



BEA WebLogic Adapter for Oracle Applications[®]

Installation and Configuration Guide

Version 8.1.0
Document Date: July 2003

Copyright

Copyright © 2003 BEA Systems, Inc. All Rights Reserved.

Portions Copyright © 2003 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 Liquid Data for WebLogic, 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.

Contents

About This Document

Who Should Read This Documentation	v
Background Knowledge	vi
Additional Information	vi
Contact Us!	vii
Documentation Conventions	viii

Installing the BEA WebLogic Adapter for Oracle Applications

Preparing to Install the Adapter	1-2
Review the Release Notes	1-2
Install the Oracle Applications JDBC Driver	1-2
Understanding the Representation of Paths	1-2
Installing the Adapter	1-3
Step 1. Obtain the BEA WebLogic Adapter for Oracle Applications	1-4
Step 2. Configure the Domain	1-4
Step 3. Extract the Adapter Files and Change the WebLogic Script	1-4
Extracting Files for Windows	1-5
Extracting Jars and Adjusting the Classpath for UNIX	1-5
Step 4. Update the BEA License	1-6
Step 5. Deploy the Adapter	1-7
Step 6. Create an Adapter Administrative User	1-10
Next Steps	1-11

Installing the BEA Application Explorer.	1-12
Configuring the Adapter to Use Oracle Applications Concurrent Programs.	1-12
Starting Integration with Oracle Applications	1-12

Configuring the Concurrent Program Agent

About the Installation and Configuration Process.	A-1
Installing the Concurrent Programs Agent	A-3
Step 1: Run the Concurrent.ora Script	A-3
Step 2: Adjust the System Path	A-4
Step 3: Verify CONCSUB	A-4
Configuring the Concurrent Programs Agent	A-5
Creating Schemas for the Concurrent Programs Agent.	A-5
Adding a Service for the Concurrent Programs Agent	A-5
Testing the Concurrent Program Agent	A-7
Determining the Request Parameters.	A-7
Creating the Request Document	A-10
Submitting the Request and Viewing the Response.	A-13

Index

About This Document

This document explains how to install and configure the BEA WebLogic Adapter for Oracle Applications.

This document is organized as follows:

- [Chapter 1, “Installing the BEA WebLogic Adapter for Oracle Applications,”](#) explains how to install the adapter.
- [Appendix A, “Configuring the Concurrent Program Agent,”](#) explains how to install and configure the concurrent programs agent to use Oracle Applications concurrent programs.

Who Should Read This Documentation

This document is intended for the following members of an integration team:

- **Integration Specialists**—Lead the integration design effort. Integration specialists have expertise in defining the business and technical requirements of integration projects, and in designing integration solutions that implement specific features of WebLogic Integration. The skills of integration specialists include business and technical analysis, architecture design, project management, and WebLogic Integration product knowledge.
- **Technical Analysts**—Provide expertise in an organization’s information technology infrastructure, including telecommunications, operating systems, applications, data repositories, future technologies, and IT organizations. The skills of technical analysts include technical analysis, application design, and information systems knowledge.

- Enterprise Information System (EIS) Specialists—Provide domain expertise in the systems that are being integrated using WebLogic adapters. The skills of EIS specialists include technical analysis and application integration design.
- System Administrators—Provide in-depth technical and operational knowledge about databases and applications deployed in an organization. The skills of system administrators include capacity and load analysis, performance analysis and tuning, deployment topologies, and support planning.

Background Knowledge

This document assumes that you have an understanding of:

- Web technologies
- WebLogic Integration
- Oracle Applications software, system, and environment. This includes understanding of Oracle Applications tools and configuration as well as processes and data models.
- Your specific Oracle Applications business needs and applications.

Additional Information

To learn more about the software components associated with the adapter, see the following documents:

- *BEA WebLogic Adapter for Oracle Applications Release Notes*
<http://edocs.bea.com/wl.adapters/oracle/docs81/pdf/relnotes.pdf>
- *BEA WebLogic Adapter for Oracle Applications User Guide*
<http://edocs.bea.com/wl.adapters/oracle/docs81/pdf/user.pdf>
- *BEA Application Explorer Installation and Configuration Guide*
<http://edocs.bea.com/wl.adapters/bae/docs81/index.html>
- *Introduction to the BEA WebLogic Adapters for WebLogic Integration*
<http://edocs.bea.com/wl.adapters/docs81/index.html>
- BEA WebLogic Adapters 8.1 Dev2Dev Product Documentation
<http://dev2dev.bea.com/products/product.jsp?highlight=wla>
- Application Integration documentation

Introducing Application Integration

<http://edocs.bea.com/wli/docs81/aiover/index.html>

Using the Application Integration Design Console

<http://edocs.bea.com/wli/docs81/aiuser/index.html>

- BEA WebLogic Integration documentation
<http://edocs.bea.com/wli/docs81/index.html>
- BEA WebLogic Platform documentation
<http://edocs.bea.com/platform/docs81/index.html>
- Oracle Applications documentation
www.oracle.com

Contact Us!

Your feedback on the BEA WebLogic Adapter for Oracle Applications 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 BEA WebLogic Adapter for Oracle Applications documentation.

In your e-mail message, please indicate that you are using the documentation for BEA WebLogic Adapter for Oracle Applications and the version of the documentation.

If you have any questions about this version of BEA WebLogic Adapter for Oracle Applications, or if you have problems using the BEA WebLogic Adapter for Oracle Applications, 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 documentation 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.
<i>italics</i>	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. <i>Examples:</i> <pre>#include <iostream.h> void main () the pointer psz chmod u+w * \tux\data\ap .doc tux.doc BITMAP float</pre>
monospace boldface text	Identifies significant words in code. <i>Example:</i> <pre>void commit ()</pre>
<i>monospace italic text</i>	Identifies variables in code. <i>Example:</i> <pre>String <i>expr</i></pre>
UPPERCASE TEXT	Indicates device names, environment variables, and logical operators. <i>Examples:</i> <pre>LPT1 SIGNON OR</pre>
{ }	Indicates a set of choices in a syntax line. The braces themselves should never be typed.

