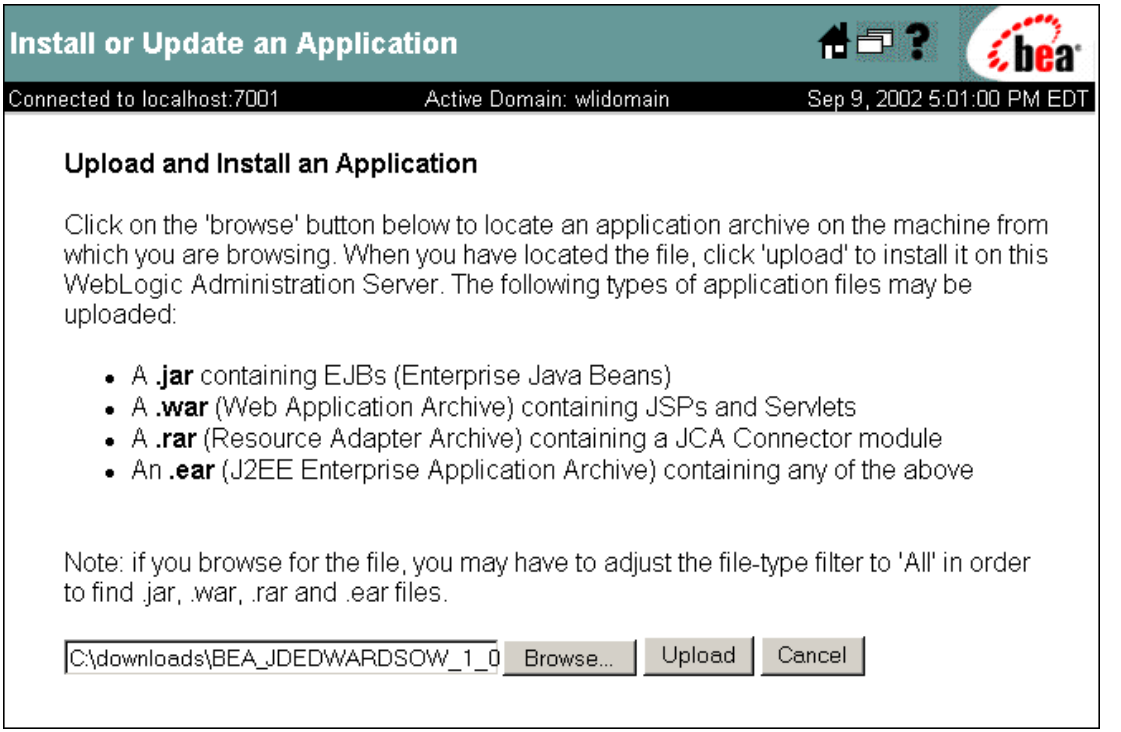
b. Click the upload it through your browser link to display the Install or Update an Application window.

c. Click Browse to display the Choose File dialog box.

d. In the Choose File dialog box, locate the file and then click Open.

   The console displays the selected location.

Figure 1-6 Install or Update an Application Window



e. Click Upload.

