



AquaLogic BPM Enterprise Configuration Guide

Version: 6.0

Contents

| | |
|---|-----------|
| Introduction..... | 5 |
| Document Scope and Audience..... | 5 |
| BEA Documentation and Resources..... | 5 |
| Getting Started..... | 7 |
| Installation Prerequisites..... | 7 |
| Required Usernames and Passwords..... | 7 |
| Configuration Overview..... | 8 |
| AquaLogic BPM Enterprise Overview..... | 9 |
| ALBPM Enterprise Components..... | 9 |
| Persona..... | 9 |
| ALBM Core Components..... | 10 |
| ALBPM Supporting Applications..... | 10 |
| ALBPM Architectural Overview..... | 10 |
| ALBPM Standalone Architecture..... | 10 |
| JEE Single Node Architecture..... | 11 |
| JEE Cluster Architecture..... | 12 |
| WebLogic Portal Architecture..... | 12 |
| ALI Architecture..... | 13 |
| Configuring ALBPM Enterprise Standalone..... | 14 |
| Creating a New Directory Service..... | 14 |
| Deploying and Publishing a New ALBPM Project..... | 15 |
| Configuring ALBPM Enterprise on J2EE..... | 17 |
| Creating a New Directory Service..... | 17 |
| Configuring WebLogic Server..... | 19 |
| Creating a WLS Domain..... | 19 |
| Creating ALBPM Deployer User..... | 19 |
| Installing the ALBPM Deployer..... | 20 |
| Creating JDBC Data Sources on WLS..... | 20 |
| Creating a JMS Server..... | 21 |
| Creating a JMS Module..... | 22 |
| Creating JMS Resources..... | 22 |
| Disabling Server Affinity for Connection Factories..... | 23 |
| Configuring WebSphere Application Server..... | 24 |
| Installing ALBPM Deployer..... | 24 |
| Adding Authentication Data for JDBC..... | 24 |
| Creating a New JDBC Provider..... | 25 |
| Creating a JDBC Datasource for Directory Database..... | 26 |
| Creating a JDBC Datasource for the Engine Database..... | 27 |
| Creating a WebSphere JMS Bus..... | 28 |
| Creating JMS Resources..... | 29 |

| | |
|--|-----------|
| Configuring a Work Manager..... | 31 |
| Configuring JVM Properties..... | 31 |
| Restarting WebSphere Servers..... | 32 |
| Setting ALBPM properties for WebSphere..... | 32 |
| Preparing ALBPM Applications..... | 33 |
| Configuring the Deployer and Deployment Targets..... | 33 |
| Enabling Clustering..... | 33 |
| Building and Deploying Application EAR Files..... | 34 |
| Deploying and Publishing a New ALBPM Project..... | 34 |
| Integrating ALBPM with Other BEA Products..... | 36 |
| Integrating BEA AquaLogic Service Bus..... | 36 |
| Registering an ALBPM Process on AquaLogic Service Bus..... | 36 |
| Publishing Projects with Catalogued ALSB Resources..... | 37 |
| Supported Transport Protocols..... | 37 |
| Integrating AquaLogic BPM with WebLogic Portal..... | 38 |
| Configuring ALBPM Domains..... | 38 |
| Configure Single Sign On (SSO)..... | 39 |
| Creating the ALBPM Application Files..... | 40 |
| Deploying the WAR File in WebLogic Portal..... | 40 |
| Updating the Portal Web Project..... | 41 |
| Disabling JSP Validation..... | 42 |
| Refreshing the Web Project..... | 42 |
| Creating ALBPM Portlets..... | 42 |
| Creating Login Portlet..... | 43 |
| Verifying your Installation..... | 44 |
| Integrating AquaLogic BPM with AquaLogic Interaction..... | 44 |
| Running the WorkSpace Extension Configuration Wizard..... | 45 |
| Configuring the ALI Identity Service Database..... | 45 |
| Configuring the AquaLogic Identity Service..... | 46 |
| Deploying the .pte File..... | 47 |
| Assigning Process Administrators..... | 47 |
| Restart ALI Services..... | 47 |
| Verifying Your Installation..... | 47 |
| Configuring ALBPM Supporting Applications..... | 49 |
| Process Instance Archiving..... | 49 |
| Archiving and Deployment..... | 49 |
| Configuring Archiving..... | 49 |
| BAM and Process Data Mart..... | 50 |
| BAM Overview..... | 50 |
| Configuring BAM..... | 55 |
| Process Data Mart Overview..... | 56 |
| Configuring Process Data Mart..... | 56 |
| Reference..... | 57 |
| Engine and Directory Database Connectivity..... | 57 |

- BEA Oracle Driver Properties.....57
- BEA DB2 Driver Properties.....58
- BEA SQL Server Driver Properties.....59
- ALBPM Configuration Wizard Reference.....60
 - What is the ALBPM Configuration Wizard?.....60
 - Running the ALBPM Configuration Wizard.....60
 - ALBPM Configuration Wizard Reference.....61
- Example web.xml for WebLogic Portal.....71

Introduction

This section provides general information about the *AquaLogic BPM Configuration Guide*. This guide assumes that you have already installed ALBPM Enterprise. See the *AquaLogic BPM Installation Guide* for more information.

Document Scope and Audience

This document is written for system administrators who need to configure AquaLogic BPM Enterprise. It focuses primarily on the tasks required to configure various components of ALBPM Enterprise. It provides procedures for configuring the standalone and JEE versions. It also provides procedures for integrating AquaLogic BPM with other BEA products, including AquaLogic Interaction and WebLogic Portal.

This guide also provides general information about AquaLogic BPM Enterprise, including general product and architectural information.

This document assumes that you are familiar with system and enterprise application administration. If you are configuring ALBPM Enterprise on JEE, it assumes that you are familiar with web application server technology and have an understanding of how to install, configure, and administer your web application server.

BEA Documentation and Resources

This section describes other documentation, resources, support, and training information provided by BEA.

The table below lists a number of BEA Documentation and Resources which will help you get started with AquaLogic BPM.

| Resource | Description |
|------------------------------|---|
| ALBPM Product Documentation | The complete ALBPM 6.1 product documentation is available at http://edocs.bea.com/albsi/docs61/index.html . |
| Release Notes | The Release Notes file includes information about new features and known issues in the release. It is available at http://edocs.bea.com/albsi/docs61/relnotes/rel_notes_AquaLogic_BPM_6_1.htm . |
| Online Help | <p>The online help is written for all levels of users. It describes the AquaLogic BPM user interface and gives detailed instructions for completing tasks in AquaLogic BPM. To access online help:</p> <ul style="list-style-type: none"> In Studio, select Help ► Help Contents to access the complete ALBPM Studio help. Context help is also available by pressing the F1 key, or by selecting Help ► Dynamic Help from the menu. In WorkSpace, click on Help in the title bar, or click on the help icon (🔍) in the title bar of any panel for help about that panel. |
| AquaLogic BPM Support Center | <p>The AquaLogic BPM Support Center is a comprehensive repository for technical information on AquaLogic BPM products. From the Support Center, you can access products and documentation, search knowledge base articles, read the latest news and information, participate in a support community, get training, and find tools to meet most of your AquaLogic BPM related needs.</p> <p>http://one.bea.com/support/</p> |

| Resource | Description |
|-----------------------------------|--|
| ALBPM Developer Center at dev2dev | <p>The BEA AquaLogic BPM Suite Developer Center features articles, blogs, and newsgroups which will help you make the most out of ALBPM.</p> <p>http://dev2dev.bea.com/albpm/</p> |
| ALBPM Product Center | <p>Download products, read Release Notes, access recent product documentation, and view interoperability information.</p> <p>http://commerce.bea.com/products/aqualogic/bpm/albpm.jsp</p> |
| BEA Education | <p>Find information about available training courses, purchase training credits, and register for upcoming classes.</p> <p>http://dev2dev.bea.com/community/education/</p> |
| User Groups | <p>Visit the User Group section to collaborate with peers and view upcoming meetings.</p> <p>http://dev2dev.bea.com/community/usergroups/</p> |
| Technical Support | <p>If you cannot resolve an issue using the above resources, BEA Technical Support is happy to assist. Our staff is available 24 hours a day, 7 days a week to handle all your technical support needs.</p> <p>E-mail: ALBPMsupport@bea.com</p> <p>Phone Numbers:</p> <p>U.S. and Canada: +1 415 263 1696 or +1 866 262 7586</p> <p>EMEA (Europe, Middle East, and Africa): +44 1494 559127</p> <p>Australia / New Zealand: +61 2 9923 4030</p> <p>Asia Pacific: +61 2 9931 7822</p> <p>Singapore: +1 800 1181 202</p> |

Getting Started

The following topics provide general information that you may need to know before configuring AquaLogic BPM Enterprise. These sections include a general overview of the ALBPM Enterprise architecture and a list of pre-installation requirements you must perform before configuration.

Installation Prerequisites

Before configuring AquaLogic BPM Enterprise, you should ensure that you have completed the configuration prerequisites described below.

| Prerequisite | Description |
|----------------------------------|--|
| Install AquaLogic BPM Enterprise | Before performing the configuration and integration procedures, ensure that you have installed AquaLogic BPM as outlined in the <i>ALBPM Installation Guide</i> . |
| Install your database server | Before configuring AquaLogic BPM, ensure that you have installed your database server. A basic AquaLogic BPM Enterprise installation requires an external database for the engine and directory service databases. If you are configuring the BAM and Process Data Mart applications, each of these require a database. See your database vendor documentation for more information. |
| Install your application server | If you are configuring AquaLogic BPM on J2EE, ensure that you have installed and configured your application server. See your application server vendor's documentation for more information. |
| Install other BEA Products | If you are integrating AquaLogic BPM with other BEA products, ensure that these products are installed and configured. |

Required Usernames and Passwords

When configuring ALBPM Enterprise, there are multiple usernames and passwords that you must configure or use. These are used to connect to the various systems ALBPM uses.

| Username and Password | Description |
|----------------------------------|---|
| Engine Database Administrator | The DBA username and password to connect to the engine database. If you do not have DBA access to the database server, you can generate the SQL commands that can be given to your DBA. |
| Directory Database Administrator | The DBA username and password to connect to the directory database. If you do not have DBA access to the database server, you can generate the SQL commands that can be given to your DBA. |
| ALBPM Administrator | The username and password of the ALBPM Administrator. You must supply this username and password when you create the directory service for your ALBPM installation. This username and password is used to access Process Administrator. |
| WebLogic Server Administrator | The username and password of the WebLogic Server administrator. This username and password is required if you are installing ALBPM on WebLogic Server. |

| Username and Password | Description |
|--|---|
| WebSphere Application Server Administrator | The username and password of the WebSphere administrator. This username and password is required if you are installing ALBPM on WebSphere Application Server. |

Configuration Overview

This guide provides procedures and scenarios for configuring both the standalone and JEE versions of ALBPM Enterprise. It also provides information on integrating ALBPM with other BEA products. The following sections outline the different configuration scenarios provided in this guide.



Note: The following table lists the appropriate section of this guide for different configuration and integration tasks.

| Task | Description |
|---|--|
| Configuring AquaLogic BPM Enterprise Standalone | <i>Configuring ALBPM Enterprise Standalone</i> on page 14 |
| Integrating AquaLogic BPM with WebLogic Portal | <i>Integrating AquaLogic BPM with WebLogic Portal</i> on page 38 |
| Integrating AquaLogic BPM with AquaLogic User Interaction | <i>Integrating AquaLogic BPM with AquaLogic Interaction</i> on page 44 |

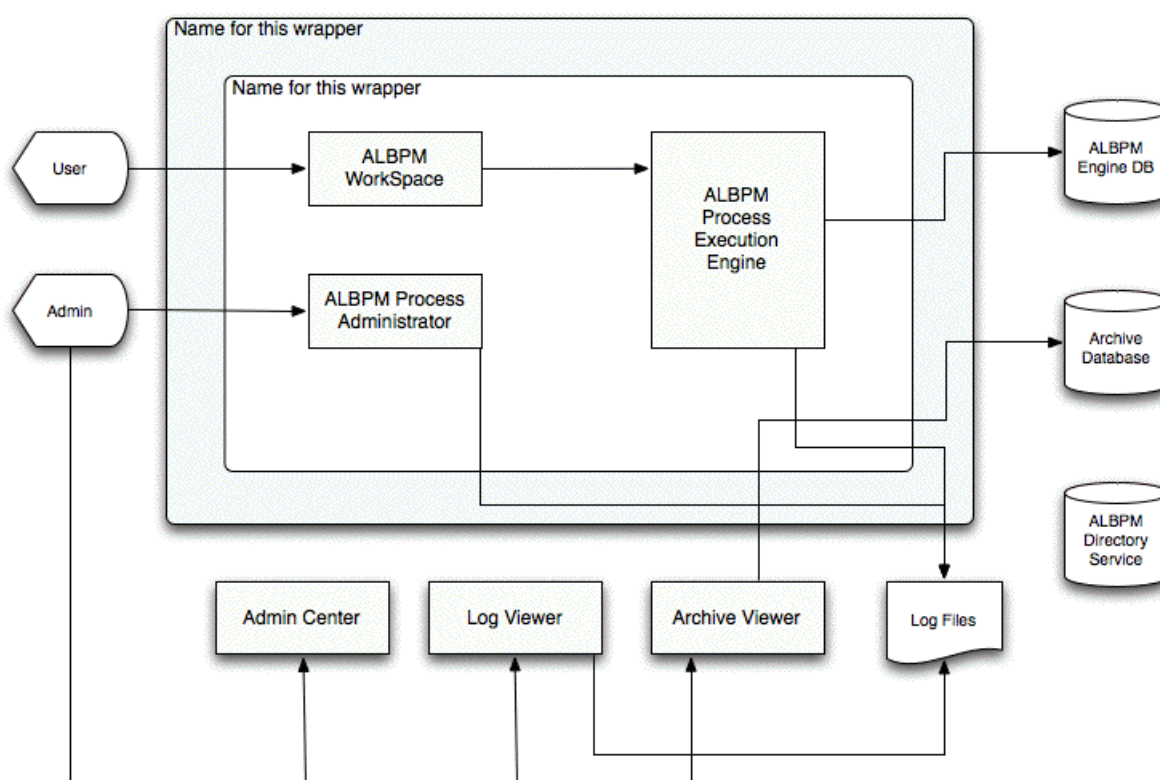
AquaLogic BPM Enterprise Overview

ALBPM Enterprise Components

The AquaLogic BPM Enterprise provides a production runtime environment.

As part of this environment ALBPM Enterprise provides the following:

- Standalone or JEE-based version
- Integration with external relational databases
- Servlet container for the user interface
- Web-based Administration Console
- Standalone administration applications for configuration, viewing logs, etc.



Persona

Persona defines the roles and responsibilities of users of AquaLogic BPM Enterprise.