Convention	Item
[]	<p>Indicates optional items in a syntax line. The brackets themselves should never be typed.</p> <p><i>Example:</i></p> <pre>buildobjclient [-v] [-o name] [-f file-list]... [-l file-list]...</pre>
	<p>Separates mutually exclusive choices in a syntax line. The symbol itself should never be typed.</p>
...	<p>Indicates one of the following in a command line:</p> <ul style="list-style-type: none"> • 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 <p>The ellipsis itself should never be typed.</p> <p><i>Example:</i></p> <pre>buildobjclient [-v] [-o name] [-f file-list]... [-l file-list]...</pre>
.	<p>Indicates the omission of items from a code example or from a syntax line. The vertical ellipsis itself should never be typed.</p>

Installing the BEA WebLogic Adapter for Oracle Applications

This section explains how to install the BEA WebLogic Adapter for Oracle Applications with WebLogic Integration on both Windows and UNIX systems.

This section is organized as follows:

- [Preparing to Install the Adapter](#)
- [Understanding the Representation of Paths](#)
- [Installing the Adapter](#)
- [Next Steps](#)

Preparing to Install the Adapter

Before you install the BEA WebLogic Adapter for Oracle Applications, be sure to complete the following tasks:

- [Review the Release Notes](#)
- [Install the Oracle Applications JDBC Driver](#)
- [Understanding the Representation of Paths](#)

Review the Release Notes

The *BEA WebLogic Adapter for Oracle Applications Release Notes* contain important information about the software you must install prior to installing the BEA WebLogic Adapter for Oracle Applications. Also, be sure to check the release notes for information about any required patches for your system. The *BEA WebLogic Adapter for Oracle Applications Release Notes* are available at the following URL:

<http://edocs.bea.com/wladapters/docs81/pdf/relnotes.pdf>

Install the Oracle Applications JDBC Driver

To install the Oracle Applications JDBC driver:

1. Find the file `classes12.zip` in your Oracle Applications distribution.
2. Copy this file to:
 - `BEA Application Explorer\lib` on a Windows system
 - `beabse/lib` on a UNIX system

Understanding the Representation of Paths

When you install WebLogic Integration, you specify the locations for files. Some of these files are required by the adapter. This document uses the following conventions to represent the locations of these files.

- `BEA_HOME` represents the BEA Home directory of your WebLogic installation. For example:
 - If you install the product in the default location on a Windows system, `BEA_HOME` represents `c:\bea`.

- If you install the product in the default location on a UNIX system, *BEA_HOME* represents `/bea`.
- *WLI_HOME* represents the root of your WebLogic Integration installation. For example:
 - If you install WebLogic Integration in the default location on a Windows system, *WLI_HOME* represents `c:\bea\weblogic81\integration`.
 - If you install WebLogic Integration in the default location on a UNIX system, *WLI_HOME* represents `/bea/weblogic81/integration`.
- *domain* is used to indicate the name of a domain.

You use the Configuration Wizard to create custom user domains. When you set up the domain configuration with the Configuration Wizard, you must specify a domain name, *domain*. You must also indicate where the directory associated with this domain is created. This directory contains files required for that domain. To learn more about the Configuration Wizard, see *Creating WebLogic Configurations Using the Configuration Wizard* which is available at the following URL:

<http://edocs.bea.com/platform/docs81/configwiz/index.html>

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

For example, if you use the Configuration Wizard to create a domain in the default location on a Windows system, *DOMAIN_HOME* represents `c:\bea\weblogic81\user_projects\domains\domain`.

If you use the Configuration Wizard to create a domain in the default location on a UNIX system, *DOMAIN_HOME* represents `/bea/weblogic81/user_projects/domains/domain`.

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.

Installing the Adapter

This section explains how to install the BEA WebLogic Adapter for Oracle Applications with WebLogic Integration. It includes the following steps:

- [Step 1. Obtain the BEA WebLogic Adapter for Oracle Applications](#)
- [Step 2. Configure the Domain](#)
- [Step 3. Extract the Adapter Files and Change the WebLogic Script](#)

- [Step 4. Update the BEA License](#)
- [Step 5. Deploy the Adapter](#)
- [Step 6. Create an Adapter Administrative User](#)

Step 1. Obtain the BEA WebLogic Adapter for Oracle Applications

The Oracle Applications adapter is packaged as a zip file. This zip file contains the EAR file for the adapter, as well as other files necessary to run the adapter. To obtain the file containing the BEA WebLogic Adapter for Oracle Applications software (BEA_ORACLEAPPS_8_1.zip), do one of the following:

- Download the file from the following URL:
http://commerce.bea.com/products/weblogicadapters/wl_adapter_home.jsp
- Obtain the software on CD.

Step 2. Configure the Domain

You must deploy the BEA WebLogic Adapter for Oracle Applications in a domain that supports application integration functionality. You can create one of your own, or use the sample integration domain. The sample integration domain is:

- On Windows: `BEA_HOME\weblogic81\samples\domains\integration`
- On UNIX: `BEA_HOME/weblogic81/samples/domains/integration`

If you have not already done so, use the Configuration Wizard to create the domain. Select the Integration domain template.

To learn more about the Configuration Wizard, see the *Creating WebLogic Configurations Using the Configuration Wizard* at the following URL:

<http://edocs.bea.com/platform/docs81/configwiz/index.html>

Step 3. Extract the Adapter Files and Change the WebLogic Script

This section explains how to extract the BEA WebLogic Adapter for Oracle Applications files and edit your WebLogic script to add files to the CLASSPATH and adjust the PATH.