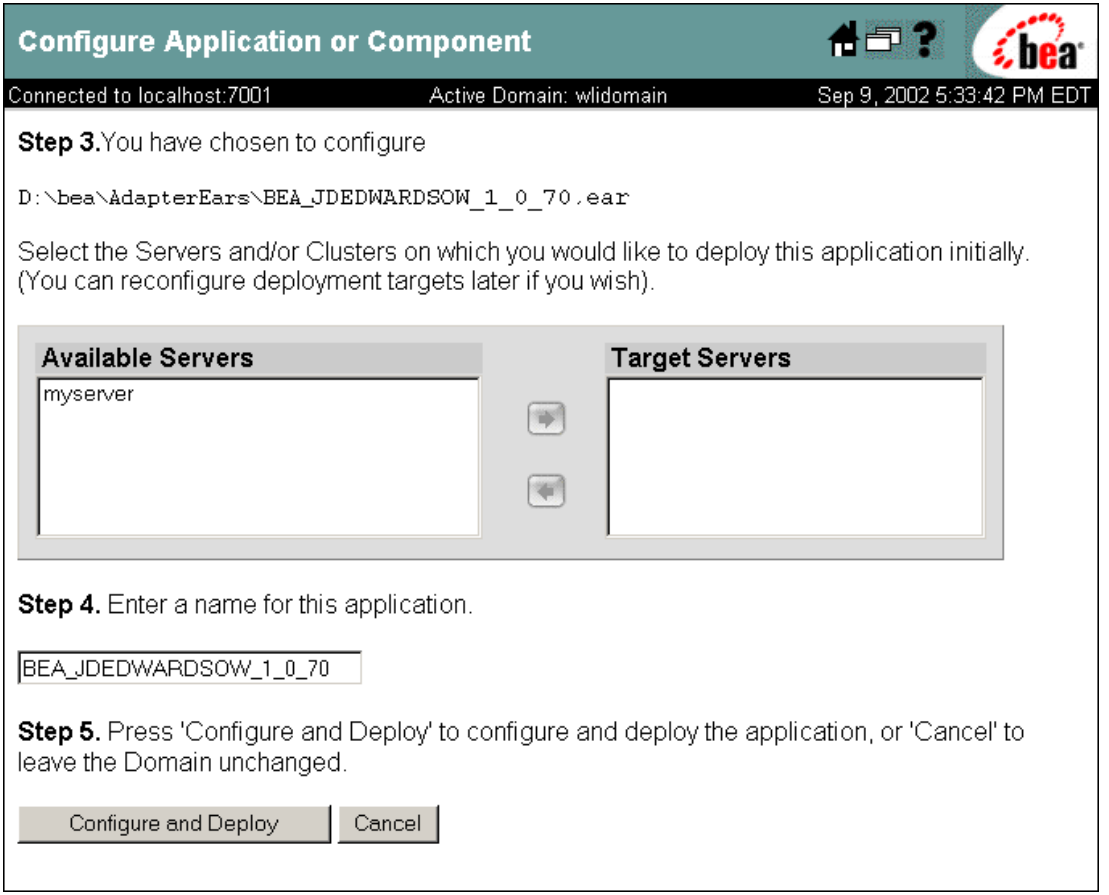
The browser status bar indicates upload progress. When the upload is complete, you are returned to the Locate Application or Component to configure window. The uploaded file now resides in the directory selected in step a.

Figure 1-7 Adapter EAR File Uploaded to Administrative Server

7. Click the [select] link to the left of the adapter EAR file.

   The console displays the Configure Application or Component window.