ALBPM Enterprise has two distinct persona:

- **User:** Defines the end-user who is responsible for using the business processes created and managed with ALBPM.
- **Admin:** Defines the person who is responsible for installing, configuring, and maintaining ALBPM Enterprise. This person is also responsible for configuring and maintaining the Engine and Directory databases and LDAP servers. Within an enterprise, this responsibility may be spread across multiple roles and IT groups.

ALBM Core Components

The following core components are the basic components of any ALBPM installation.

Process Execution Engine

A Process Execution Engine provides a runtime environment for process instances. After you design your business process and export it to an ALBPM project, the Process Execution Engine runs this process and allows end-users to interact with it.

A Process Execution Engine coordinates interaction between process instances and their resources, including:

-

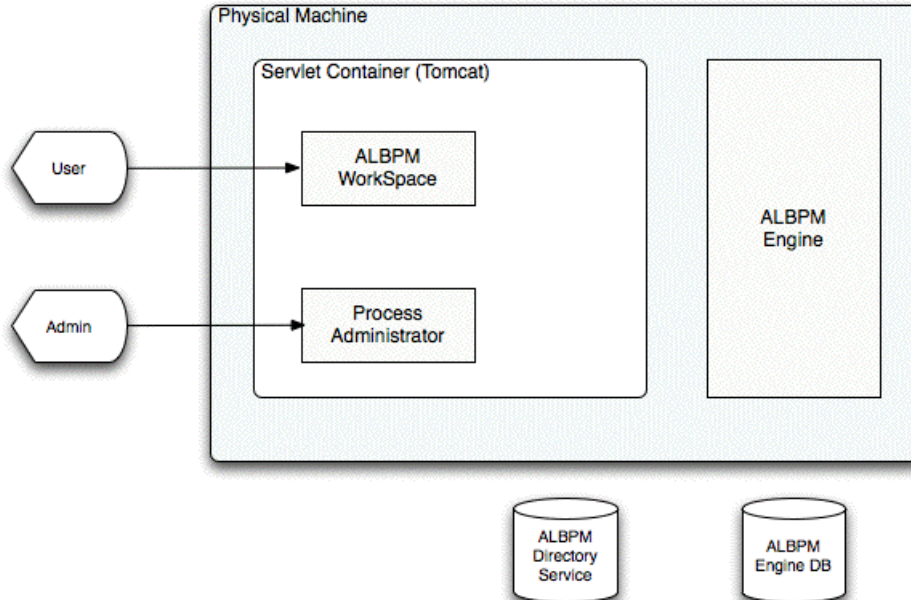
ALBPM Supporting Applications

Log Viewer Application

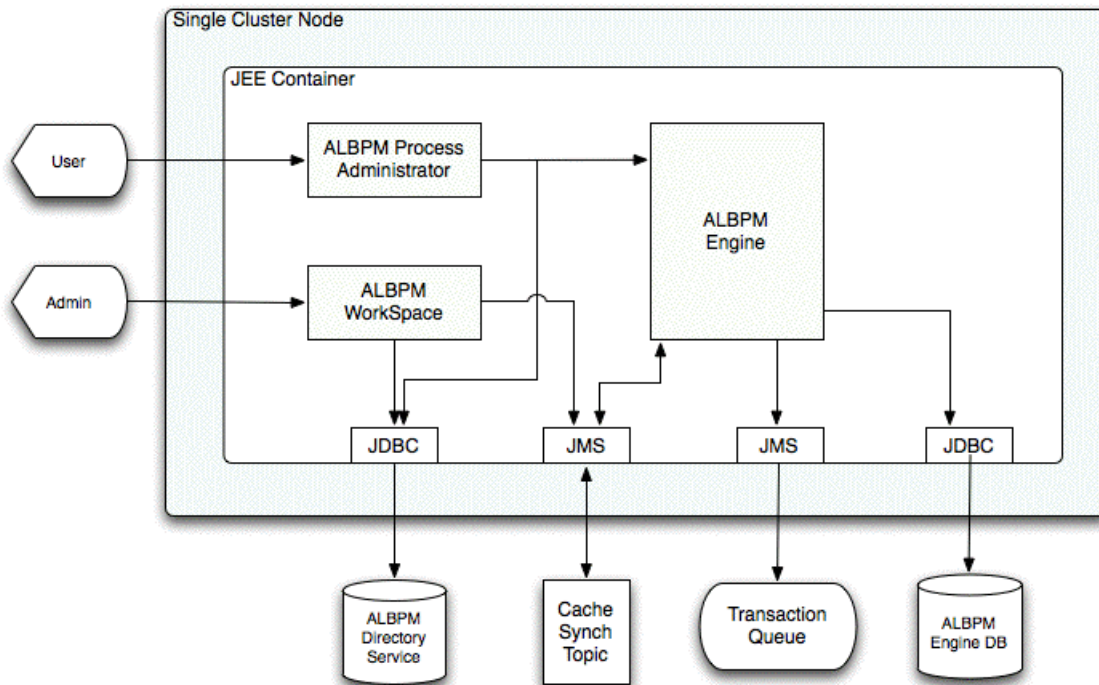
The Log Viewer is a Java-base utility for viewing the Process Execution Engine log files. It retrieves log records. It loads logs from the default log location.

ALBPM Architectural Overview

ALBPM Standalone Architecture



JEE Single Node Architecture



JMS Queues and Topics

The JEE version of ALBPM Enterprise requires you to configure a JMS queue and topic.

JMS Topic

Used to cache process instances. To render the inbox. Issue of keeping cache up-to-date. Used to tell client about internal changes.

JMS Queue

ALBPM uses a JMS queue to schedule transaction on the Engine Database.

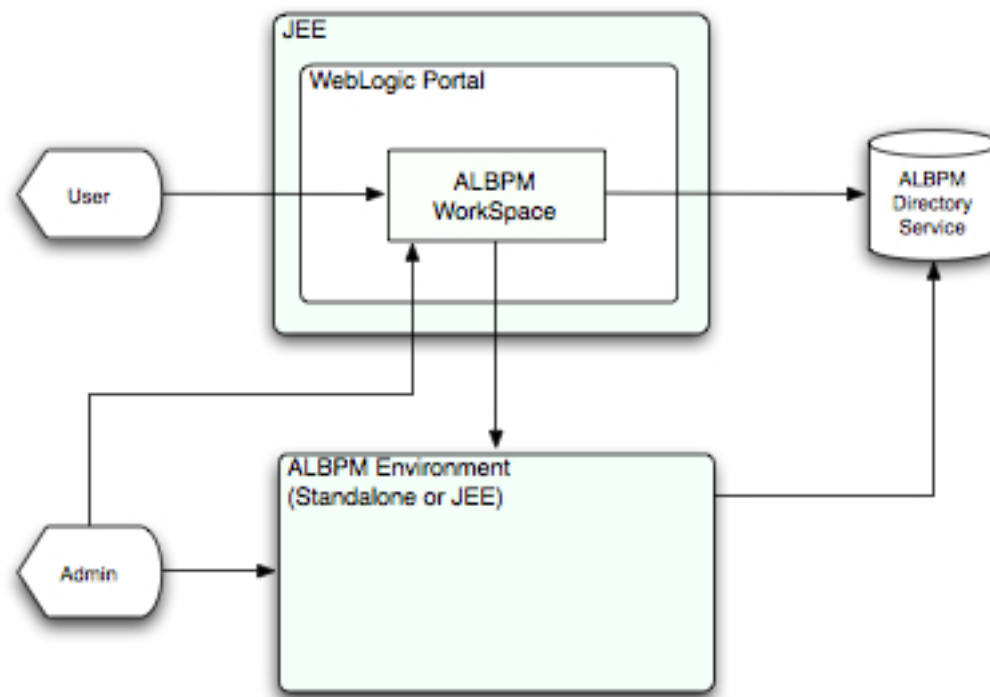
1. Schedules Asynchronous Transactions
2. Spreads Loads Across Multiple Nodes



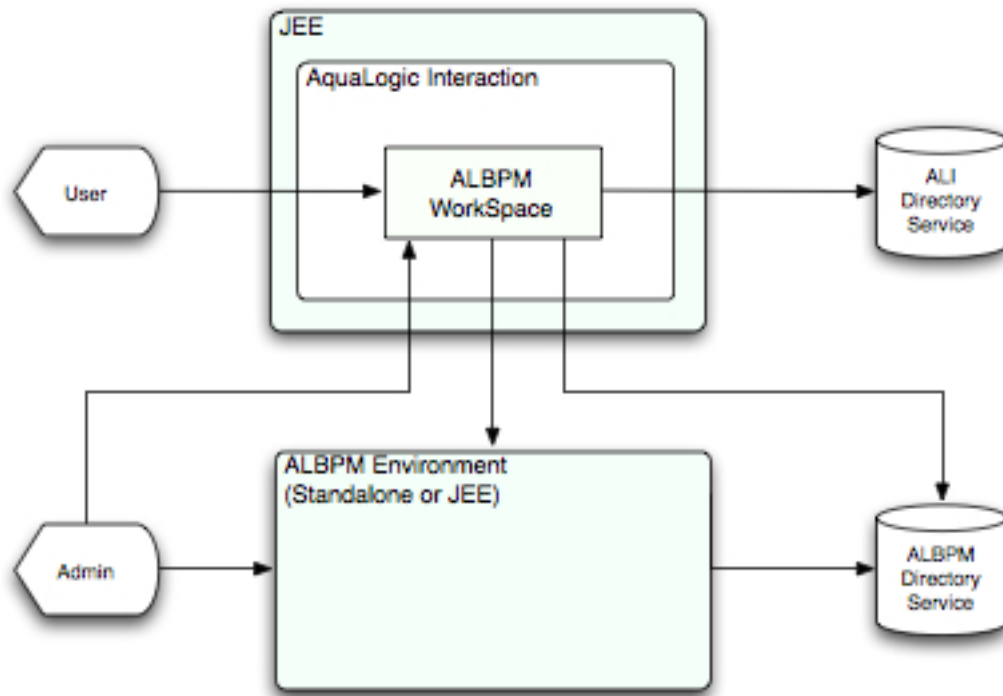
Note: The ALBPM JMS queue is not used to integrate with other applications. It is only used internally by the Process Execution Engine. External applications can not use this queue.

JEE Cluster Architecture

WebLogic Portal Architecture



ALI Architecture



Configuring ALBPM Enterprise Standalone

The following sections describe how to configure ALBPM Enterprise Standalone.

Before you begin configuring ALBPM Enterprise, ensure that you have reviewed [Configuration Overview](#) on page 8 and [Installation Prerequisites](#) on page 7 to determine what needs to be done before you begin and the order you need to perform the following procedures.

1. Review Architectural Information

See [ALBPM Standalone Architecture](#) on page 10. It is important to review this section if you are new to ALBPM.

2. Review Configuration Prerequisites

See [Installation Prerequisites](#) on page 7. This topic provides information about the required software you must install before using this guide.

3. Create an Organization

If you are configuring AquaLogic BPM in a production environment, you should ensure that you have created and configured your organizational structure before deploying and publishing a project. However, if you are creating a test environment, this is not required.

See the *ALBPM Enterprise Administration Guide* for more information on creating an organization.

4. Configure a Directory Service

The directory service is what allows you to connect projects deployed on ALBPM Enterprise with your organization.

5. Deploy a Project

After creating a directory structure, you are ready to deploy a project.

6. Test Your Configuration

To test your configuration you should login to the WorkSpace application as an ALBPM Administrator or using the login and password of a participant in your process.

Creating a New Directory Service

If you are planning to use a database-only directory service, you need to install and configure only a relational database. If you are planning to use a hybrid directory service, you need to install and configure both a relational database and an LDAP server.

To create a Directory Service, you use the ALBPM Configuration Wizard within the ALBPM Admin Center.



Note: The exact path you follow in the Configuration Wizard depends on whether you are configuring a database-only or hybrid directory service.



Note: If you are creating this directory service as part of a new installation, you can also use the ALBPM Configuration Wizard to configure the Process Execution Engine database.

1. Launch the ALBPM Admin Center

2. Click **Configuration**

The Configuration window appears.

3. Select the **Directory** tab.

4. Click **Add**.

The Configuration Wizard Tasks window appears.

5. Select **Create Directory Service**, and then click **Next**.

The Directory Provider Type window appears.

The Directory Provider Type window enables you to choose the type of directory service:

- A database managed by ALBPM. Selecting this option configures a directory service consisting of a relational database only.
- An external directory service provider plus a relational database. This configuration is also called a hybrid directory service. Selecting this option indicates that configuration information for the project is to be stored in the relational database, and participant identity information—including security credentials, roles and permissions, groups, and organizational units—is to be stored in either of the following:
 - One of the supported LDAP directories
 - The database for AquaLogic User Interaction (ALI), if you are integrating with ALI

6. Select the type of directory provider type, and then click **Next.**

The Directory Provider Selection window appears.

If you chose to configure a database managed by ALBPM, the Directory Provider Selection window prompts you to specify:

- The Directory Provider, that is, the relational database for the directory service. For information about supported directory providers and the parameters you must configure for each one, see [Engine and Directory Database Connectivity](#) on page 57.
- The user identifier and password for the BPM Administrator User. These are the credentials the Process Administrator retrieves when authenticating the ALBPM administrator.



Note: Once a directory is created with the respective directory.xml file, the BPM Administrator cannot be changed.

If you choose to create a hybrid directory service, the Directory Provider Selection window also prompts you to specify the organization provider, that is, an one of the supported LDAP directories or the ALI Identity Service.

7. Select the organization provider and click **Next.**

The Configure Directory Provider window prompts you for the connectivity information for the directory provider database. The information required depends on the database—for example, DB2, MS SQL, or Oracle. To learn more about the connectivity information you need to enter, see [Engine and Directory Database Connectivity](#) on page 57

8. Enter the connectivity information and click **Next.**

The Enter Directory Creation window prompts you for:

- The username and password of the database administrator
- The logical name of the organization for this environment. This symbolic name is used in contexts where processes in different environments communicate with each other.

After the Configuration Wizard runs successfully, your newly created directory service appears on the **Directory** tab page of the Configuration Wizard.

See [Configuring Organization using a DB-Only Directory Service](#) for information on creating an organization if you are using a database-only directory service.

See [Configuring a Hybrid Directory Service](#) *This high-level task demonstrates how to configure a hybrid directory service using Process Administrator.* for information on configuring a hybrid directory service.

Deploying and Publishing a New ALBPM Project

This task outlines the procedures for deploying and publishing a new project using the Process Administrator. See [Creating a Project Version](#) if you are publishing and deploying a new version of an existing project.

- 1. Launch the Process Administrator
- 2. Click **Projects**
- 3. Click Publish
The **Publication Source** pane appears
- 4. Select the Publication Source

| Option | Description |
|----------------------------|--|
| Project at Web Server Host | Allows you to select an ALBPM Project from the file system of the server where the Process Administrator is running. |
| Exported Project | Allows you to select an exported ALBPM Project from the file system of the local computer where your web browser is running. |

- 5. Select **Deploy processes after publishing them.**
- 6. Click**Ok.**
The**Publish Process**pane appears.
- 7. Expand**Role Mapping.**
- 8. Map the Roles
- 9. Click**Publish.**
The **Deployment Topology** pane appears.
- 10. Click**Ok.**
To perform a basic deployment of a project, use the default values for these fields.