Note: For WebLogic Integration 8.1 SP2, the script file you edit is `setDomainEnv.cmd` or `setDomainEnv.sh`. For versions of WebLogic Integration, earlier than 8.1 SP2, the file name is `startWeblogic.cmd` or `startWebLogic.sh`.

Set the classpath using the procedure appropriate for your system:

- [Extracting Files for Windows](#)
- [Extracting Jars and Adjusting the Classpath for UNIX](#)

Extracting Files for Windows

To extract the BEA WebLogic Adapter for Oracle Applications files:

1. Use WinZip (or another similar extracting product) to extract the contents of the `BEA_ORACLEAPPS_8_1.zip` file to a directory of your choice (for example, `BEA_HOME\adapters\oracleapps`).
2. Go to your Oracle RDBMS home directory, `ORACLE_HOME`. If this directory is *not* on the same machine as the one running the WebLogic Server, you must copy the Oracle AQ drivers to a directory on the machine running WebLogic Server. The driver file is named `aqapi.jar`. The Oracle installation places this file in `ORACLE_HOME\rdbms\jlib`. This file can reside anywhere on the machine running WebLogic Server, (for example, `BEA_HOME\adapters\oracleapps`).

Extracting Jars and Adjusting the Classpath for UNIX

To extract the BEA WebLogic Adapter for Oracle Applications jar file and edit the WebLogic script:

1. Use jar (or another similar extracting product) to extract the contents of `BEA_ORACLEAPPS_8_1.zip` to a directory of your choice (for example, `BEA_HOME/adapters/oracleapps`).
2. Go to your Oracle RDBMS home directory, `ORACLE_HOME`. If this directory is *not* on the same machine as the one running the WebLogic Server, you must copy the Oracle AQ drivers to a directory on the machine running WebLogic Server. The driver file is named `aqapi.jar`. The Oracle installation places this file in `ORACLE_HOME/rdbms/jlib`. This file can reside anywhere on the machine running WebLogic Server, (for example, `BEA_HOME/adapters/oracleapps`).
3. Find the `tools.jar` file in your JDK directory.

4. Go to the root directory for your domain:

```
cd DOMAIN_HOME
```

Note: You must choose a WebLogic Integration domain that supports application integration functionality.

5. Open the WebLogic script file with an editor. For WebLogic Integration 8.1 SP2, the file is `setDomainEnv.sh`. For earlier versions of WebLogic Integration, the file is `startWebLogic.sh`.
6. In the script file, find the variable `PRE_CLASSPATH`.
7. Add the path to the `tools.jar` file to the `PRE_CLASSPATH` variable.
8. Save your changes to the script file.

Step 4. Update the BEA License

In order to use the BEA WebLogic Adapter for Oracle Applications you must have a valid software license. If you have downloaded the adapter for evaluation, see the instructions on the adapter download page to obtain an evaluation license. If you have purchased a license for the adapter, you should receive the license file as an e-mail attachment. Once you have the license file for the adapter, you must update your `license.bea` file to include the new information for the adapter.

To update your `license.bea` file:

1. Save the adapter license file in the `BEA_HOME` directory. To avoid overwriting your `license.bea` file, use a name other than `license.bea`. For example, save the file as `oracleapps_adapter_license.bea`. The adapter license file is the `license_update_file` referred to in step 4 of this procedure.

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

2. Go to the `BEA_HOME` directory:
 - 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. Add the JDK to your `PATH` variable. If it is already included, skip to step 4.
 - On a Windows system:

```
set PATH=BEA_HOME\jdk141_03\bin;%PATH%
```


- On a UNIX system:

```
PATH=BEA_HOME/jdk141_03/bin:$PATH
export PATH
```

4. Merge the adapter license file into your existing license:

- 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 of the adapter license file you saved in step 1.

5. Save a backup copy of your updated `license.bea` file. This backup location should be a safe place that is neither the WebLogic Integration nor the application installation directories.

Step 5. Deploy the Adapter

After you have installed the BEA WebLogic Adapter for Oracle Applications, you must deploy it to your domain.

To deploy the adapter:

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

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

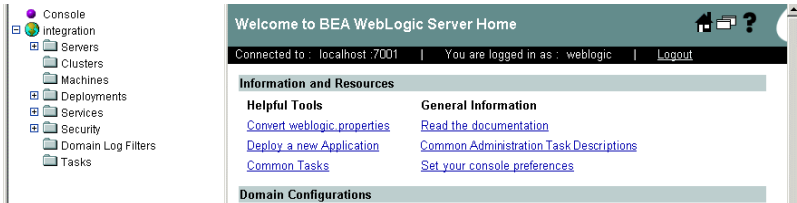
where

- *host* represents the machine on which WebLogic Server is running
- *port* represents the listening port.

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

3. Enter the user name and password for the server.

The WebLogic Server Administration Console appears.



4. In the left pane, expand the Deployments node.
5. Under the Deployments node, click the Applications node.

The Administration Console displays the Applications window.



An application is a J2EE application or Web Service contained in an Enterprise Application Archive (EAR) file or exploded EAR directory. Individual components contained in a J2EE application can be deployed to one or more target servers or clusters.

This Applications page displays key information about the EAR files or exploded EAR directories that have been configured for deployment in this WebLogic Server domain.

[Deploy a new Application](#)

[Customize this view...](#)

6. Click the Deploy a new Application link.

The Administration Console displays the Deploy an Application window.



7. Click the upload your files(s) link.

The Administration Console displays the Install or Update an Application window.