Figure 1-8 Configure Application or Component Window

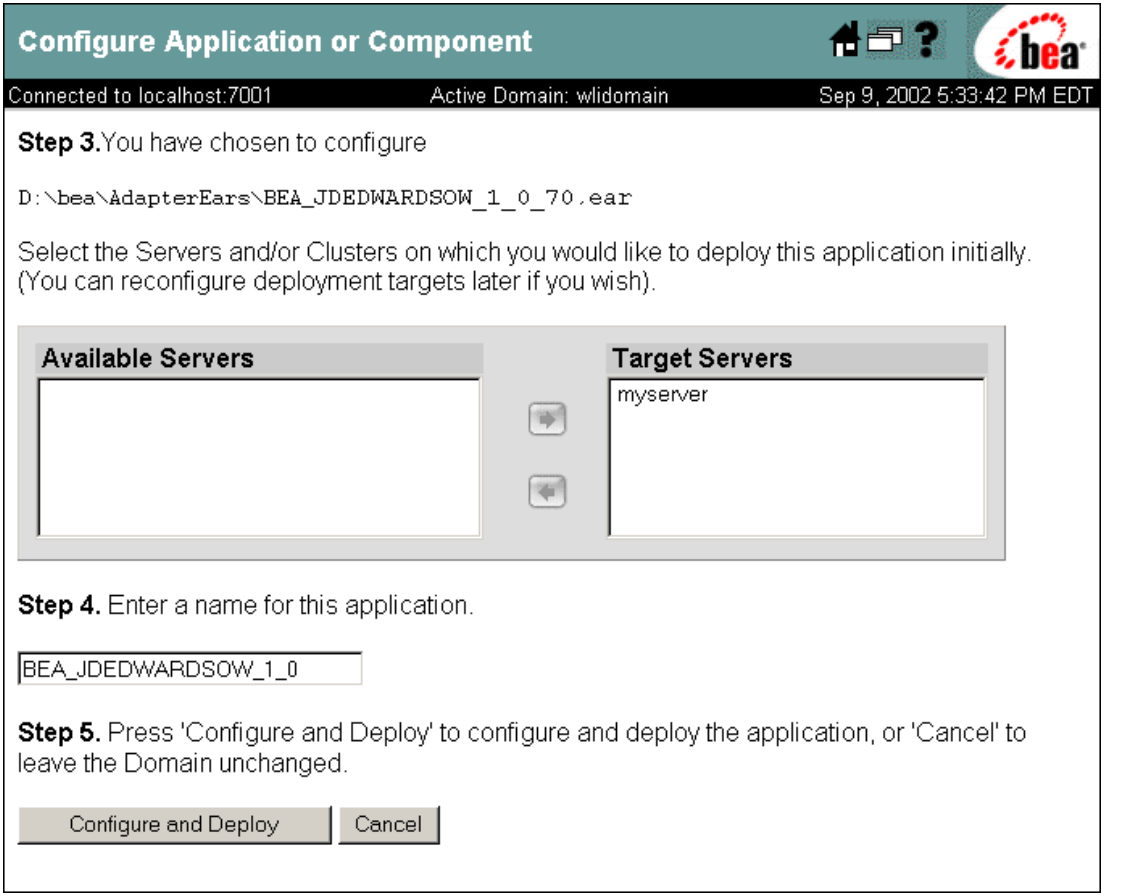


8. Select the servers or clusters on which to deploy the adapter by using the arrow buttons to move entries from the available list to the target list.
9. You must change the application name from `BEA_JDEDWARDSOW_1_0_70` to `BEA_JDEDWARDSOW_1_0`.

   Note: If you do not change the application name, the adapter will not be available to application views.

The following figure shows a configured application.

Figure 1-9 Configured Application



10. Click Configure and Deploy.

    The console displays the deployment status, which includes a description, status, begin time, and end time for the deployed adapter.

Step 6. Creating or Updating the Adapter Group

Before you create an application view that uses the events or services supported by an adapter, the following requirements must be met:

- The user group, `adapter`, must be defined.
- The administrative server user name must be a member of the `adapter` group.

The action required to complete the configuration depends on which domain template you selected when you created the domain. See the following table for guidelines.

Table 1-1 Configuration Requirements

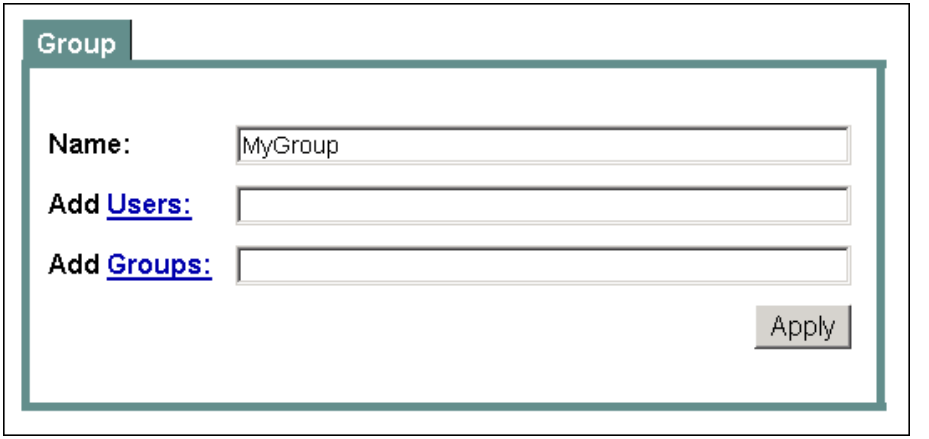
If you created a domain based on the . . .	And the administrative server user name is . . .	Then . . .
Platform domain template	Any value	Create the adapter group and add the administrative user name to it as described in "Creating the Adapter Group" on page 1-17.
WLI or EAI domain template	`system`	The `adapter` group is already defined. This group includes the `system` user name by default. No further configuration is required.
WLI or EAI domain template	A value other than `system`	The `adapter` group is already defined. Add the user name to the group as described in "Adding the User Name to the Adapter Group" on page 1-18.

Creating the Adapter Group

To create the `adapter` group and add the administrative server user name to it, complete the following steps:

1. In the left pane of the WebLogic Server Administration Console, choose Compatibility Security and then Groups from the navigation tree.
2. Click the Create a New Group link to display the Group window.