Your project appears in the list of deployed projects.

Configuring ALBPM Enterprise on J2EE

The following sections describe how to configure the AquaLogic BPM Enterprise core applications. on J2EE. It contains specific procedures for BEA WebLogic Server and IBM WebSphere Application Server.

Before you begin configuring ALBPM Enterprise, ensure that you have reviewed [Configuration Overview](#) on page 8 and [Installation Prerequisites](#) on page 7 to determine what needs to be done before you begin and the order you need to perform the following procedures.

1. Review Architectural Information

See [JEE Cluster Architecture](#) on page 12. It is important to review this section if you are new to ALBPM.

2. Review Configuration Prerequisites

See [Installation Prerequisites](#) on page 7. This topic provides information about the required software you must install before using this guide.

3. Create an Organization

If you are configuring AquaLogic BPM in a production environment, you should ensure that you have created and configured your organizational structure before deploying and publishing a project. However, if you are creating a test environment, this is not required.

See the *ALBPM Enterprise Administration Guide* for more information on creating an organization.

4. Configure a Directory Service

The directory service is what allows you to connect projects deployed on ALBPM Enterprise with your organization.

5. Configure your J2EE application server

- If you are configuring BEA WebLogic Server see [Configuring WebLogic Server](#) on page 19.
- If you are configuring IBM WebSphere Application Server see [Configuring WebSphere Application Server](#) on page 24

6. Configure the Deployer

[Configuring the Deployer and Deployment Targets](#) on page 33

7. Enable clustering

[Enabling Clustering](#) on page 33

8. Build and deploy application EAR Files

[Building and Deploying Application EAR Files](#) on page 34

9. Deploy and Publish your ALBPM project.

[Deploying and Publishing a New ALBPM Project](#) on page 34


Creating a New Directory Service

If you are planning to use a database-only directory service, you need to install and configure only a relational database. If you are planning to use a hybrid directory service, you need to install and configure both a relational database and an LDAP server.

To create a Directory Service, you use the ALBPM Configuration Wizard within the ALBPM Admin Center.



Note: The exact path you follow in the Configuration Wizard depends on whether you are configuring a database-only or hybrid directory service.

 **Note:** If you are creating this directory service as part of a new installation, you can also use the ALBPM Configuration Wizard to configure the Process Execution Engine database.

1. Launch the ALBPM Admin Center
2. Click **Configuration**
The Configuration window appears.
3. Select the **Directory** tab.
4. Click **Add**.
The Configuration Wizard Tasks window appears.
5. Select **Create Directory Service**, and then click **Next**.
The Directory Provider Type window appears.


The Directory Provider Type window enables you to choose the type of directory service:

- A database managed by ALBPM. Selecting this option configures a directory service consisting of a relational database only.
- An external directory service provider plus a relational database. This configuration is also called a hybrid directory service. Selecting this option indicates that configuration information for the project is to be stored in the relational database, and participant identity information—including security credentials, roles and permissions, groups, and organizational units—is to be stored in either of the following:
 - One of the supported LDAP directories
 - The database for AquaLogic User Interaction (ALI), if you are integrating with ALI

6. Select the type of directory provider type, and then click **Next**.
The Directory Provider Selection window appears.

If you chose to configure a database managed by ALBPM, the Directory Provider Selection window prompts you to specify:

- The Directory Provider, that is, the relational database for the directory service. For information about supported directory providers and the parameters you must configure for each one, see [Engine and Directory Database Connectivity](#) on page 57.
- The user identifier and password for the BPM Administrator User. These are the credentials the Process Administrator retrieves when authenticating the ALBPM administrator.

 **Note:** Once a directory is created with the respective directory.xml file, the BPM Administrator cannot be changed.

If you choose to create a hybrid directory service, the Directory Provider Selection window also prompts you to specify the organization provider, that is, an one of the supported LDAP directories or the ALI Identity Service.

7. Select the organization provider and click **Next**.
The Configure Directory Provider window prompts you for the connectivity information for the directory provider database. The information required depends on the database—for example, DB2, MS SQL, or Oracle. To learn more about the connectivity information you need to enter, see [Engine and Directory Database Connectivity](#) on page 57
8. Enter the connectivity information and click **Next**.
The Enter Directory Creation window prompts you for:
 - The username and password of the database administrator
 - The logical name of the organization for this environment. This symbolic name is used in contexts where processes in different environments communicate with each other.

After the Configuration Wizard runs successfully, your newly created directory service appears on the **Directory** tab page of the Configuration Wizard.

See [Configuring Organization using a DB-Only Directory Service](#) for information on creating an organization if you are using a database-only directory service.

See [Configuring a Hybrid Directory Service](#) *This high-level task demonstrates how to configure a hybrid directory service using Process Administrator.* for information on configuring a hybrid directory service.

Configuring WebLogic Server

1.

Creating a WLS Domain

A domain is the basic administration unit for WebLogic Server instances. A domain consists of one or more WebLogic Server instances (and their associated resources) that you manage with a single Administration Server. To configure AquaLogic BPM on WebLogic Server, you must have a working WebLogic Server domain.

See *Configuring and Managing WebLogic Server* for more information on WebLogic Server domains.

The following procedures show you how to use the WebLogic Server Configuration Wizard to create a new domain configuration. However, there are other tools you can use to create a new WebLogic Server domain including the ALBPM Configuration Wizard. See [ALBPM Configuration Wizard Reference](#) on page 60

1. Launch the WebLogic Server Configuration Wizard

See *Creating WebLogic Domains Using the Configuration Wizard* for more information.

2. Create your domain configuration using the WebLogic Server Configuration Wizard.

You can use this wizard to create multiple managed servers and cluster depending on the requirements for you environment.

After creating your new domain configuration, you should start the Administration Server and each managed server to verify that everything has been configured successfully.

Creating ALBPM Deployer User

Before deploying the ALBPM Deployer application, you must create a WLS user with Administration privileges. The ALBPM Deployer application uses this user to install and manage applications on WebLogic.

1. Launch the WebLogic Server Administration Console.

2. Click **Lock and Edit**.

3. Click on **Security Realms**.

4. Select the security realm where you want to create the new user.

The default security realm is myrealm.

5. Click **Users and Groups**

6. Click **New**.

7. Enter the following information

| Option | Description |
|-----------------|---|
| Name | Specifies the name of the user. You must specify FuegoWebLogicDeployer. |
| Password | Defines the password for this user. |

8. Click **OK**

The new user appears in the list of users defined for this security realm.

9. Select the **FuegoWebLogicDeployer** user.
10. Select the **Groups** tab.
11. Add the **Administrator** groups to the user.
12. Click **Save**.
13. Click **Activate Changes** when you are finished.

Installing the ALBPM Deployer

The AquaLogic BPM Deployer is a JEE application that simplifies the procedures for deploying, starting, and stopping the ALBPM web applications using JEE.

The ALBPM Deployer acts as a bridge between the web application server and the ALBPM Process Administrator. Using the ALBPM deployer you can use the Process Administrator to control your web applications without using the administration console of your application server.

1. Launch the WebLogic Server Administration Console.
2. If you have not done so, click **Lock and Edit**.
3. Select **Deployments**.
4. Select **Install**.
5. Browse to the location of the Deployer EAR file.

The default location is: `<BEA_HOME>/albpm6.0/j2eewl/j2ee/weblogic/deployer`

6. Select `wlj2eedeployer.ear`
7. Click **Next**
8. Select **Install this deployment as an application**.
9. Click **Next**
10. Select the Administration Server as the deployment target



Note: You must target the Deployer to the administration server.

11. Click **Next**.
12. Click **Finish**.

After deployment, the `wlj2eedeployer` application appears in the list of deployments for the administration server.

13. Select **Activate Changes**.
14. Start the Deployer.
 - a) Select **Deployments**.
 - b) Click the checkbox next to `wlj2eedeployer`
 - c) Select **Start ► Servicing all requests**
 - d) Click **Yes**

The status of the `wlj2eedeployer` application shows active.

Creating JDBC Data Sources on WLS

To use AquaLogic BPM Enterprise on WebLogic Server, you must configure two JDBC data sources. These data sources allow the Process Execution Engine to connect to the Engine and Directory databases.


The following procedures show you how to create JDBC data sources using the WebLogic Server Administration Console. See *Configuring and Managing WebLogic JDBC* for more information.




Note: You must perform the following procedures twice to create two separate JDBC data sources for the Engine and Directory databases. These data sources must have the following names and JNDI names:

| Database | Data Source Name | JNDI Name |
|--------------------|------------------|------------|
| Engine Database | ALBPMServerDS | XAEngineDS |
| Directory Database | FDIDS | FDIDS |

1. Launch the Administration Console
2. In the **Domain Structure** tree, expand **Services ► JDBC**
3. Select **Data Sources**.
4. Click **New** on the Summary of Data Sources page.
5. Enter or select the following information

| Property | Description |
|-----------------|---|
| Name | The name of your JDBC data source. The data source name provided here must correspond to the required names for the Engine and Directory databases. |
| JNDI Name | The JNDI name of this data source. The JNDI name provided here must correspond to the required JNDI names for the Engine and Directory databases. |
| Database Type | Specifies the vendor of your directory and engine databases. |
| Database Driver | Specifies the JDBC driver version used by this data source.  Note: You must select a driver that supports transactions. |

6. Click **Next**.
The Transaction Options screen appears.
7. Provide the connectivity properties for your database.
The specific information you must provide depends on your database vendor.
8. Click **Next**
The Test Database Connection page appears.
9. Click **Test Configuration** to verify that you JDBC data source is configured correctly.
Although this step is not required, it is important to verify that the connectivity information you provided is correct.
10. Click **Next**.
11. Select the targets where you want to deploy your JDBC data source.
 **Note:** You MUST target each data source to all managed servers within each cluster.
12. Click **Finish**.


Your new JDBC data source appears in the list of data sources on the Summary of JDBC Data Sources page.

Creating a JMS Server

When running on WebLogic Server, AquaLogic BPM Enterprise uses JMS to synchronize and communicate between different components. A JMS Server functions as a management container for resources within JMS Modules.

See *Configuring and Managing WebLogic JMS* for more information on using JMS in WebLogic Server.

The following procedures show you how to configure JMS servers using the WebLogic Administration Console.

 **Note:** You must create a different JMS Server for each managed server in your environment.

1. Click **Lock and Edit**
2. Expand **Services ► Messaging**
3. Select **JMS Servers**.
4. Click **New**.
5. Enter a name for your JMS Server
6. Click **Next**.
7. Select the managed server where you want to target this JMS Server
8. Click **Finish**.


Your new JMS Server appears in the list of JMS servers. You should ensure that you have one JMS server targeted to each managed server in your cluster.

After you create a JMS Server, you must create and configure a JMS System Modules and JMS Resources managed by this server.


Creating a JMS Module

JMS Modules are application-related definitions that are independent of the domain environment. AquaLogic BPM requires you to create a JMS system module that is targeted to each managed server within a cluster.

JMS System modules are created at the container level as opposed to the applications level. System modules are used to configure JMS resources including queues and topic connections factories. System modules are configured as subdeployments to allow them to be targeted to multiple managed servers.

 **Note:** AquaLogic BPM Enterprise requires that each JMS module is targeted the cluster.

1. Expand **Services ► Messaging**
2. Select **JMS Modules**
3. Click **New**.
4. Provide a Name for the new Module
5. Click **Next**
6. Target the new module to the cluster.


 **Note:** The module must be targeted to the cluster, not to the managed servers within the cluster.

7. Click **Next**.
8. Click **Finish** to create the JMS system module.


The new JMS system module appear in the list of JMS modules. You can now create the required JMS resources.

Creating JMS Resources

Within a JMS module, you can define different configuration resources that are targeted to each JMS server.

 **Note:** JMS resources must be configured with the same JNDI names required by ALBPM Enterprise.

The name and JNDI name of your JMS resources must correspond to the names defined in Process Administrator. The default names and JNDI names are listed in the table below.

 **Note:** If you do not use the default names, you must change the values defined in Process Administrator. See the *ALBPM Enterprise Administration Guide* for more information.

| JMS Resource | JNDI Name | Description |
|--------------------------|--------------------------|-------------|
| Distributed Topic | topic/EngineNews | |
| Distributed Queue | queue/ToDoQueue | |
| XAConnectionFactory | XAConnectionFactory | |
| XATopicConnectionFactory | XATopicConnectionFactory | |

1. Expand **Services ► Messaging**
2. Select **JMS Modules**.
3. Select the system module you created in the previous task.
4. Click **New**.
5. Select the type of resource you want to create.
You must create each of the resources defined in the table above.
6. Click **Next**.
7. Provide the following properties for your resource:

| Property | Description |
|-----------|---|
| Name | Specifies the name of your JMS resource. See the table above for the correct names. |
| JNDI Name | Specifies the JNDI name for your JMS Resource. See the table above for the correct JNDI name. |

8. Click **Next**.
9. Click **Advanced Targeting**.
10. Select the subdeployment created in the previous task.
11. Select the JMS Server where you want to target the subdeployment.



Note: You must target the subdeployment to each of the JMS servers you defined in earlier tasks. Do not target the subdeployment to the cluster or managed servers.

12. Click **Finish**.

Disabling Server Affinity for Connection Factories

You should disable server affinity for each of the connection factory resources required by ALBPM

To ensure that ALBPM work correctly within a WLS cluster, you must disable server affinity from the XAConnectionFactory and XATopicConnectionFactory created previously.

1. Click **Lock and Edit**.
2. Expand **Services ► Messaging**.
3. Select **JMS Modules**
4. Select the system module where the ALBPM connection factories are stored.
5. Select the connection factory.
6. Click the **Load Balancing** tab.
7. Deselect **Server Affinity Enabled**.
8. Click **Save**.
9. Click **Activate Changes**.

Configuring WebSphere Application Server

1.

Installing ALBPM Deployer

The AquaLogic BPM Deployer is a JEE application that simplifies the procedures for deploying, starting, and stopping the ALBPM web applications using JEE.

The ALBPM Deployer acts as a bridge between the web application server and the ALBPM Process Administrator. Using the ALBPM deployer you can use the Process Administrator to control your web applications without using the administration console of your application server.

1. Login to WebSphere's Administration Console.

By default, it runs on `http://host:9060/ibm/console`.

2. Go to **Applications ► Install New Applications** and select ALBPM Deployer EAR file.

The ALBPM Deployer consists of an EAR file provided with the installation of ALBPM Enterprise. It is located under the following directory: `BEA_HOME/albpm6.0/j2eews/j2ee/websphere/deployer`.