Upload and Install an Application or Module

Click the Browse... button below to locate an application or module file on the machine from which you are currently browsing. When you have located the file, click the Upload button to upload and install the application or module on this Administration Server. The following types of files may be uploaded and installed:

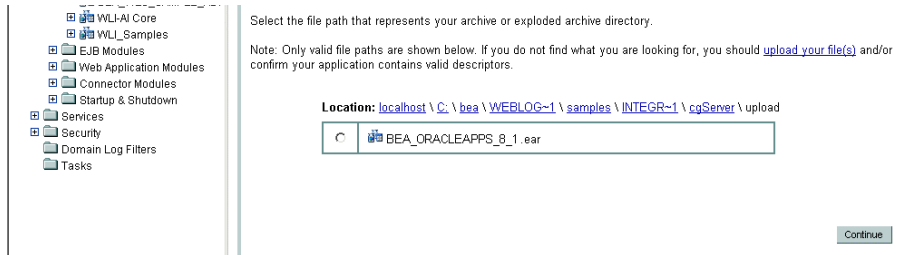
- A **.jar** containing EJBs (Enterprise JavaBeans)
- A **.war** (Web Application Archive) containing JSPs and Servlets
- A **.rar** (Resource Adapter Archive) containing a Connector module
- An **.ear** (J2EE Enterprise Application Archive) containing any of the above

Note: If you browse for the file, you may have to adjust the file-type filter to 'All' in order to find .jar, .war, .rar and .ear files.



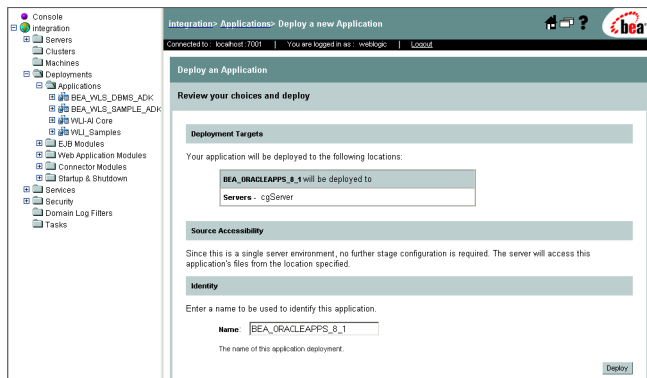
- Click Browse, navigate to the directory in which the `BEA_ORACLEAPPS_8_1.ear` file resides, and then click Upload.

The Administration Console prompts you to confirm the uploaded application.



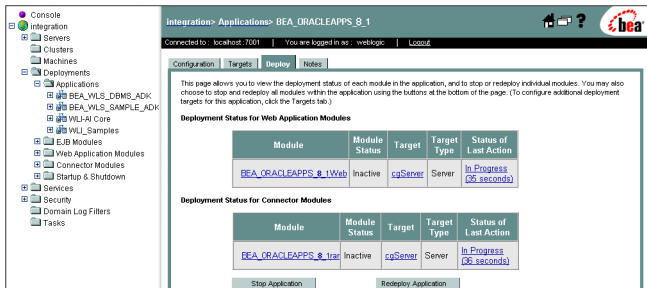
- Click the radio button next to the application file and then click Continue.

The Administration Console uploads the file and displays the Deploy an Application window (in which the default target server is specified).



- Click Deploy.

The Administration Console deploys the application and displays its deployment status.



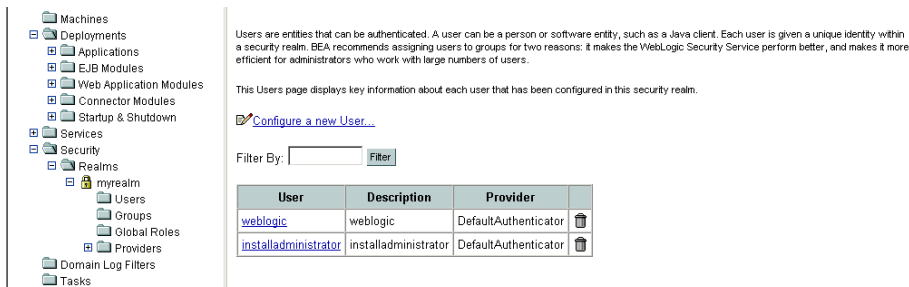
Step 6. Create an Adapter Administrative User

If you want to manage security for the BEA WebLogic Adapter for Oracle Applications, you can create an administrative user (such as `oracleappsAdapterAdmin`) who is authorized to log in to the Application View Console, create application views, configure services and events, deploy, and test. This adapter administrator also needs to be added to the `Administrators` group.

To create a new adapter administrative user:

1. In the left pane of the WebLogic Server Administration Console, click the Security node.
2. In the left pane, click the Realms node.
3. In the left pane, click the name of the domain for which you want to configure security.
4. In the left pane, click Users.

The Users page opens.



Users are entities that can be authenticated. A user can be a person or software entity, such as a Java client. Each user is given a unique identity within a security realm. BEA recommends assigning users to groups for two reasons: it makes the WebLogic Security Service perform better, and makes it more efficient for administrators who work with large numbers of users.

This Users page displays key information about each user that has been configured in this security realm.

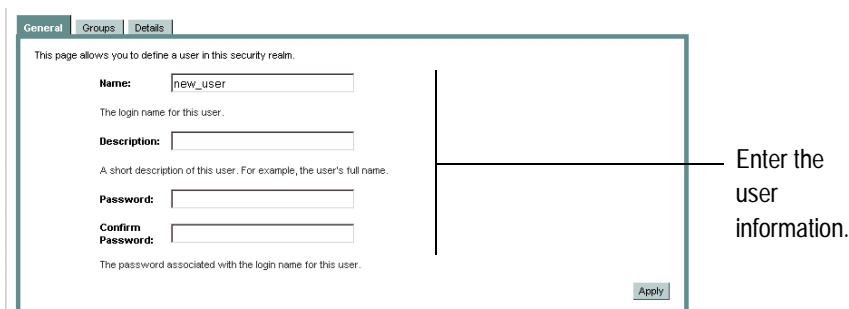
[Configure a new User...](#)

Filter By:

User	Description	Provider	
weblogic	weblogic	DefaultAuthenticator	<input type="button" value=""/>
installadministrator	installadministrator	DefaultAuthenticator	<input type="button" value=""/>

5. Click the Configure a New User link.

The Create User page opens.