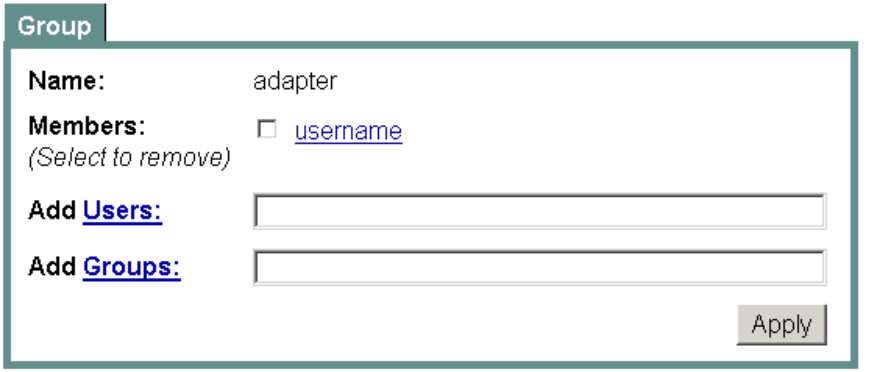
Figure 1-10 Group Window



3. Enter `adapter` in the Name field.
4. Enter the administrative server user name in the Add Users field.
5. Click Apply.

   The Group window is updated as shown in the following figure.

Figure 1-11 Group Window



Adding the User Name to the Adapter Group

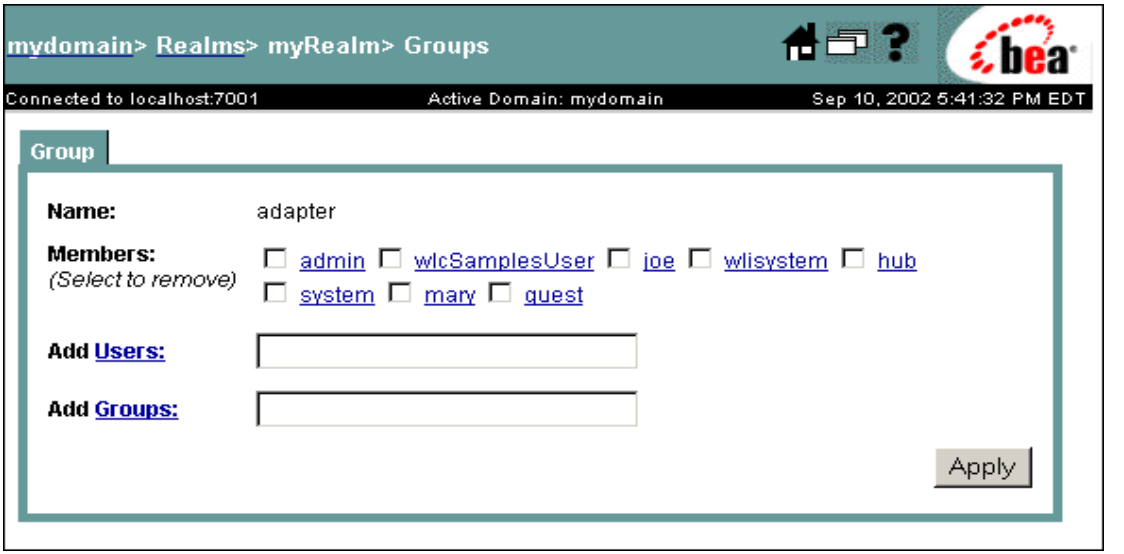
To add the administrative server user name to the `adapter` group, complete the following steps:

1. In the left pane of the WebLogic Server Administration Console, choose Compatibility Security and then Groups from the navigation tree.

   The console displays the groups currently defined for the domain.

2. Locate and click the link for the `adapter` group to display the group definition.

Figure 1-12 Group Definition



3. If the administrative server user name is not included in the Members list, enter the user name in the Add Users field.
4. Click Apply to add the user name to the group.

   The name is added to the Members list.

Next Steps

When you have successfully installed and deployed the adapter, you can log on to the WebLogic Integration Application View Console to create application views that employ events and services supported by the BEA WebLogic Adapter for J.D. Edwards OneWorld. For more information, see the BEA WebLogic Adapter for J.D. Edwards OneWorld *User Guide*.

2 Installing and Configuring the OneWorld Event Listener

The OneWorld Event Listener is invoked by J.D. Edwards OneWorld for specific business functions as configured in the OneWorld environment. For related information, refer to Appendix A, "Configuring J.D. Edwards OneWorld for Outbound Processing," in the *BEA WebLogic Adapter for J.D. Edwards OneWorld User Guide*.

This section describes how to install and configure the J.D. Edwards OneWorld Event Listener for use with specific business functions. It includes the following topics:

How the Event Listener Is Supplied

The J.D. Edwards OneWorld Event Listener (`owevent`) is supplied in the `BEA_JDEDWARDSOW_SAMPLES.zip` file, in the directory path `\Listeners\operating_ system\owevent.ext`. For example, for Microsoft Windows 2000, the module is located in the `\Listeners\win32\` subdirectory and is named `plugin.dll`.