Two copies are provided. You should pick one depending on whether your WebSphere environment has *Global Security* enabled:

- `wasj2eedeployer.ear`: If security is disabled.
- `wasj2eedeployersecured.ear`: If security is enabled.



Note: If WebSphere's Administration Console didn't prompt you for a password at login, it probably means *Global Security* is disabled.

3. Click **Next** and review the installation options.

Accepting the default installation options works for most cases. Always refer to WebSphere's official documentation for details.

4. Click **Next** and select the WebSphere server where to install ALBPM Deployer.

On a single-server setup, you must install it on the only server available.



Restriction: On a clustered WebSphere environment, ALBPM Deployer must be installed in the Deployment Manager server (*dmgr*). WebSphere's Console may not allow you to pick the Deployment Manager server as a target for installing applications; If this is the case, you must install ALBPM Deployer using other means, such as WebSphere's `wsadmin` command-line tool.

5. Click **Next** again, and **Finish** after reviewing all selected options.

You should get confirmation that *Fuego WebSphere Deployer* application was successfully installed.

6. Save your WebSphere configuration changes.

7. Go to **Applications ► Enterprise Applications** and start Fuego WebSphere Deployer application.

The ALBPM WebSphere Deployer is now installed and ready.

Adding Authentication Data for JDBC

Before creating JDBC datasources, you must define the authentication information for the JDBC connections.

You must create two sets of authentication data: one for the ALBPM Directory Database and one for ALBPM Engine's backend database.

1. Login to WebSphere's Administration Console. By default, it runs on `http://host:9060/ibm/console`.

- Go to **Security ► Secure administration, applications and infrastructure**, expand the **Java Authentication and Authorization Service** section on the right pane and click on **J2C authentication data**.
- Click **New** to create a new entry for ALBPM Directory database.

Enter the following fields:

| Option | Value |
|----------|--|
| Alias | ALBPM Directory DB Auth |
| User ID | <i>The JDBC user id to access ALBPM Directory Database.</i> |
| Password | <i>The password for the above user to access ALBPM Directory Database.</i> |

You must enter the same user id and password you specified when creating the ALBPM Directory database using the [Configuration Wizard](#).

- Click **OK** to save.
- Click **New** to create a new entry for ALBPM Engine database.

Enter the following fields:

| Option | Description |
|----------|---|
| Alias | ALBPM Engine DB Auth |
| User ID | <i>The JDBC user id to access ALBPM Engine Database.</i> |
| Password | <i>The password for the above user to access ALBPM Engine Database.</i> |

You must enter the same user id and password you specified when creating the ALBPM Engine database using the [Configuration Wizard](#).

- Click **OK** to save.
- Click on **Save** to persist your WebSphere configuration changes.

The new J2C authentication data entries are ready to be used from a JDBC Datasource configuration.

Creating a New JDBC Provider

You must create and configure a new JDBC provider to use the JDBC drivers provided by BEA.

- Login to WebSphere's administration console.
By default, the administration console runs at: `http://host:9060/ibm/console`.
- Select **Resources ► JDBC ► JDBC Providers**
- Click **New** to add a new JDBC Provider.



Note: You must select a WebSphere configuration *scope* for your new resources. You may select *cell* scope. Refer to WebSphere's official documentation for more details about its configuration scoping rules.

- Enter the following configuration values for the new provider:

| Option | Value |
|---------------------------|---|
| Database type | User-Defined |
| Implementation class name | <p>Enter one of the following class names depending on which database provider you are using:</p> <ul style="list-style-type: none"> Oracle: <code>albpn.jdbcx.oracle.OracleDataSource</code> |

| Option | Value |
|--------|---|
| | <ul style="list-style-type: none"> • SQL Server: albpn.jdbcx.sqlserver.SQLServerDataSource • DB2: albpn.jdbcx.db2.DB2DataSource |
| Name | BEA JDBC Provider |



Note: Ensure that you select User-defined for each of the supported database types. You must select this option in order to configure the JDBC drivers supplied by ALBPM.

5. Press **Next** to enter the JDBC Classpath information.

You must list the following .jar files:

```
BEA_HOME/albpn6.0/j2eews/libraries/bloracle.jar (Oracle only)
BEA_HOME/albpn6.0/j2eews/libraries/bldb2.jar (DB2 only)
BEA_HOME/albpn6.0/j2eews/libraries/blsqlserver.jar (SQL Server only)
BEA_HOME/albpn6.0/j2eews/libraries/blutil.jar
BEA_HOME/albpn6.0/j2eews/libraries/blresource.jar
BEA_HOME/albpn6.0/j2eews/libraries/blbase.jar
BEA_HOME/albpn6.0/j2eews/ext/fuego.database.websphere.helper.jar
```

These files are installed with ALBPM Enterprise. You must specify the full path to each of these jar files. Each must be defined on its own line.

To avoid hard-coding the full paths, you can copy the .jar files to a common directory, then define a new WebSphere environment variable (e.g.BEA_JDBC_DRIVERS_PATH) containing the path to this directory. You can then, specify the classpath as follows:

```
${BEA_JDBC_DRIVERS_PATH}/bloracle.jar
${BEA_JDBC_DRIVERS_PATH}/blutil.jar
${BEA_JDBC_DRIVERS_PATH}/blresource.jar
${BEA_JDBC_DRIVERS_PATH}/blbase.jar
${BEA_JDBC_DRIVERS_PATH}/fuego.database.websphere.helper.jar
```

6. Click **Next** again, and **Finish** after reviewing all selected options.

7. Click on **Save** to persist your WebSphere configuration changes.

The new JDBC provider using BEA's Oracle drivers is now configured.

Creating a JDBC Datasource for Directory Database

After creating a JDBC Provider, you must create a JDBC datasource to provide database connectivity to ALBPM applications. The following procedures show you how to create a JDBC datasource for the ALBPM directory database.

Before creating the JDBC Datasources you must have defined:

- A new JDBC provider containing the drivers provided by BEA, as explained in [Creating a New JDBC Provider](#) on page 25.
- Two authentication data entries: one for the ALBPM Directory Database and one for ALBPM Engine's backend database, as explained in [Adding Authentication Data for JDBC](#) on page 24.

To create a JDBC datasource for the directory DB:

1. Login to WebSphere's Administration Console. By default, it runs on <http://host:9060/ibm/console>.
2. Go to **Resources** ► **JDBC** ► **JDBC Providers** and click on the BEA JDBC Provider you created previously.
3. Click on **Data sources**, located in the **Additional properties** section.

4. Click **New** to create a new Data source for ALBPM Directory database.
5. Enter the following configuration information for your new data source:

| Option | Value |
|--|--|
| Data Source Name | ALBPM Directory Data Source |
| JNDI Name | FDIDS |
| Component-managed Authentication Alias | Select the Authentication entry you defined previously for ALBPM Directory database. For example: <i>node/ALBPM Directory DB Auth</i> |

6. Click **Next** and enter the following Data store helper class name:
`albpn.jdbc.websphere.ALBPMDataStoreHelper`.
7. Click **Next**, then **Finish** after reviewing all selected options.
8. Select the data source you just created.
9. Click **Custom properties** located under the **Additional properties** section in the right pane.

You should see a complete list of all driver properties. If you do not see this list, there is an error with your configuration.
10. Configure the connectivity properties for your database vendor.

See [Engine and Directory Database Connectivity](#) on page 57 for more information. For general information on using the JDBC drivers provided with BEA products, see [JDBC Drivers](#).
11. Click **Save** to persist your WebSphere configuration changes.
12. Test your datasource configuration.
 - a) Expand **JDBC ► JDBC Providers**
 - b) Select the JDBC datasource you just configured
 - c) Click **Data sources**
 - d) Select the checkbox next to the datasource you want to test
 - e) Click **Test Connection**

The JDBC datasources for the ALBPM Directory database is configured.

Creating a JDBC Datasource for the Engine Database

After creating a JDBC datasource for the directory database, you another JDBC resource to provide database connectivity to ALBPM applications for the ALBPM engine database.

Before creating the JDBC Datasources you must have defined:

- A new JDBC provider containing the drivers provided by BEA, as explained in [Creating a New JDBC Provider](#) on page 25.
- Two authentication data entries: one for the ALBPM Directory Database and one for ALBPM Engine's backend database, as explained in [Adding Authentication Data for JDBC](#) on page 24.

To configure a JDBC datasource for the engine database:

1. Login to WebSphere's Administration Console. By default, it runs on `http://host:9060/ibm/console`.
2. Go to **Resources ► JDBC ► JDBC Providers** and click on the BEA JDBC Provider you created previously.
3. Click on **Data sources**, located in the **Additional properties** section.
4. Click **New** to create a new Data source for ALBPM Engine database.

Enter the following configuration information for the new data source:

| Option | Value |
|--|--|
| Data source name | ALBPM Engine Datasource |
| JNDI name | XAEngineDS |
| Component-managed Authentication Alias | <i>Select the Authentication entry you defined previously for ALBPM Engine database. Example:</i> <node>/ALBPM Engine DB Auth |

- Click **Next** and enter the following Data store helper class name:
`albpn.jdbc.websphere.ALBPMDataStoreHelper`.
- Click **Next** again, and **Finish** after reviewing all selected options.
- Select your newly created Data source.
- Click **Custom properties** located under the **Additional properties** section.
- Configure the connectivity properties for your database vendor.
See [Engine and Directory Database Connectivity](#) on page 57 for more information. For general information on using the JDBC drivers provided with BEA products, see [JDBC Drivers](#).
- Click on **Save** to persist your WebSphere configuration changes.

The JDBC Datasource for ALBPM Engine database is configured.

Creating a WebSphere JMS Bus

You need to provide a JMS service to ALBPM applications. This procedure shows you how to configure WebSphere Application Server's Embedded JMS service. Other external JMS providers may be used. Check the Installation Guide for compatibility information.

WebSphere Application server defines the concept of a *Bus* to manage JMS resources. Before creating Queues and Topics, you must create a Bus.

- Login to WebSphere's Administration Console. By default, it runs on `http://host:9060/ibm/console`.
- Create a new Bus
 - Go to **Service Integration ► Buses** press **New** to add a new Bus.
 - Enter a name for this Bus. Example: ALBPM Bus.
 - Press **Next** again, and **Finish**.

- Add members (servers/clusters) to your newly created Bus.

Click on your Bus and then on **Bus members**, located in the **Topology** section.

For each server you want to add as a member:

- Press **Add** and select the server/cluster to add as member.
 - Press **Next** and select the type of message store you want. You may use the default.
 - Press **Next** and review the message store properties. You may use the default values.
 - Press **Next** again, and **Finish** after reviewing all selected options.
- Create a Queue Destination in your Bus:
 - Go back to **Service Integration ► Buses**, and click on your newly created Bus.
 - Click on **Destinations** located in the **Destination Resources** section on the right.
 - Press **New** and select Queue for the destination type.
 - Enter `ALBPMEngineQueue` as the Identifier.
 - Press **Next** and select the bus member which will serve this Queue destination.
 - Press **Next** again, and **Finish**.
 - Create a Topic Space in your Bus:

- a) Go back to **Service Integration ► Buses**, and click on your newly created Bus.
 - b) Click on **Destinations** located in the **Destination Resources** section on the right.
 - c) Press **New** and select Topic space for the destination type.
 - d) Enter ALBPMTopic as the Identifier.
 - e) Press **Next** and **Finish**.
6. Click on **Save** to persist your WebSphere configuration changes.

Creating JMS Resources

You need to define a JMS Queue Connection Factory, a Queue, a Topic Connection Factory and a Topic for ALBPM applications.

1. Login to WebSphere's Administration Console. By default, it runs on `http://host:9060/ibm/console`.

2. Create a Queue Connection Factory:

- a) Go to **Resources ► JMS ► Queue connection factories**.



Note: You must select a WebSphere configuration *scope* for your new resources. You may select *cell* scope. Refer to WebSphere's official documentation for more details about its configuration scoping rules.

- b) Press **New**. Select **Default messaging provider** and press **OK**.
- c) Review the configuration properties for this Connection Factory.

Make sure you set the following properties:

| Option | Value |
|-----------|---|
| Name | ALBPM QCF |
| JNDI Name | XAConnectionFactory |
| Bus name | Select the name of the JMS Bus you created (see Creating a WebSphere JMS Bus on page 28). Example: ALBPM Bus |

- d) Press **OK** when finished setting the configuration properties.

3. Create a Topic Connection Factory

- a) Go to **Resources ► JMS ► Topic connection factories**.



Note: You must select a WebSphere configuration *scope* for your new resources. You may select *cell* scope. Refer to WebSphere's official documentation for more details about its configuration scoping rules.

- b) Press **New**. Select **Default messaging provider** and press **OK**.
- c) Review the configuration properties for this Connection Factory.


Make sure you set the following properties:

| Option | Value |
|-----------|---|
| Name | ALBPM TCF |
| JNDI Name | XATopicConnectionFactory |
| Bus name | Select the name of the JMS Bus you created (see Creating a WebSphere JMS Bus on page 28). Example: ALBPM Bus |

- d) Press **OK** when finished setting the configuration properties.

4. Create a Queue

- a) Go to **Resources ► JMS ► Queues**.

 **Note:** You must select a WebSphere configuration *scope* for your new resources. You may select *cell* scope. Refer to WebSphere's official documentation for more details about its configuration scoping rules.

- b) Press **New**. Select **Default messaging provider** and press **OK**.

- c) Review the configuration properties for this Queue.


Make sure you set the following properties:

| Option | Value |
|---------------|---|
| Name | ALBPM Engine Queue |
| JNDI Name | queue/ToDoQueue |
| Bus name | Select the name of the JMS Bus you created (see Creating a WebSphere JMS Bus on page 28). Example: ALBPM Bus |
| Queue name | Select the name of your Queue Destination (see Creating a WebSphere JMS Bus on page 28). Example: ALBPMEngineQueue |
| Delivery Mode | Nonpersistent |

- d) Press **OK** when finished setting the configuration properties.

5. Create a Topic

- a) Go to **Resources ► JMS ► Topics**.

 **Note:** You must select a WebSphere configuration *scope* for your new resources. You may select *cell* scope. Refer to WebSphere's official documentation for more details about its configuration scoping rules.

- b) Press **New**. Select **Default messaging provider** and press **OK**.

- c) Review the configuration properties for this Topic.


Make sure you set the following properties:

| Option | Value |
|-------------|---|
| Name | ALBPM Topic |
| JNDI Name | topic/EngineNews |
| Bus name | Select the name of the JMS Bus you created (see Creating a WebSphere JMS Bus on page 28). Example: ALBPM Bus |
| Topic space | Select the name of your Topic space (see Creating a WebSphere JMS Bus on page 28). Example: ALBPMTopic |

- d) Press **OK** when finished setting the configuration properties.

6. Create a JMS Activation specification

- a) Go to **Resources ► JMS ► Activation specification**.

 **Note:** You must select a WebSphere configuration *scope* for your new resources. You may select *cell* scope. Refer to WebSphere's official documentation for more details about its configuration scoping rules.