This page allows you to define a user in this security realm.

Name:

The login name for this user.

Description:

A short description of this user. For example, the user's full name.

Password:

Confirm Password:

The password associated with the login name for this user.

Enter the user information.

6. Enter the user name, description and password, and then click Apply.

The User page opens.

[Configure a new User...](#)

General Groups Details

This page allows you to define a user in this security realm.

Name: MyAdapterAdmin
The login name for this user.

Description: MyAdapter Administrator
A short description of this user. For example, the user's full name.

Password: [Change...](#)

Apply

Enter a description for this user.

Click to change the password.

7. Click the Groups tab.

The Groups page opens.

[Configure a new User...](#)

General Groups Details

This page allows you to select the groups to which this user belongs.

Group Membership:

Possible Groups	Current Groups
Administrators	
Deployers	
Monitors	
Operators	

Apply

Select a group for the user.

8. In the Possible Groups list, select Administrators and then click the right arrow to add the Administrators group to the list of current groups.
9. Click Apply.
10. In the left pane, click the Users node and confirm that the user you created appears in the list of users.

Next Steps

After you have finished installing the BEA WebLogic Adapter for Oracle Applications, you can proceed to the following tasks:

- [Configuring the Concurrent Program Agent](#)
- [Installing the BEA Application Explorer](#)

- [Configuring the Adapter to Use Oracle Applications Concurrent Programs](#)
- [Starting Integration with Oracle Applications](#)

Installing the BEA Application Explorer

To proceed, you must install the BEA Application Explorer. If you do not already have it installed, do so now. To learn more about installing the BEA Application Explorer, see the *BEA Application Explorer Installation and Configuration Guide* at the following URL:

<http://edocs.bea.com/wlapters/docs81/index.html>

Configuring the Adapter to Use Oracle Applications Concurrent Programs

To learn more about configuring the adapter to use Oracle Applications concurrent Programs, see [Appendix A, “Configuring the Concurrent Program Agent.”](#)

Starting Integration with Oracle Applications

After you have successfully installed and deployed the BEA WebLogic Adapter for Oracle Applications and the BEA Application Explorer, you can begin integrating with your Oracle Applications system using the adapter and BEA WebLogic Integration. To learn more about integrating with Oracle Applications, see the *BEA WebLogic Adapter for Oracle Applications User Guide* at the following URL:

<http://edocs.bea.com/wlapters/docs81/index.html>

Configuring the Concurrent Program Agent

Concurrent programs perform several functions, such as transferring data from an interim table to a production table. The adapter supports this functionality with a concurrent programs agent. The concurrent agent submits the concurrent program to an Oracle concurrent manager, which controls Oracle Applications background processing.

In order to enable support for concurrent Oracle programs, you must install and configure the concurrent programs agent. After the concurrent programs agent is installed and configured, you can test it to make sure that it behaves as you expect.

This document includes the following topics:

- [About the Installation and Configuration Process](#)
- [Installing the Concurrent Programs Agent](#)
- [Configuring the Concurrent Programs Agent](#)
- [Testing the Concurrent Program Agent](#)

Note: You can also run a concurrent program using the Oracle Applications Web Client. See your Oracle documentation for more information about using the Web Client.

About the Installation and Configuration Process

The concurrent programs agent uses services to invoke concurrent programs in Oracle. Therefore, configuring the concurrent program agent is similar to configuring any other service with the adapter.

The steps to configure the concurrent programs agent are:

1. Install the concurrent program manager. To learn more, see “Installing the Concurrent Programs Agent” on page 3.
2. Generate the service request and response schemas. To learn more, see “Creating Schemas for the Concurrent Programs Agent” on page 5.

Note: Generate these schemas only once for an agent. These request and response schemas are sufficient for all concurrent programs run from a given Oracle Applications connection.

3. Add a service. To learn more, see “Adding a Service for the Concurrent Programs Agent” on page 5.

Note: Add a service only once for an agent. This service is sufficient for all concurrent programs run from one Oracle Applications connection.

4. Determine the request parameters. You need these parameters to create the request document that the service submits to the Oracle concurrent programs manager. To learn more, see “Determining the Request Parameters” on page 7.
5. Create the request document. To learn more, see “Creating the Request Document” on page 10.
6. Submit the request document. To learn more, see “Submitting the Request and Viewing the Response” on page 13.

These are the same steps you use to test the concurrent programs agent. To learn more about testing a concurrent programs agent, see [Testing the Concurrent Program Agent](#).

Installing the Concurrent Programs Agent

The installation process ensures that both your machine and your Oracle Applications system are prepared for the concurrent programs agent.

To install the concurrent programs agent:

- [Step 1: Run the Concurrent.ora Script](#)
- [Step 2: Adjust the System Path](#)
- [Step 3: Verify CONCSUB](#)

Step 1: Run the Concurrent.ora Script

The first step of the installation process is to run the `concurrent.ora` script.