Installing the OneWorld Event Listener

To install the J.D. Edwards OneWorld Event Listener, perform the following steps:

1. Create a directory in which the module will reside. For example, go to the BEA home directory, and enter the following

   `mkdir` *`dirname`*

   Here, *`dirname`* is the name of the subdirectory to be created.

Note: If WebLogic Server is not installed on the same computer as the J.D. Edwards OneWorld application server, you may need to create a BEA root directory first.

2. Use a utility such as Winzip or jar to extract the version of the module required for your J.D. Edwards OneWorld server operating system from the `BEA_JDEDWARDSOW_ SAMPLES.zip` file.

3. On the J.D. Edwards OneWorld server, create a separate directory for the `owevent.cfg` file (optional).

4. On the J.D. Edwards OneWorld server, create an `owevent.cfg` file in the defined directory. Refer to the section "Creating owevent.cfg" on page 2-3 for information on the contents of that file.
5. Create an environment variable `OWEVENT_HOME` to point to the directory containing the `owevent.cfg` file.
   - On Windows: Add `OWEVENT_HOME` to the system environment variables.
   - On UNIX: Add the command

     ```
     export OWEVENT_HOME =/directory_name
     ```

     to your startup script.

Creating owevent.cfg

For the OneWorld Event Listener to properly initiate events in WebLogic Integration, the information required to connect to WebLogic Server must be supplied. This information is contained in the `owevent.cfg` file. You must create this file and add the connection information to it.

A sample `owevent.cfg` file is included in the `BEA_JDEDWARDSOW_ SAMPLES.zip` file.

Add the Server and Port entries to `owevent.cfg`.

For example,

```
Server=ipaddress or dsn
Port=nnnn
```

Here,

- *`ipaddress`* or *`dsn`* is the IP address or the DSN of WebLogic Server.
- *`nnnn`* is the local port defined for the event.

For example:

```
Server=localhost
Port=4575
```

How the OneWorld Event Listener Works

The OneWorld Event Listener is comprised of the listener module (`owevent`), which is deployed under the J.D. Edwards OneWorld server, and the outbound agent (`XDJdeOutboundAgent`), which is deployed on WebLogic Server. The listener module passes the key fields from the outbound transaction table record for the event to WebLogic Server for processing. The outbound agent then uses the key fields to retrieve the event information.

The OneWorld Listener accesses the configuration file, called `owevent.cfg` (case sensitive), and based on the information in the file, sends the event notification to WebLogic Server. If WebLogic Server is unavailable or some exception occurs, the listener saves the event information in a file called *`timestamp`*`.xml`. In this file name, *`timestamp`* is a number indicating the time. All the log information is saved in a file called `owevent.log`.

When WebLogic Server receives an XML request from the listener exit, it invokes the XDJdeOutboundAgent to process the request. The XDJdeOutboundAgent creates a J.D. Edwards XML request and executes the request against the OneWorld system.

Sample Request and Response

Here is a sample request sent to J.D. Edwards OneWorld by the listener exit to retrieve event information.

Listing 2-1 Sample Request

```
<jdeRequest environment="DV7333" user="JDE" type="trans" sessionidle="
300" session="" pwd="JDE">
        <transaction type="JDESOOUT" action="transactionInfo">

             <key>
               <column name="EdiUserId">islywm</column>
               <column name="EdiBatchNumber">100</column>
               <column name="EdiTransactionNumber">100100</column>

             </key>
         </transaction>
</jdeRequest>
```

Here is a sample response from J.D. Edwards OneWorld as processed by the outbound agent.

Listing 2-2 Sample Response

```
<jdeResponse type='trans' user='user' session='session1'
                                          environment='env'>
     <transaction type='JDESOOUT' action='transactionInfo'>
        <returnCode code='0'>XML Request OK</returnCode>
           <key>
              <column name='EdiUserId'></column>
              <column name='EdiBatchNumber'></column>
              <column name='EdiTransactNumber'></column>
           </key>
           <table name='F4201Z1' type='header'>
               <column name='EdiUserId'></column>
               <column name='EdiBatchNumber'></column>
           </table>
           <table name='F4211Z1' type='detail'>
               <column name='EdiUserId'></column>
               <column name='EdiBatchNumber'></column>
           </table>
            <table name='F49211Z1' type='additionalHeader'>
                <WARNING>No record found</WARNING>
            </table>
       </transaction>
</jdeResponse>
```

The connection setting for retrieving information from J.D. Edwards OneWorld is defined at event creation time.