- b) Press **New**. Select **Default messaging provider** and press **OK**.
- c) Review the configuration properties for this Activation Spec.

Make sure you set the following properties:


| Option | Value |
|-----------------------|---|
| Name | ALBPM Activation |
| JNDI Name | EngineActiveSpec |
| Destination type | Queue |
| Destination JNDI name | queue/ToDoQueue |
| Bus name | Select the name of the JMS Bus you created (see Creating a WebSphere JMS Bus on page 28). Example: ALBPM Bus |

- d) Press **OK** when finished setting the configuration properties.
7. Click on **Save** to persist your WebSphere configuration changes.

Configuring a Work Manager

You need to create a Work Manager configuration in WebSphere to provide asynchronous transaction processing capabilities to ALBPM Engine.

1. Login to WebSphere's Administration Console. By default, it runs on `http://host:9060/ibm/console`.
2. Go to **Resources** ► **Asynchronous beans** ► **Work managers** and press **New** to create a new Work Manager.

 **Note:** You must select a WebSphere configuration *scope* for your new resources. You may select *cell* scope. Refer to WebSphere's official documentation for more details about its configuration scoping rules.

3. Enter the following configuration values for the new Work manager:

| Option | Value |
|-----------|---------------------|
| Name | ALBPM WorkManager |
| JNDI name | wm/albpmWorkManager |

4. Press **OK** after reviewing all selected options.
5. Click on **Save** to persist your WebSphere configuration changes.

The new Work manager for ALBPM Engine is now configured.

Configuring JVM Properties

You need to configure each WebSphere Server to use additional system properties when launching the Java Virtual Machine.

1. Login to WebSphere's Administration Console. By default, it runs on `http://host:9060/ibm/console`.
2. Go to **Servers** ► **Application servers**. You will see a list of all available servers.
3. Follow this procedure for each Server in which you will install ALBPM applications:
 - a) Expand **Java and Process Management** and click on **Process Definition**
 - b) Click on **Java Virtual Machine** then on **Custom Properties**.

c) Add the following properties:

| New Property | Value |
|-------------------|-------|
| java.awt.headless | true |

Restarting WebSphere Servers

After configuring all the needed resources in your WebSphere installation, you must re-start all your WebSphere Servers, Node managers and Deployment manager.

After re-starting WebSphere, you may want to check your Database connections:

1. Login to WebSphere's Administration Console. By default, it runs on `http://host:9060/ibm/console`.
2. Go to **Resources** ► **JDBC** ► **Data sources**.
3. Select your ALBPM Engine and ALBPM Directory data sources.
4. Click **Test Connection** button.

You might want to check WebSphere's log files in case of errors.

Setting ALBPM properties for WebSphere

You must configure your Process Engine with some WebSphere-related properties via ALBPM Process Administrator.

You must have an ALBPM Engine for WebSphere configured in order to define this configuration properties.

1. Login to ALBPM Process Administrator. By default, it runs on `http://host:8686/webconsole`.
2. Click on **Engines** and then click on the name of your ALBPM Engine for WebSphere.
You should see the configuration properties for your Engine.
3. Click on the **Application Server** tab.
4. In the **JMX Engine Management Configuration** section you must specify the **Host** and **Port** of one of the WebSphere servers that will be running ALBPM Engine.

The port number is that of the WebSphere server's SOAP Connector address. You can obtain this value from the WebSphere Administration console: go to **Servers** ► **Application servers** ► **YOUR_SERVER (e.g. server1)** ► **Ports**, and look for the value of the **SOAP_CONNECTOR_ADDRESS** property.

For example, on a default single-node WebSphere configuration where ALBPM Process Admin is running on the same host as WebSphere you would use:

| | |
|-------------|-----------|
| Host | localhost |
| Port | 8880 |

5. Locate the **BPM Application Deployer URL** field, and change its value to match your WebSphere configuration.

For example, on a default single-node WebSphere configuration, it would be:

```
http://localhost:9080/fuego/deployer/servlet/worker
```

The URL should be that of the WebSphere server where ALBPM Deployer is installed, which must be your WebSphere's Deployment Manager server (dmgr).

Refer to [Installing ALBPM Deployer](#) on page 24 for more details.

6. Change the **WebSphere Server/Cluster Name** field to match your WebSphere configuration. This is the name of the WebSphere Server or Cluster where you want to install the ALBPM applications.

For example, on a default single-node WebSphere configuration, it would be: `server1`.

Preparing ALBPM Applications

Some of the ALBPM applications need additional configuration changes before deploying them into WebSphere.

1. Copy file `al bpm6.0/j2eews/j2ee/websphere/was6stubs/fuegoj2ee-was6-stubs.jar` into each of following directories:

```
al bpm6.0/j2eews/webapps/archivingviewer/WEB-INF/lib
al bpm6.0/j2eews/webapps/feeds/WEB-INF/lib
al bpm6.0/j2eews/webapps/mobile/WEB-INF/lib
al bpm6.0/j2eews/webapps/papiws/WEB-INF/lib
al bpm6.0/j2eews/webapps/portal/WEB-INF/lib
al bpm6.0/j2eews/webapps/portaladmin/WEB-INF/lib
al bpm6.0/j2eews/webapps/webconsole/WEB-INF/lib
al bpm6.0/j2eews/webapps/workspace/WEB-INF/lib
```

This is to allow client applications to connect with the EJB-based Process Engine running on WebSphere.

2. Create a new directory named `jmxextensions` under `al bpm6.0/j2eews/webapps/webconsole/WEB-INF/`.
3. Copy file `WEBSPPHERE_HOME/AppServer/runtimes/com.ibm.ws.admin.client_6.1.0.jar` to the `jmxextensions` directory you just created on the previous step:
`al bpm6.0/j2eews/webapps/webconsole/WEB-INF/jmxextensions/`.

This is to allow the Process Administrator application to connect with the Process Engines via JMX.

Configuring the Deployer and Deployment Targets

1. Launch the Process Administrator
2. Select **Engines**
3. Select the engine where you want to configure the deployer.
4. Select the **Application Server** tab.
5. Enter the following information:

| Property | Description |
|---|---|
| BPM Application Deployer URL | Specifies the URL of the deployer. The default location is: <code>http://localhost:7001/fuego/deployer/servlet/worker</code> |
| WebLogic/WebSphere Server or Cluster Name | Specifies the name server or cluster target where the ALBPM applications are deployed. |

After defining the Deployer URL and specifying the target cluster or server, you can use the Process Administrator to deploy the AquaLogic BPM applications.

Enabling Clustering

If you are configuring your ALBPM Enterprise to run within a clustered JEE environment, you must enable clustering using the Process Administrator.

1. Launch the Process Administrator

2. Select **Engines**
3. Select the engine where you want to enable clustering.
4. Select **Basic Configuration**
5. Enable the **Cluster** checkbox.

After enabling clustering, the **Cluster** tab appears in the **Edit Engine** page.


Building and Deploying Application EAR Files



The ALBPM Process Administrator allows you to create the ALBPM application .ear files and deploy them on your application server.


To use Proces Administrator to create and deploy ALBPM application EAR files, you must deploy and start the ALBPM Deployer application. Before creating the ALBPM application archives, you must configure a process execution engine.

1. Login to ALBPM Process Administrator.
2. Click on **Engines**
3. Select the engine where you want to create the application
You should see the configuration properties for your Engine.
4. Click **Basic Configuration**.
5. Click **J2EE Application Server Files**.

This page allows you to generate the EAR files for the ALBPM applications associated with this Engine.

 **Note:** When you access this page, the Process Administrator gets the status of each of the applications by contacting ALBPM Deployer. You will get a warning message at the bottom of the page if there was any problem contacting ALBPM Deployer. If this is the case, make sure the **BPM Application Deployer URL** (within the **Application Server** tab) is correct and that ALBPM Deployer is up and running on WebSphere.

6. Click **Create EAR** () next to each of the applications you want to install.
7. Click **Install** () next to each of the applications you want to install.

 **Attention:** This may take several minutes. Do not click any link on the page and do **back** in your browser until the page is automatically reloaded. When you click on the icon, ALBPM Process Administrator transfers the file over to WebSphere's Deployment Manager (by means of ALBPM Deployer) and then WebSphere goes through the application installation process.

Deploying and Publishing a New ALBPM Project

This task outlines the procedures for deploying and publishing a new project using the Process Administrator. See [Creating a Project Version](#) if you are publishing and deploying a new version of an existing project.

1. Launch the Process Administrator
2. Click **Projects**
3. Click **Publish**
The **Publication Source** pane appears
4. Select the Publication Source

| Option | Description |
|----------------------------|--|
| Project at Web Server Host | Allows you to select an ALBPM Project from the file system of the server where the Process Administrator is running. |
| Exported Project | Allows you to select an exported ALBPM Project from the file system of the local computer where your web browser is running. |

5. Select **Deploy processes after publishing them**.

6. Click **Ok**.

The **Publish Process** pane appears.

7. Expand **Role Mapping**.

8. Map the Roles

9. Click **Publish**.

The **Deployment Topology** pane appears.


10. Click **Ok**.

To perform a basic deployment of a project, use the default values for these fields.

Your project appears in the list of deployed projects.

Integrating ALBPM with Other BEA Products

This section contains tasks for integrating AquaLogic BPM with other BEA products.

 **Note:** Before performing the tasks in this section, review [Configuration Overview](#) on page 8

Integrating BEA AquaLogic Service Bus

AquaLogic Service Bus functions as a repository for business services exposed as web services. AquaLogic BPM processes exposed as web services can be consumed by ALSB. Also, web services provided by ALSB can be cataloged as part of an ALBPM project.

See <http://e-docs.bea.com/alsb/docs20/index.html> for more information on using AquaLogic Service Bus.

Registering an ALBPM Process on AquaLogic Service Bus

The following high-level task outlines the procedures for registering ALBPM processes with BEA AquaLogic Service Bus.

1. Create a Management Host Configuration external resource.
See [Creating a Management Host Configuration](#) on page 36
2. Create a Process Registration external resource.
See [Creating a Process Registration Configuration](#) on page 37
3. Register Your ALBPM Process with AquaLogic Service Bus.
See [Registering an ALBPM Process with ALSB](#) on page 37

Creating a Management Host Configuration

A Management Host configuration allows you to define the connectivity information for your ALSB installation.

1. Launch Process Administrator
2. Click**External Resources**.
3. Click**Add**.
4. Provide a name for your external resource.
5. Select **AquaLogic Service Bus** under**Type**.
6. Select **Management Host** under**Subtype**.
7. Click**Next**.
8. Provide the following connectivity information for your ALSB installation:

| Property | Description |
|----------|--|
| Host | Specifies the hostname for your ALSB installation. This can be the DNS hostname or IP address. |
| Port | Specifies the port for your ALSB installation. |
| User | Specifies the user name for your ALBSB adminsitrator. |
| Password | Specifies the password for your ALSB administrator. |

9. Click **Save** to create the new external resource.

The new external resource is created based on the connectivity information you provided.



Note: Process Administrator does not verify the connection to the ALSB installation. If you have trouble connecting to your ALSB installation verify the connectivity information.

Creating a Process Registration Configuration

A Process Registration external resource allows you to define the server, project, and folders where the ALBPM process is registered.

1. Launch Process Administrator
2. Click **External Resources**.
3. Click **Add**.
4. Provide a name for your external resource.
5. Select **AquaLogic Service Bus** under **Type**.
6. Select **Process Registration** under **Subtype**.
7. Click **Next**.
8. Select a Management Host external resource.

The Management Host configuration specifies the connectivity information to the ALSB installation.

9. Enter the following information:

| Property | Description |
|---------------------------------|--|
| Project | Specifies the project where ALSB stores the ALBPM processes. |
| WSDLS Folder | Specifies WSDL folder for the ALSB project. The default is WSDL |
| Business Services Folder | Specifies the Business Services Folder for the ALSB project. |

10. Click **Create Structure**.

Process Administrator connects to AquaLogic Service Bus to create the WSDL definitions

11. Click **Save**.

Registering an ALBPM Process with ALSB

Before performing the following task, ensure that you have exposed the processes you want to register as Web Services. You must do this before publishing and deploying your ALBPM project. See the *ALBPM Studio User Guide* for more information.

You should also ensure that you have created external resources for the ALSB Management Host Configuration and Process Registration Configuration.

- 1.

Publishing Projects with Catalogued ALSB Resources

Supported Transport Protocols

You can configure AquaLogic BPM and AquaLogic Service Bus proxy services and business services to use different transport protocols. The transport protocol you select depends on the service type, the type of authentication required, the service type of the invoking service, and so on.

AquaLogic BPM supports the following transport protocols:

- HTTP
- HTTPS

- ALBPM Custom Transport Protocol

For information on using the HTTP and HTTPS transport protocols, see [Transports](#) in the *BEA AquaLogic Service Bus User Guide*.

ALBPM Custom Transport

In addition to HTTP and HTTPS, ALBPM provides a custom transport protocol. To use the ALBPM custom transport protocol, you must deploy the ALBPM transport provider.

You must deploy the ALBPM transport provider like other ALSB transport providers. See [Deploying the Transport Provider](#) in the *Transport SDK User Guide* for information.

The AquaLogic BPM transport provider .ear file is installed with ALBPM. The default location is:

`ALBPM_HOME / j2ee/w1/j2ee/weblogic/alsb/alsb.transport.ear.`

If your ALSB installation resides on a separate machine than your ALBPM installation, you must copy this file to the ALSB location.

Integrating AquaLogic BPM with WebLogic Portal

The tasks in this section show you how to configure AquaLogic BPM WorkSpace to be used as a portal within WebLogic Portal.

Before you begin configuring ALBPM Enterprise, ensure that you have reviewed [Configuration Overview](#) on page 8 and [Installation Prerequisites](#) on page 7 to determine what needs to be done before you begin and the order you need to perform the following procedures.

1. Review Architectural Information

See [WebLogic Portal Architecture](#) on page 12. It is important to review this section if you are new to ALBPM.

2. Review Configuration Prerequisites

See [Installation Prerequisites](#) on page 7. This topic provides information about the required software you must install before using this guide.

3. Configure your ALBPM domain.

See [Configuring ALBPM Domains](#) on page 38

4. Configure Single Sign On (SSO)

See [Configure Single Sign On \(SSO\)](#) on page 39.

5. Create the ALBPM application files

6. Create a portal

7. Update the Portal Web Project

8. Disable JSP validation

9. Refresh the web project

10. Create ALBPM portlets

11. Create the Login portlet

12. Verify your installation

Configuring ALBPM Domains


1.

Configuring a Single Domain

The following procedures show how to configure AquaLogic BPM and WebLogic Portal integration using a single domain.

The BEA WebLogic Configuration Wizard allows you to easily create and configure a new WebLogic Server domain.