To run the `concurrent.ora` script:

1. Log on to a machine that:
 - Has a TNSnames entry in the `tnsnames.ora` file that points to your Oracle Applications database instance.
 - Has SQL*Plus installed.
2. If you are on a Windows system, open a command prompt window.
3. Go to the directory to which you extracted `concurrent.ora`.
4. Issue the following command. You must use the APPS user ID:

```
SQLPLUS APPS/password@database @concurrent.ora
```

Here,

password is the password for the APPS Oracle user ID.

database is a TNSnames entry that points to your Oracle Applications database instance. This TNSnames entry must be in the `tnsnames.ora` file.

Step 2: Adjust the System Path

The next step is to make sure that the Oracle Applications foundation directory is in the system path.

To adjust the system path:

1. Log on to the server that runs your Oracle Applications instance.
2. Add the Oracle Applications foundation directory to the system path.
 - On Windows, add `%FND_TOP%\bin` to the system path.
 - On UNIX, add `$FND_TOP/bin` to the system path.

Here, `FND_TOP` is the Oracle Applications foundation directory. This value is set in the `vis.env` (or `prod.env`) file located under this directory:

- `oraclehome\visappl` on Windows
- `oraclehome/visappl` on UNIX

Step 3: Verify CONCSUB

Verify that the `CONCSUB` executable file is present on the server running your Oracle Applications instance.

This file must be in:

- `$FND_TOP/bin` on UNIX
- `%FND_TOP%\bin` on Windows

If this file is missing, contact your Oracle Applications administrator.

Configuring the Concurrent Programs Agent

Now that you have installed the concurrent programs agent, you must configure it .

This section contains the following topics:

- [Creating Schemas for the Concurrent Programs Agent](#)
- [Adding a Service for the Concurrent Programs Agent](#)
- [Testing the Concurrent Program Agent](#)

Creating Schemas for the Concurrent Programs Agent

For the concurrent programs agent, as for any other service, you must create a request schema and a response schema. For each Oracle Applications connection that you will use to run concurrent programs, you only need to create these schemas *once*. These schemas are sufficient for all the concurrent programs invoked through that connection.

To create the schemas:

1. Start the BEA Application Explorer and connect to an Oracle Applications database instance.
2. From the left pane of the BEA Application Explorer, select Specialty Agents. This displays the Concurrent Agent node.
3. Right-click Concurrent Agent and select Create Service Schema.

This creates the service schemas in the session path directory. In the right pane, the Request tab and Response tab appear.

You have finished creating the service schemas. To learn more about creating schemas, see *Using Application Integration* at the following URL:

<http://edocs.bea.com/wli/docs81/aiuser/index.html>

Adding a Service for the Concurrent Programs Agent

Adding a service for the concurrent programs agent is similar to configuring any other Oracle Applications service. For a concurrent program service, you do need to specify some particular configuration parameters in the Application View Console's Add Service window.

To add a service for the concurrent programs agent:

1. Log on to the application view console.

2. Create a new application view, or add this service to an existing application view. To learn more about creating and modifying application views, see “Defining an Application View” in [Introduction to Application Integration](#) in *Using Application Integration* at the following URL:

<http://e-docs.bea.com/wli/docs81/aiuser/1usrnr.html>

3. Set the parameters for the application view.

Select: Concurrent Request

Userid*	APPS
Password*
Host*	ORACLE11X
Port*	1523
ServiceId*	vis
TnsName*	vis

settings

Trace on/off	<input type="checkbox"/>
Verbose Trace on/off	<input type="checkbox"/>
Document Trace on/off	<input type="checkbox"/>

schema: Concurrent_Request

Use these parameters for a service for the concurrent programs agent:

Table 1. Service Properties for Concurrent Programs Agent

Parameter	Definition
Select	Concurrent Request
Userid	The user ID for the Oracle Applications database. This must be APPS.
Password	The password for the APPS Oracle user ID.
Host	The name of the server running your Oracle Applications instance.
Port	The number of the port on which the database is listening.
ServiceId	The unique name that identifies the database. Contact your Oracle Applications DBA for this information.
TnsName	The logical name of the Oracle Applications database instance. Contact your Oracle Applications DBA for this information.

Testing the Concurrent Program Agent

Once the concurrent program agent is installed and configured, you can test it to make sure that you have correctly configured the service associated with the agent.

The steps to test the concurrent program agent are:

1. Determine the parameters you need to create the XML request document. To learn more, see “Determining the Request Parameters” on page 7.
2. Create the request document. To learn more, see “Creating the Request Document” on page 10.
3. Submit the request document. To learn more, see “Submitting the Request and Viewing the Response” on page 13.
4. View the response document. To learn more, see “Submitting the Request and Viewing the Response” on page 13.

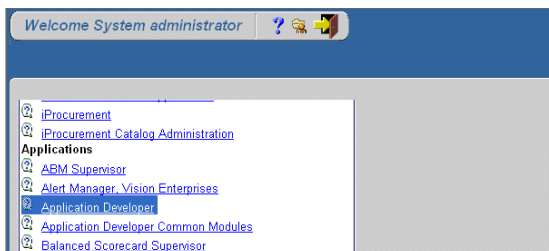
Determining the Request Parameters

In order to submit a request to a concurrent program, you must include the parameters the program requires. You can use the Oracle Applications Web Client to determine what parameters

To determine the parameters:

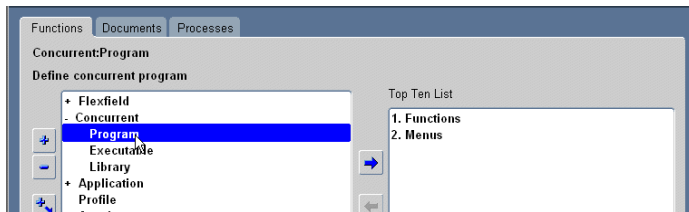
1. Start the Oracle Applications Web Client. Log on to the Oracle Applications instance on which you wish to run the concurrent program.
2. From the client’s main window, select Application Developer.