1. Start the BEA WebLogic Configuration Wizard
2. Using the following table as a guide, complete each screen of the wizard:

| Wizard Screen | Notes |
|---|--|
| Welcome | Select Create a New WebLogic Domain |
| Select Domain Source | Choose Generate a domain configured automatically, then select the following: <ul style="list-style-type: none"> • WorkShop for WebLogic Platform • WebLogic Portal |
| Configure Administrator Username and Password | Enter the domain administrator username and password. For simplicity in a testing environment, it is recommended that you use weblogic/weblogic.  Note: This username must also exist in the ALBPM user base. |
| Configure Server Start Mode and JDK | Select the Jrockit SDK |
| Customize Environment and Services Settings | Choose No . |
| Create WebLogic Domain | Enter the domain name. You will point to this domain when running the ALBPM Configuration Wizard. |
| Creating Domain | Select Done after the domain has been created. |

3. Start the Administration Server

When running the ALBPM Configuration Wizard in the next step, the Administration Server must be running. This also allows you to verify that your domain was created and configured successfully.

Configuring Multiple Domains

- 1.

Configure Single Sign On (SSO)

The following procedures show you how to configure Single Sign On (SSO) when integrating AquaLogic BPM with WebLogic Portal.

Single Sign On (SSO) allows a user to access more than one application, while providing login information once.

1. Launch the AquaLogic BPM Admin Center
2. Click **Configuration**.
3. Click the **WorkSpace** tab.
4. Select **Enable SSO**.
5. Click **Ok**.

Creating the ALBPM Application Files

ALBPM WorkSpace is designed to perform authentication by itself. To use WorkSpace within a WebLogic Portal container, ALBPM authentication must be container-based. This requires you to configure the ALBPM Directory Service to work in trusted mode. WorkSpace must also be configured to support this configuration from a client perspective.

The following procedures show you how to create and configure the ALBPM WorkSpace WAR file. After creating the WAR file, you must expand, edit, and reassemble it.

1. Launch the ALBPM Admin Center
2. Select **Start BPM Applications**.
3. Click **Launch Process Administrator**.
4. Provide Login information.
5. Click **Engines** in the left hand pane.
6. Click the engine whose WAR file you want to generate.
7. Click **J2EE Application Server File**.
8. Click the icon next to **BEA AquaLogic BPM WorkSpace WAR**.

The ALBPM WorkSpace WAR file is generated and the link appears under the explanation.

9. Right-click `08-workspace-wp-FDIDS.war`, then select **Save Link As**.
10. Select a location to save the WAR file.
11. Click **Logout**.

Deploying the WAR File in WebLogic Portal

The following procedures show you how to deploy the modified WAR file in WebLogic Portal using WorkShop for WebLogic Platform.

1. Start BEA WorkShop for WebLogic Platform.
2. Select a workspace, then click **Ok**.
3. Click Workbench
4. Create a new WebLogic Portal Ear project.
 - a) Select **File ► New ► Other**
 - b) Expand WebLogic Portal
 - c) Select Portal Ear Project, then click **Next**.
 - d) Provide a project name, then click **Next**.
 - e) Ensure that the following are selected:
 - EAR
 - WebLogic EAR Extensions
 - WebLogic Portal
 - f) Click **Finish**.
5. Create a Portal Web Project

You must add this project to the Portal EAR project created in the previous step. You must also select JSF from Project Facelets when creating the Web Project.

- a) Select **File ► New ► Portal Web Project**
- b) Enter the Project Name
- c) Check the **Add project to an EAR** checkbox.
- d) Select the EAR file edited in previous tasks.
- e) Click **Next**.

- f) Ensure that the JSF option is selected.
- g) Click **Next**.
- h) Click **Finish**.

6. Specifying a WebLogic Server definition.

This must point to the same WebLogic Portal domain created earlier.

- a) Open the Server View.
- b) Right click in the view, then select **New ► Server**
- c) Select **WebLogic Server 10**.
- d) Click **Next**.
- e) Click **Browse**, then select the WebLogic Portal domain.

This must be the WebLogic domain created earlier. If you are using a two domain configuration, then this must be the WebLogic Portal domain created earlier.

- f) Click **Ok**.
- g) Click **Next**.
- h) Select the EAR file edited previously.
- i) Click **Add**.

This adds the EAR to the list of configured projects.

- j) Click **Next**.
- k) Click **Finish**.

Updating the Portal Web Project

The following procedures show you how to merge the ALBPM WAR file with the WebLogic Portal WAR file. You must manually copy elements from the AquaLogic BPM WAR file to the WebLogic Portal Web Content directory.

During this manual merge process, it is important to remember the source and destination location of the files.

- **Source:** The ALBPM WAR file which was generated earlier in [Creating the ALBPM Application Files](#) on page 40.
- **Destination:** The WebLogic Portal Web Content directory. This directory is usually located at: `BEA_HOME/user_projects/w4WP_workspaces/wlpPortalProject/WebContent`

1. Expand 08-workspace-wlp-FDIDS.war.
2. Copy the following folders and their contents from the source WAR to the Portal Project Web Content folder:
 - charts/*
 - css/*
 - help/*
 - img/*
 - js/*
 - jsf/*
 - jsp/*
3. Copy the `WEB-INF/facelets` folder from the source WAR to the Portal Project WebContent folder.
4. Copy all files from `WEB-INF/classes` to the `WEB-INF/classes` folder in the Portal Project WebContent folder.
5. Copy all the files under `WEB-INF/lib` to the `WEB-INF/lib` folder in the Portal Project WebContent folder.
6. Copy the following files from the source WAR to the `WEB-INF` target in the Portal Project WebContent folder:
 - `/WEB-INF/application.xml`
 - `/WEB-INF/bmpWorkspace.tld`
 - `/WEB-INF/components.xml`
 - `/WEB-INF/directory.xml`

- /WEB-INF/htmlComponents.xml
- /WEB-INF/managed-beans.xml
- /WEB-INF/navigation.xml
- /WEB-INF/workspace.properties

7. Update the WebLogic Portal's web.xml

A revised version of web.xml is available at: http://edocs.bea.com/albsi/docs60/resources/workspace_wlp/web.xml. For a listing of the contents of this file, see [Example web.xml for WebLogic Portal](#) on page 71

Disabling JSP Validation

The following procedures show you how to disable JSP validation.

1. Right-click on the Portal Web Project
2. Click **Properties**.
3. In the left-hand tree, click **Validation**.
4. Unselect the Build column for the JSP Syntax Validator.
5. Click **Apply**.
6. Click **Ok**.

Refreshing the Web Project

To ensure that all of your configuration changes have taken effect, you must refresh and validate the Portal Web Project.

1. Right-click on the Portal Web Project, then select **Refresh**.
This may take a few minutes while the workspace is rebuilt.
2. Right-click on the Portal Web Project, then select **Validate**.

The Portal Web Project should refresh with no errors.

Creating ALBPM Portlets

The following tasks show you how to create ALBPM Portlets using BEA Workshop for WebLogic Platform.

1. Create a new Portlets folder Portal Web Project.
 - a) In the Project Navigator view, expand your Web Portal Project folder
 - b) Right-click the WebContent folder
 - c) Select **New ► Folder**
 - d) Enter **portlets** in the Folder Name field.



Note: This folder must be named **portlets** in all lower case to avoid resource conflicts.

- e) Click **Finish**.
2. Create each of the portlets listed in the following table:


| Portlet | Template Location |
|-----------------|--|
| Actions Menu | /jsf/menu/menuActionPortlet.xhtml |
| Views Menu | /jsf/menu/menuViewPortlet.xhtml |
| Work List | /jsf/view/viewPresentationPortlet.xhtml |
| Instance Detail | /jsf/view/instance/viewInstanceDetailPortlet.xhtml |

To create these portlets, perform the following procedures. You must repeat these procedures for each type of portlet above.

- a) Right-click on the portlets folder created in the previous step.
- b) Select **New ► Portlet**

The **New Portlet** window appears.

- c) Enter the name of your new portlet.

 **Note:** To avoid confusion, you should name your new portlet according to the type mentioned in the table above.

The **Portlet Wizard** appears.

- d) Select **Java Server Faces (JSF) Portlet**.
- e) Click **Next**.
- f) Provide details for your portlet.
 - Title: For simplicity, Provide a title that corresponds to the titles listed in the table above.
 - Content Path: Provide the location of the portlet template. For each portlet type, the template location is listed in the table above. You can copy this location from the table to the Content Path field.
- g) Click **Create**.

The new portlet appears in the editor window. You must repeat these procedures for each type of portlet listed in the table above.

In order to view these portlets, you must install a Login portlet provided by BEA. The portlets created above are added to the login portlet. See [Creating ALBPM Portlets](#) on page 42 for more information.

Creating Login Portlet

1. Download the Login portlet from the Dev2Dev website.

This portlet is available at:

<https://codesamples.projects.dev2dev.bea.com/servlets/Scarab/remcurreport/true/template/ViewIssue.vm/id/S275/nbrresults/47> under the **Attachments** tab.

You should download this portlet and save it to a location where you can import it in the following steps.

2. Import the Login portlet.
 - a) Right-click on the Portal Web Project
 - b) Select **Import ► Import**.
 - c) Expand **General**.
 - d) Select **Archive File**.
 - e) Click **Next**.
 - f) Click **Browse**.
 - g) Select the Login portlet downloaded above.
 - h) Click **Open**.
 - i) Click **Finish**.

3. Expand the Portal Web Project

4. Right-click on the WebContent folder


5. Select **New ► Portal**

The **New Portal** page appears.

6. Provide a name for the new portal.

The new portal appears in the editor window.

7. Set the layout type for the new portlet.

 **Note:** You must ensure that you are not using any of the flow layouts.

a) Right-click in the portlet editor window.

- Select **Layout ► Single Column Layout**

8. Rename the default page to Login in the Properties view.
9. Add the Login portlet imported above to the Login page by dragging it from the Project Navigator view.
10. Right-click in the portal editor window
11. Select **Insert ► New Page**
12. In the Properties view, change the Layout Type to Two Column Layout.
13. In the Properties view, change the Page title to ALBPM.
14. Add the ALBPM Portlets to the ALBPM page by dragging them from the Project Navigator view.
15. Save the Portlet.
16. Republish the server from the Servers view.

After performing the steps above, you should be able to start the WebLogic Portal using the portal and portlets.

Verifying your Installation

This section shows you how to test your configuration of the ALBPM WorkSpace using the WebLogic Portal.

1. Start the WebLogic Server instance configured to run ALBPM and WebLogic Portal.
2. Ensure that your WebLogic Portal and ALBPM user bases are synchronized.
3. Deploy a sample ALBPM Project using the Process Administrator.


ALBPM example projects are located in the Samples directory of you ALBPM installation. See [Deploying and Publishing a New ALBPM Project](#) on page 34 for more information.

4. Open the WebLogic Portal URL

To access the WorkSpace within WebLogic portal, use the following URL:
URL:`http://host:port/your_portal_project/portal_file`

For example: `http://localhost:7001/wlpPortalProject/albpm.portal`

5. Login to the WebLogic Portal using `weblogic` as the username and password.

 **Note:** You must ensure that this user has been created using the ALBPM Process Administrator and has been granted the proper roles.

6. Select the ALBPM Page.

The ALBPM portlets appear containing the activities of the sample project you deployed.

Integrating AquaLogic BPM with AquaLogic Interaction

ALBPM provides out-of-the-box integration using WorkSpace Extensions for AquaLogic User Interaction.

WorkSpace Extensions for ALI allow you to use ALBPM with the following functionality:

- Use ALI as the user interface layer for ALBPM
- Use ALI to handle user authentication
- Use ALI Collaboration as the document repository for attachments.

1. Review [ALI Architecture](#) on page 13
2. Review [Installation Prerequisites](#) on page 7

This topic provides information about the required software you must install before using this guide.

3. Run the WorkSpace Extension configuration wizard.
See [Running the WorkSpace Extension Configuration Wizard](#) on page 45.
4. Configure the ALI Identity Service database.
See [Configuring the ALI Identity Service Database](#) on page 45.
5. Configure the AquaLogic Identity Service.
See [Configuring the AquaLogic Identity Service](#) on page 46
6. Deploy the .pte File
See [Deploying the .pte File](#) on page 47
7. Assign an ALI administrator to the process administrator group.
See [Assigning Process Administrators](#) on page 47
8. Restart the ALI services
See [Restart ALI Services](#) on page 47
9. Verify your installation
See [Verifying Your Installation](#) on page 47

Running the WorkSpace Extension Configuration Wizard

The following procedures show you how to run the WorkSpace Extension Configuration Wizard.

The WorkSpace Extension Configuration Wizard is located at:

BEA_HOME/albpm6.0/enterprise/bin/weconfigwizard.exe

BEA_HOME/albpm6.0/j2eew1/bin/weconfigwizard.exe

BEA_HOME/albpm6.0/j2eews/bin/weconfigwizard.exe .

1. Run the Configuration Wizard as outlined in the following table:


| Configuration Wizard Page | User Input |
|--|--|
| ALUI Database Type | Select the database type used by AquaLogic Interaction Portal. |
| ALUI Database Connection Information | Provide the connectivity information for the AquaLogic Interaction Portal database. |
| ALI Identity Service Database Type and Service Ports | Select the database type and service ports for the AquaLogic Interaction Identity Service (Hydrogen). You will have to create this database after running this Wizard. |
| ALI Identity Service Database Connection Information | Provide the connectivity information for the AquaLogic Interaction Identity Service (Hydrogen) database. |
| Show SQL Script | Copy the generated SQL script to the clipboard or save it to a file. You will use this script later to create the table structures for the ALI Identity Service Database. |
| Enter Image Server URL | Provide the connectivity information to the ALI Image Server. With this information in place ALBPM WorkSpace will leverage ALI Image Server capabilities, improving overall performance. |
| Configuration Finished | Click Finish to close the WorkSpace Extensions configuration wizard. |

The generated SQL script will be used to create the ALI Identity Service (Hydrogen) database.

Configuring the ALI Identity Service Database

This topic describes how to create the AquaLogic Identity Service database.

1. Create the database user and database schema that will contain the ALI Identity Service tables. You will need DBA privileges to perform this step.

 **Note:** These new user id and password you create must match the ones you specified in the **ALI Identity Service Database** step of the WorkSpace Extensions configuration wizard.