If you do not see the Application Developer option, contact your system administrator to make sure the account ID you used to log on to the system is enabled.



The Application Developer window appears.

3. Expand the Concurrent node and double-click Program.



The Concurrent Programs window appears.

The screenshot shows the Oracle Concurrent Programs form. The form is divided into several sections: 'Program' (with fields for Program, Short Name, Application, and Description), 'Executable' (with fields for Name and Method), 'Request' (with fields for Type, Incrementor, and MLS Function, and checkboxes for 'Use In SRS', 'Run Alone', 'Enable Trace', 'Allow Disabled Values', 'Restart on System Failure', and 'NLS Compliant'), and 'Output' (with a 'Format' dropdown set to 'Text', checkboxes for 'Save (C)' and 'Print', and fields for 'Columns', 'Rows', 'Style', and 'Printer'). At the bottom, there are buttons for 'Copy to...', 'Session Control', 'Incompatibilities', and 'Parameters'.

4. From the Web Client menu bar, select View→Query By Example→Enter.

To show that the form is in query mode, the fields in the Web Client's Concurrent Programs window turn blue.

5. Search for the record that contains the name or description of the concurrent program you wish to run.

- To search on the name, enter the concurrent program's short name in the Short Name field. For example, if your system has a concurrent program named WICMLP, enter that in the Short Name field.

The names of the concurrent programs available for your use depend on your Oracle Applications installation.

- To search on the description, enter the description in the description field.

This example shows a search for the WICMLP for Work In Process (WIP) Mass Load concurrent program.

The screenshot shows the configuration for the WICMLP program. The fields are as follows:

- Program: Enabled
- Short Name: **WICMLP**
- Application:
- Description:
- Executable**
 - Name:
 - Options:
 - Method:
- Request**
 - Type:
 - Incrementor:
 - MLS Function:
 - Use In SRS
 - Run Alone
 - Enable Trace
 - Allow Disabled Values
 - Restart on System Failure
 - NLS Compliant
- Output**
 - Format:
 - Save (C)
 - Print
 - Columns:
 - Rows:
 - Style:
 - Style Required
 - Printer:

Buttons at the bottom: Copy to..., Session Control, Incmpatibilities, Parameters

6. Start the search by selecting View→Query By Example→Run.

The record appears.

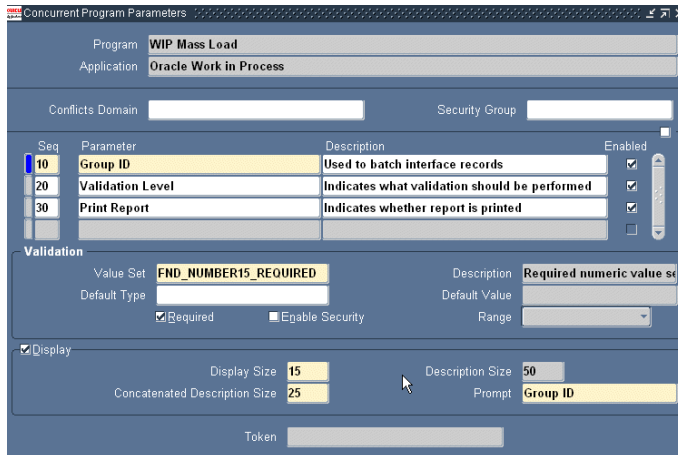
The screenshot shows the search results for the WIP Mass Load program. The fields are as follows:

- Program: **WIP Mass Load** Enabled
- Short Name: **WICMLP**
- Application: **Oracle Work in Process**
- Description: **WIP Mass Load**
- Executable**
 - Name: **WICMLX**
 - Options:
 - Method: **Spawned**
- Request**
 - Type:
 - Incrementor:
 - MLS Function:
 - Use In SRS
 - Run Alone
 - Enable Trace
 - Allow Disabled Values
 - Restart on System Failure
 - NLS Compliant
- Output**
 - Format: **Text**
 - Save (C)
 - Print
 - Columns: **132**
 - Rows: **45**
 - Style: **Landscape**
 - Style Required
 - Printer:

Buttons at the bottom: Copy to..., Session Control, Incmpatibilities, Parameters

7. At the bottom of the Concurrent Programs window, click Parameters .

The window displays the parameters for this concurrent program.



8. View a parameter's details by clicking to the left of its sequence number.
9. Make a note of the parameter names and their sequence. For example, in the previous figure, Group ID is first, Validation Level is second, and Print Report is third. You will use this information when you create the request document.
10. Close the Oracle Applications Web Client.

You have finished determining the request document parameters. The next step is to create the request document, as explained in “Creating the Request Document” on page 10.

Creating the Request Document

You must create the XML request document for the concurrent program. The adapter will send this request document to the Oracle concurrent program manager to invoke the concurrent program.

The syntax of the XML request document the concurrent program agent uses to invoke a concurrent program is as follows:

Listing 1 Concurrent Program Request Document Syntax

```
<submit_request>
  <resp_appl_shrtnm/>
  <responsibility/>
```



```

<username/>
<wait/>
<prog_appl_shrtnm/>
<program/>
<parm/>
<parm/>
<parm/>
. . .
<parm/>
</submit_request>

```

Here,

- `resp_appl_shrtnm` is the application short name of your responsibility.

Valid values for this field depend on your Oracle Applications installation. Generally you can find valid values for this field in the `APPLICATION_SHORT_NAME` column in the `FND_APPLICATION` table.

The default is `SYSADMIN`.

- `responsibility` is the responsibility of the application user. Note the spelling of this tag name.

The default is `System Administrator`. Valid values for this field depend on your Oracle Applications installation. Generally you can find valid values for this field in the `RESPONSIBILITY_NAME` column in the `FND_RESPONSIBILITY` and `FND_RESPONSIBILITY_TL` tables.