For the case of Oracle:

1. You may create the new database user with the following script. Replace *aliis_db* and *password* with the ID and password for the new user you want to create:

```
create user aliis_db identified by password;
grant connect to aliis_db;
grant resource to aliis_db;
grant create view to aliis_db;
```

2. You need to have an Oracle tablespace in place where the ALI Identity Service database indexes will be created. A standard installation of ALI defines a "PLUMINDEX" tablespace; you may reuse it for Identity Service. If you don't have a tablespace available, you may create a new one following this example SQL script (assuming the database SID is "PLUM"):

```
REM * Create a tablespace for indexes.
REM *
CREATE TABLESPACE ALIIS_INDEX
  DATAFILE '$ORACLE_BASE/oradata/PLUM/aliisidx.dbf'
  SIZE 65536K REUSE AUTOEXTEND ON
  EXTENT MANAGEMENT LOCAL AUTOALLOCATE
  SEGMENT SPACE MANAGEMENT AUTO
;
```

On Windows, the DATAFILE path would be:

```
DATAFILE '%ORACLE_BASE%\oradata\PLUM\aliisidx.dbf'
```

3. You must modify the SQL script generated by the WorkSpace Extension Configuration Wizard to specify the right tablespace. In the following line, replace "PROCESSINDEX" with the name of your tablespace:

```
DEF IDX_TBSP=PROCESSINDEX
```

For example, to specify the "PLUMINDEX" tablespace you should change it to:


```
DEF IDX_TBSP=PLUMINDEX
```

2. Run the SQL script generated by the WorkSpace Extensions configuration wizard. You should run the script using the id of the user created in the previous steps to ensure the database objects are created on the right schema and namespace.

After running the SQL script, your database will contain the necessary ALI Identity Service tables.

Configuring the AquaLogic Identity Service

This procedure shows you how to configure the AquaLogic Identity Service for integrating ALBPM with ALUI.

 **Important:** If you have a older version of ALI Identity Service installed on your system, you should un-install it first to avoid conflicts.

To install the ALI Identity Service:

1. Install the ALI Identity Service located in `BEA_HOME/albpm6.0/enterprise/ptids/1.2/bin` `BEA_HOME/albpm6.0/j2eew1/ptids/1.2/bin` `BEA_HOME/albpm6.0/j2eews/ptids/1.2/bin` using the following command:

On Windows: `.\service.bat install`

2. Start the ALI Identity Service.

On Windows you may start the **BEA ALI Identity Service** from the standard Windows Services panel.

On Unix environments, you start the service with the following command:

```
./service.sh start
./service.sh console (to run on the foreground)
```

3. Verify that the ALI Identity Service started without errors.

You may check the log file located at: `BEA_HOME/albpm6.0/enterprise/ptids/1.2/logs/service.log`
`BEA_HOME/albpm6.0/j2eewl/ptids/1.2/logs/service.log`
`BEA_HOME/albpm6.0/j2eews/ptids/1.2/logs/service.log`.

If the service started successfully, there should be no errors and the final line of the log should state `***Initial Sync Completed***`.

Deploying the .pte File

You must import the ALI (.pte) file in order to expose the WorkSpace and Process Administrator applications within the ALI Portal and access the Portal Community that hosts it. This file defines all the associations between ALI and the ALBPM environment.

The .pte file is provided with the ALBPM Enterprise and is located in
`BEA_HOME/albpm6.0/enterprise/serverpackages/ALBPM-60-ALI-template.pte`
`BEA_HOME/albpm6.0/j2eewl/serverpackages/ALBPM-60-ALI-template.pte`
`BEA_HOME/albpm6.0/j2eews/serverpackages/ALBPM-60-ALI-template.pte`.



Note: If you are installing WorkSpace extensions on BEA WebLogic Server or IBM WebSphere Application Server, you must edit the Remote Server objects in the portal to point to the application server. The Remote Server Objects are located in the Remote Server section in the Process folder of the Portal Administration Console.

See the [AquaLogic Interaction Administrator Guide](#) for more information on how to deploy the .pte file.

Assigning Process Administrators

You must add at least one ALI user to the Process Administrator group.

1. Login to ALI's portal as an Administrator. By default, it runs on `http://host:8080/portal/server.pt`.
2. Click on the **Administration** tab.
You should see a new **Process** folder
3. Click on the **Process** folder. Expand **Groups** and click on **Process Administrators**.
4. You may add new members to the group by pressing the **Add User/Group** button.

Restart ALI Services

You must re-start ALI services to force a quick replication of users and groups information from the ALI Portal database to ALI's Identity Service (Hydrogen).

1. Re-start of ALI Services

On Windows you may use the standard Windows Services panel to re-start the following services:

- **BEA ALI API Service**
- **BEA ALI Identity Service**

On UNIX: `BEA_HOME/alui/ptws/6.1/bin/apiserviced.sh restart`

Verifying Your Installation

Follow this procedure to verify the configuration of ALBPM WorkSpace Extensions for ALUI.

1. Start ALBPM Web Applications.

See *Starting BPM Web Applications* The following procedures show you how to start ALBPM Web Applications using the Admin Center. for more information.


2. Verify that the ALBPM Process Administrator is correctly configured.


- a) Login to ALI's Portal as a user in the Process Administrator group. By default, it runs on `http://host:8080/portal/server.pt`.
- b) Click on the **Administration** tab.
- c) Select **Process Administrator** from the **Select Utility...** drop down.
The ALBPM Process Administrator welcome page should appear.
- d) You can go to **Organization > Participants** and **Organization > Groups** to verify that the ALI users appear as ALBPM Participants.

3. Verify that your Process Execution Engine is up and running.

- a) In the Process Administrator click on **Engines**. Verify that the **Status** of your Engine (by default named **albpengine**) is **Running**.

If the Engine status is **Not Running**, try to start it by clicking on the **Start** icon (🔴▶).

 **Note:** If the Engine fails to start, click on the **Start-up log** icon (🔍) to look for errors.

 **Note:** By default, the Process Execution Engine uses TCP port 10099 to accept client connections. If ALBPM is running on the same box as the ALI components, you might need to change this default TCP port to avoid conflicts. To change this port click on **albpengine > Engine Nodes > [your_node_host]** and change the **Port** field.

4. Verify that the Process Portal Community is correctly configured.

- a) Click the **My Communities** drop-down option
- b) Select **Process Community**.

The ALBPM Workspace appears in the main Portal window.

Configuring ALBPM Supporting Applications

The following sections describe how to install and configure ALBPM Enterprise supporting applications.

Process Instance Archiving

AquaLogic BPM allows you to archive information about completed or aborted process instances. After user-specified duration, instance information is moved from the engine database to the archive database. You can use the Archive Viewer application to view instance data once it has been moved to the archive database.

Archiving and Deployment

You must enable archiving when deploying a project to collect all information about the project.

When deploying a project, you can configure the following archiving options:

- Enable archiving
- Archive attachments
- Archive notes

If archiving is enabled globally in Proc Admin, but you do not enable archiving during deployment, old process instances are not archived. You must re-deploy the project to enable archiving. This redeployment requires a project version change.

Configuring Archiving

The following high-level task shows you how to configure process instance archiving, including creating the archive database and enabling archiving using Process Administrator.

1. Create an external resource for the archive database.
2. Enable archiving
Process instance archiving is not enabled by default. You must enable archiving before you can create the archive database.
See [Enabling Archiving](#) on page 49 for more information.
3. Create the archive database
See [Creating the Archive Database](#) on page 50
4. Republish and redeploy your project
If you have any projects that were previously published and deployed, you must republish and redeploy them before their process instances are archived.
5. View process instance archives with the Archive Viewer.
[Launching the Archive Viewer Application](#) on page 50

Enabling Archiving

1. Launch Process Administrator.
2. Select **Engine**.
3. Select the engine where you want to enable archiving.
4. Select **Services**.

5. Click **Enable Archiving** under the **Disposer** section.
6. Select the external resource for your archive database.


If you have not created an external resource for your archive database, click **Create a New Configuration**.

After you have enabled archiving, you can create the archive database using the **Manage Database** page under **Edit Engines**. See [Creating the Archive Database](#) on page 50.

Creating the Archive Database

Before performing the following procedures, ensure that you have enabled archiving. See [Enabling Archiving](#) on page 49 for more information. You should also ensure that you have created an external resource for your archive database server and specified this external resource on the **Services** tab. Process Administrator uses this external resource to connect to the archive database to create the required database tables.

1. Launch Process Administrator
2. Select the engine where you are creating the archive database.

 **Note:** You must create a separate archive database for each engine.
3. Select the **Basic Configuration** tab.
4. Select **Manage Database**.
5. Select **Create the archiving database** and **Create the archiving database structure**.
6. Enter the DBA username and password for your database server.
7. Click **Ok** to create the archive database.

The archive database is created.

Launching the Archive Viewer Application

Before using the Archive Viewer to view process instance information, you must create an database configuration for the Archive Viewer.

1. Start the Admin Center.
2. Click **Configuration**.
3. Select **BPM Web Applications**.
4. Ensure that the Archive Viewer checkbox is enabled
5. Click **Ok**.
6. Click **Start BPM Web Applications**.
7. Click **Launch Archive Viewer**.

The Archive Viewer application starts.


BAM and Process Data Mart

BAM Overview

Business Activity Monitoring (BAM) allows you to store, analyze, and display statistics about your business process execution.

BAM provides information about process instance performance and process workload. This information can be used to present almost real-time business processes metrics. You can then use these to analyze and then improve or adapt business processes based on real-world conditions.

To store and present this information, BAM contains the following:

| | |
|----------------|--|
| Database | BAM data is stored within a database. In ALBPM Studio, this information is stored internally as part of the embedded process execution engine database. In ALBPM Enterprise, you must configure an external database to function as the BAM database. |
| SQL Queries | You can write queries that access the information stored in the BAM database. These queries are contained within a BPM Object Method. When you create a BAM Dashboard using the wizard, the wizard automatically creates queries based on the type of Dashboard template you choose. You can customize these queries or create your own queries and dashboards to customize the way you present BAM. |
| BAM Dashboards | BAM Dashboards allow you to display BAM information in a meaningful and useful way. BAM Dashboards also allow you to drill down from a general view of a process to more specific information such as an order or claim.  Note: BAM Dashboards require the Flash Plugin. |

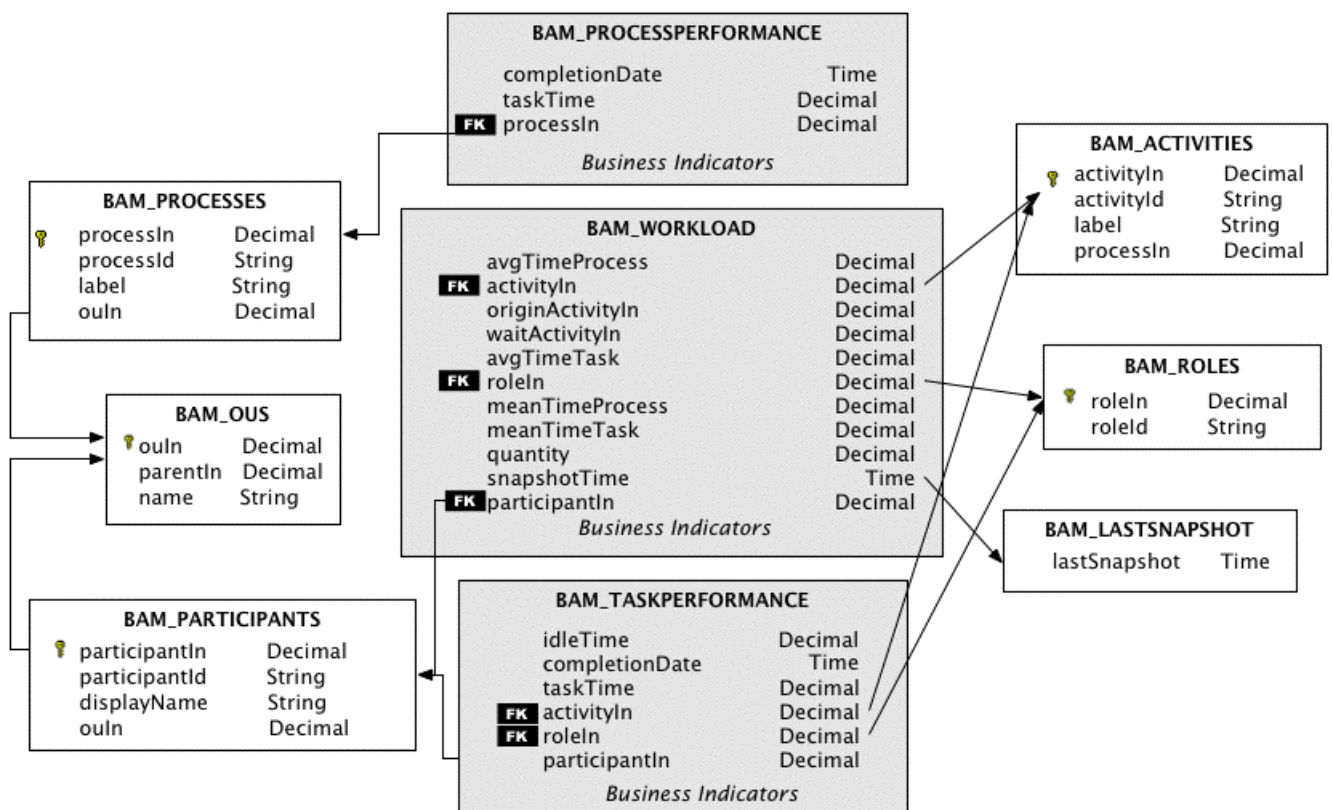
BAM Database

The BAM database is used to store information about your business processes.

The BAM database stores the following types of information about a process:

1. Workload
2. Task Performance
3. Process Performance

The following diagram shows the relationship between each of the BAM tables:



How BAM Database is Populated

BAM database is populated with the information generated by auditing events. Auditing events generation can be enabled for the whole process, for a subset of activities or for a particular activity. For more information on how to configure auditing events generation please see the *ALBPM Enterprise Administration Guide* and the *ALBPM Studio User Guide*.

Using Variables in BAM

When creating a Project variable, you can define it as a Business Indicator variable. This allows the variable to be stored in BAM the database.

When you add Business Indicator variable to your process, a column is added to the following BAM database tables: Workload, Task Performance and Process Performance. The name of this column is the Business Indicator name preceded by the prefix "V_".

If you define a business dimension, the workload table contains one row for each possible value of this business dimension present in the process. Each of this rows will show the quantity of instances that match that business dimension.

When you define a measurement business variable the sum of this variable's value for all in flight is stored into workload table. If business dimensions were defined as well, then this sum will be divided into as many rows as business dimension values present in flight instances.

Task performance table stores one row for each instance that completes an activity. Each of these rows contains the value of dimensions and measurements at the time the instance completed the activity.

In a similar way, process performance table stores one row for each instance that gets to the end activity. Each of these rows contains the value of dimensions and measurements at the time the instance completed the whole process.

BAM Database Reference

The BAM database contains the following tables.