Note: The `responsibility` and `resp_appl_shrtnm` need not be the same, but they must match. In this example, the `resp_appl_shrtnm` is `SYSADMIN` and the `responsibility` is `System Administrator`.

- `username` is the name of your Application Object Library user.

This name will be used to update WHO information for data that the concurrent program changes and to create the report output file name for the specific request. This parameter is different from the user ID used to log on to the Oracle database.

Valid values for this field depend on your Oracle Applications installation. Generally you can find valid values for this field in the `USER_ID` column in the `FND_USER` table.

The default is `SYSADMIN`.

- `wait` is a boolean indicating whether the request is synchronous or asynchronous.

This must be Y or N:

- Y indicates that the request is synchronous and that the agent will check the status of the request will be checked every 60 seconds.
- N indicates that the request is asynchronous and therefore the status will not be checked. This is the default.
- `prog_appl_shrtnm` is the application short name of your program.

This value cannot be null.

- `program` is the name of the concurrent program.

Valid values for this field depend on your Oracle Applications installation. Generally you can find valid values for this field in the `CONCURRENT_PROGRAM_NAME` column in the `FND_CONCURRENT_PROGRAMS` and `FND_CONCURRENT_PROGRAMS_TL` tables.

This value cannot be null.

Note: The `program` and `prog_appl_shrtnm` need not be the same, but they must match. In this example, the `prog_appl_shrtnm` is WIP and the `program` is WICMLP.

- `parm` is a concurrent program parameter. There are as many of these as the application requires. The parameters are distinguished by their place in the sequence.

To learn more about determining the parameters for your concurrent program and their sequence sequence, see “Determining the Request Parameters” on page 7.

The following listing is a request for running the WIP Mass Load example concurrent program.

Listing A-1 Request Document for WIP Mass Load Concurrent Program

```
<submit_request>
  <resp_appl_shrtnm>SYSADMIN</resp_appl_shrtnm>
  <responsibility>System Administrator</responsibility>
  <username>SYSADMIN</username>
  <wait>Y</wait>
  <prog_appl_shrtnm>WIP</prog_appl_shrtnm>
  <program>WICMLP</program>
  <parm>1211</parm>
  <parm>0</parm>
  <parm>1</parm>
</submit_request>
```

In this listing, the parameters are the same as those shown in the Oracle Applications Web Client's Concurrent Program Parameters window in "Determining the Request Parameters" on page 7:

- The first parameter is Group ID.
- The second parameter is Validation Level.
- The third parameter is Print Report.

For a complete list of Oracle Concurrent program parameters, see your Oracle documentation.

Submitting the Request and Viewing the Response

To submit a request document to a concurrent programs agent and view the response:

1. Create the schemas for the concurrent programs agent. To learn more, see "Creating Schemas for the Concurrent Programs Agent" on page 5.
2. Add a service for the concurrent programs agent. To learn more, see "Adding a Service for the Concurrent Programs Agent" on page 5.
3. Start the Application View Console, and open the Application View Summary window.
4. In the Services area, locate the service and click Test.
The Test Service page opens.
5. Enter the XML that invokes the service request. To learn more about constructing the XML request document, see "Determining the Request Parameters" on page 7 and "Creating the Request Document" on page 10.
6. Click Test to test the service.
The Results window opens, displaying the request and response documents.
7. Log on to the Oracle Applications instance through the Oracle Applications Web Client.
8. From the main drop-down menu for the desired module, select View→Requests.

The Find Request window appears.

9. If you wish, you can enter selection criteria (for example, Request ID).

10. Click Find.

The Requests window appears.

Request ID	Name	Parent	Phase	Status	Parameters
1459311	WIP Mass Load		Completed	Warning	9999, 0, 1
1434160	WIP Move Transaction M		Completed	Normal	
1431715	Deactivate Concurrent M		Completed	Normal	(None)
1431693	WIP Mass Load		Completed	Warning	3, 0, 1
1423944	Deactivate Concurrent M		Completed	Normal	(None)
1407866	Generate All Business Vi		Completed	Cancelled	T,
1404753	Security List Maintenan		Completed	Normal	.
1395526	Flexfield PL/SQL Pre-Prc		Completed	Normal	ARTOKEN, ARFXVIEW, ALL
1395525	Flexfield PL/SQL Pre-Prc		Completed	Normal	ARTOKEN, ARPLXSTX, ALL
1395524	Flexfield PL/SQL Pre-Prc		Completed	Normal	ARTOKEN, ARPLXLOC, ALL

The new request is displayed at the top of the list.

Depending on the concurrent program that was executed, one more of the buttons at the bottom of the window may be enabled. You can click one to view detail information about the concurrent program.

You have finished testing the concurrent program agent.

Index

A

- adapter software, obtaining 1-4
- administrative user, creating 1-10

B

- BEA Application Explorer, installing 1-12
- BEA_HOME* variable 1-2
- before you install 1-2

C

- classpath, updating 1-4
- customer support contact information vii

D

- domain* name 1-2
- domains, configuring 1-4

E

- extracting JARs 1-4

I

- installing
 - before you install 1-2
 - configuring the domain 1-4
 - creating an adapter administrative user 1-10
 - deploying the adapter 1-7
 - extracting JARs 1-4
 - obtaining the adapter software 1-4
 - steps, summary of 1-3

- updating the BEA license 1-6
- updating the classpath 1-4

L

- license, updating 1-6

P

- paths, representation of 1-2
- product support vii

R

- related information vi
- Release Notes*, reviewing 1-2

S

- security, configuring 1-10
- support vii

T

- technical support vii

U

- updating the BEA license 1-6

W

- WebLogic Server Administration Console 1-7
- WLI_HOME* variable 1-2