BAM_OU

| Row Name | Value | Null Value |
|----------|-------------|------------|
| ouIn | DECIMAL(10) | NOT NULL |
| parentIn | DECIMAL(10) | NOT NULL |
| name | STRING(255) | NOT NULL |

Primary Key: ouIn

BAM_ROLES

| Row Name | Value | NULL Value |
|----------|-------------|------------|
| roleIn | DECIMAL(10) | NOT NULL |
| roleID | DECIMAL(10) | NOT NULL |

Primary Key: roleIn

BAM_Participants

| Row Name | Value | NULL Value |
|---------------|-------------|------------|
| participantIN | DECIMAL(10) | NOT NULL |
| participantID | STRING(255) | NOT NULL |
| ouIn | DECIMAL(10) | NOT NULL |

| Row Name | Value | NULL Value |
|-------------|-------------|------------|
| displayName | STRING(255) | |

Primary Key: participantIn

Foreign Key: ouin, referenceTable="OUs"

BAM_Processes

| Row Name | Value | NULL Value |
|-----------|-------------|------------|
| ouIn | DECIMAL(10) | NOT NULL |
| processIn | DECIMAL(10) | NOT NULL |
| processId | STRING(255) | NOT NULL |
| label | STRING(255) | NOT NULL |

Primary Key: processIn .

Foreign Key: ouIn, referencedTable="OUs" .

BAM_Activities

| Row Name | Value | NULL Value |
|------------|-------------|------------|
| activityIn | DECIMAL(10) | NOT NULL |
| activityId | STRING(255) | NOT NULL |
| processIN | DECIMAL(10) | NOT NULL |
| label | STRING(255) | |

Primary Key: activityIn .

Foreign Key: processIn, referencedTable="BAM_Processes" .

BAM_Workload

This table contains a record of the number of instances, the average time since the instance was created and the average time waiting for the activityIn to be processed. This information is stored for an activity, role, participant, and if applicable, the activity in a subprocess containing the child instances created by the activityIn.

| Row Name | Value | NULL Value | Description |
|----------------|-------------|------------|---|
| snapshot | TIMESTAMP | NOT NULL | |
| activityIn | DECIMAL(10) | NOT NULL | |
| roleIn | DECIMAL(10) | NOT NULL | |
| participantIn | DECIMAL(10) | NOT NULL | |
| origActivityIn | DECIMAL(10) | NOT NULL | Contains the activityIn (Activity Identification Number) of the subflow or process creation activity that triggers this subprocess. This value is different from only for those activities that are part of a subprocess. |

| Row Name | Value | NULL Value | Description |
|----------------|-------------|------------|---|
| waitActivityIn | DECIMAL(10) | NOT NULL | Within a subflow Process Data Mart and Business Activity Monitoring BAM this activity is the subflow or process creation activity that create the instances taken into account in the current record. |
| quantity | DECIMAL(10) | NOT NULL | Specifies the number of instances in the activity waiting to be processed. |
| avgTimeTask | DECIMAL(10) | NOT NULL | Specifies the average time, in seconds, that an instances waits to be processed within the activity. |
| avgTimeProcess | DECIMAL(10) | NOT NULL | Specifies the average time, in seconds, that an instance was created. |

Foreign Keys:

- activityIn, referencedTable="Activities"
- waitActivityIn, referencedTable="Activities"
- origActivityIn, referencedTable="Activities"
- roleIn, referencedTable="Roles"
- participantIn, referencedTable="Participants"

BAM_TASKPERFORMANCE

This table contains a record for each instance that is processed in the activityIn, roleIn, and participantIn.

| Row Name | Value | NULL Value | |
|----------------|-------------|------------|--|
| activityIn | DECIMAL(10) | NOT NULL | |
| roleIn | DECIMAL(10) | NOT NULL | |
| participantIn | DECIMAL(10) | NOT NULL | |
| completionDate | DECIMAL(10) | NOT NULL | Specifies the date when the instance was processed in the activity and flew to the next activity. To maintain the coherence between the data, the completionDate is stored in GMT-0, as there might be different servers running with different hours. |
| taskTime | DECIMAL(10) | NOT NULL | Specifies the total processing time, in seconds, for the instance in the activity. |

Foreign Keys:

- activityIn, referencedTable="Activities"

- roleIn, referencedTable="Roles"
- participantIn, referencedTable="Participants"

BAM_ProcessPerformance

| Row Name | Value | NULL Value | Description |
|----------------|-------------|------------|---|
| processIn | DECIMAL(10) | NOT NULL | |
| completionDate | TIMESTAMP | NOT NULL | Date when the instance reached the End activity of the process. To maintain the coherence between the data, the completionDate is stored as GMT-0 since there might be different servers running with different hours. |
| taskTime | DECIMAL(10) | NOT NULL | Stores the time, in seconds, required to process the instance. |

Foreign Key: processIn, referencedTable="Processes".

BAM_LASTSNAPSHOT

This table stores a view of the BAM_WorkLoad table, including the time the BAM updater was last executed.

| Row Name | Value | NULL Value |
|--------------|-----------|------------|
| lastshapshot | TIMESTAMP | NOT NULL |

Configuring BAM

The following high-level task outlines the procedures for configuring BAM on ALBPM Enterprise.

1.

Configuring BAM

1.

Creating the BAM Database

1.

Starting the Process Monitoring Service

The Process Monitoring Service works with the process execution engine to store instance data in the Process Data Mart and BAM databases. Both BAM and Process Data Mart use the same service.

The Process Monitoring Service runs as an operating system service and is started separately from ALBPM Enterprise. It is supported on Windows and UNIX.

1. Install the Process Monitoring Service (Windows)

If you are running the Process Monitoring Service on Windows, install the service as follows:

- a) Run `ALBPM_HOME/bin/albpmwarehouse.bat install`
This command installs the Windows service.
- b) Ensure that the `JAVA_HOME` environment variable is set.

2. Start the Process Monitoring Service

- On Windows, you can start the ALBPM 6.0 DataWarehouse Service as a normal Windows Service.
- On UNIX, run `ALBPM_HOME/bin/albpmwarehouse.sh start`

Process Data Mart Overview

Configuring Process Data Mart

The following high-level task outlines the procedures for configuring Process Data Mart on ALBPM Enterprise.

1.

Configuring Process Data Mart

1.

Configuring the Process Data Mart Database

1.

Starting the Process Monitoring Service

The Process Monitoring Service works with the process execution engine to store instance data in the Process Data Mart and BAM databases. Both BAM and Process Data Mart use the same service.

The Process Monitoring Service runs as an operating system service and is started separately from ALBPM Enterprise. It is supported on Windows and UNIX.

1. Install the Process Monitoring Service (Windows)

If you are running the Process Monitoring Service on Windows, install the service as follows:

- a) Run `ALBPM_HOME/bin/albpmwarehouse.bat install`
This command installs the Windows service.
- b) Ensure that the `JAVA_HOME` environment variable is set.

2. Start the Process Monitoring Service

- On Windows, you can start the ALBPM 6.0 DataWarehouse Service as a normal Windows Service.
- On UNIX, run `ALBPM_HOME/bin/albpmwarehouse.sh start`

Reference

The following sections contain general information about ALBPM Enterprise configuration.

Engine and Directory Database Connectivity

The following connectivity properties are required when configuring AquaLogic BPM Enterprise.

These connectivity properties are used when configuring JDBC drivers and directory services. They are used to connect to the following databases:

- Engine Database
- Directory Database
- BAM Database
- Archive Database

BEA Oracle Driver Properties

You can specify the following connectivity properties for your Oracle database:

Basic

| Property | Description |
|-------------------|---|
| Host | Specifies the hostname or IP address of the database server |
| Port | Specifies the TCP port of the Oracle listener running on the Oracle database server. The default is 1521, which is the Oracle default port number when installing the Oracle database software. |
| User | Specifies the case-insensitive default user name used to connect to your Oracle database. |
| Password | Specifies the case-insensitive password used to connect to your Oracle database. |
| SID | Specifies the Oracle System Identifier that refers to the instance of the Oracle database running on the server. |
| Schema (optional) | Specifies the schema of the oracle database server. |
| URL | Defines the URL used to connect to your database. |

Advanced

| Property | Description |
|--------------------------------|-------------|
| Tablespace | |
| Temporary Tablespace | |
| Profile | |
| Use Timestamp for Date Columns | |

Properties

You can define name/value pairs to provide additional configuration properties to your database. See your vendor's documentation for more information.



Note: Connection property names are case-insensitive.

Runtime

| Property | Description |
|--------------------------------|---|
| Maximum Pool Size | Determines the maximum number of connections that can be created within the connection pool. |
| Maximum Connections Per User | Determines the maximum number of connections that can be created per user. |
| Connection Idle Time (minutes) | Specifies the maximum time, in minutes, that a database connection can remain idle before it is closed automatically. |
| Minimum Pool Size | Determines the minimum number of connections that can be created within a connection pool. |
| Maximum Opened Cursors | Determines the maximum number of cursors that can be opened at one time. |

BEA DB2 Driver Properties

You can specify the following connectivity properties for your DB2 database:

Basic

| Property | Description |
|----------|---|
| Host | Specifies the database server host. |
| Port | Specifies the port of the database host. |
| User | Defines user ID you want to use to connect to the database. This user must already exist in DB2 and have permissions to create the schema and tables used to store information. |
| Password | Specifies password for the user. |
| Database | Specifies the database you wish to connect to. |
| Schema | Specifies the database schema to use. (optional) |
| URL | Defines the URL for the database entry. |

Properties

You can define name/value pairs to provide additional configuration properties to your database. See your vendor's documentation for more information.



Note: Connection property names are case-insensitive.

Runtime

| Property | Description |
|--------------------------------|---|
| Maximum Pool Size | Determines the maximum number of connections that can be created within the connection pool. |
| Maximum Connections Per User | Determines the maximum number of connections that can be created per user. |
| Connection Idle Time (minutes) | Specifies the maximum time, in minutes, that a database connection can remain idle before it is closed automatically. |
| Minimum Pool Size | Determines the minimum number of connections that can be created within a connection pool. |
| Maximum Opened Cursors | Determines the maximum number of cursors that can be opened at one time. |

BEA SQL Server Driver Properties

You can specify the following connectivity properties for your SQL Server database:

Basic

| Property | Description |
|----------|---|
| Host | Specifies the hostname or IP address of the database server. |
| Port | The TCP port of the primary database server that is listening for connections to the Microsoft SQL Server database. The default is 1433. |
| User | Specifies the case-insensitive user name used to connect to your Microsoft SQL Server database. |
| Password | Specifies a case-insensitive password used to connect to your Microsoft SQL Server database. |
| Database | Specifies either the IP address or the server name, if your network supports named servers, of the primary database server. |
| URL | Defines the URL format used to connect to your database. |

Properties

You can define name/value pairs to provide additional configuration properties to your database. See your vendor's documentation for more information.

 **Note:** Connection property names are case-insensitive.

Runtime

| Property | Description |
|------------------------------|--|
| Maximum Pool Size | Determines the maximum number of connections that can be created within the connection pool. |
| Maximum Connections Per User | Determines the maximum number of connections that can be created per user. |

| Property | Description |
|--------------------------------|---|
| Connection Idle Time (minutes) | Specifies the maximum time, in minutes, that a database connection can remain idle before it is closed automatically. |
| Minimum Pool Size | Determines the minimum number of connections that can be created within a connection pool. |
| Maximum Opened Cursors | Determines the maximum number of cursors that can be opened at one time. |

ALBPM Configuration Wizard Reference

The following topics provide general information about the AquaLogic BPM Configuration Wizard. They also provide reference information for screen.

What is the ALBPM Configuration Wizard?

The ALBM Configuration Wizard provides a simple way of configuring ALBPM Enterprise.

The Configuration Wizard lets you choose between the following:

- Whether you want to create a new directory provider database or use an existing one. If you choose to create a new database, you can determine one of the following:
 - The Configuration Wizard adds a new database for you.
 - The Configuration Wizard generates SQL scripts that can be run by your DBA.
- Whether to create a new ALBPM Process Execution Engine database or use an existing one. If you choose to create a new database, you can determine one of the following:
 - The Configuration Wizard adds a new database for you.
 - The Configuration Wizard generates SQL scripts that can be run by your DBA.
- Whether to Publish and Deploy a sample project.
- Whether to use a single ALBPM database or a combination database and external directory provider.

After you have created a clustered WLS domain, you can use the ALBPM Configuration Wizard to perform the following tasks related to application server deployment:

- Create the ALBPM Directory database
- Create the ALBPM Process Execution Engine database
- Create the ALBPM web application EAR files
- Create a JDBC data source
- Create a JMS server and queue
- Deploy the ALBPM web applications to a server

Running the ALBPM Configuration Wizard

The following procedures show you how to use the Configuration Wizard. The exact path depends on the options you choose.

1. Determine what tasks you want the configuration Wizard to perform.

This allows you to define the tasks performed. See [Configuration Wizard Tasks](#) on page 61 .

2. Determine if you want to use a database only or a hybrid.

See [Directory Provider Type](#) on page 62


3. Enter information about your directory provider type.

See [Directory Provider Selection](#) on page 62 . If you have chosen to implement the

4. Enter connectivity information about your database.

| Provider | More information |
|------------|--|
| Oracle | Configure Directory Provider - Oracle on page 63 |
| DB2 | Configure Directory Provider - DB2 on page 64 |
| SQL Server | Configure Directory Provider - SQL Server on page 64 |

5. Enter connectivity information for you external organization provider.

 **Note:** This page appears only when you have chosen to configure an external directory service for your organizational data.

6. Enter one of the following:

- If you have selected to have the Configuration Wizard create the database, provide information DBA username and password.
- If you have selected to generate SQL script, enter org log name.

7. If you have selected to generate SQL scripts, you can choose to save it as a file or you can copy and paste it to a file.

8. Select the type of database you want to use for your Process Execution Engine database

9. Enter connectivity information for you Process Execution Engine database

10. If you have selected to have the Configuration Wizard create the database, provide information DBA username and password.

11. If you have selected to generate SQL scripts, you can choose to save it as a file or you can copy and paste it to a file.

12. Select the EAR files you want to create and deploy

13. Provide connectivity information for your WebLogic Server Installation

ALBPM Configuration Wizard Reference


The following topics provide detailed information for each page of the Configuration Wizard

Configuration Wizard Tasks

This page allows you to specify the tasks performed by the Configuration Wizard. These tasks can be groups according to the following:

- Configure the ALBPM Directory Service Database.
- Configure the ALBPM Process Engine Database.
- Publish and deploy a sample project.
- Generate the EAR files for each ALBPM application.
- Create a new single-node WebLogic Server domain


You can determine which of these the Configuration Wizard performs by selecting from the following options:

| Option | Description |
|--------------------------------|---|
| Create Directory Service | Allows you to create a new ALBPM Directory Service |
| Use Existing Directory Service | Allows you to use an existing ALBPM Directory Service.  Note: If you choose to use an existing directory service, it must be created using the current version of AquaLogic BPM. |

| Option | Description |
|--|---|
| Generate Directory Service SQL Script only | Generates a SQL script that can be used to configure the database server and schema. Use this option if you do not have DBA permissions on the database. The generated scripts can be used by your DBA. |
| Create Process Engine | Allows you to create the Process Execution Engine database using the Configuration Wizard. |
| Generate Process Engine SQL Script only | Generates a SQL script that can be used to configure the database server and schema. Use this option if you do not have DBA permissions on the database. The generated scripts can be used by your DBA. |
| Publish and Deploy Sample Project | Allows the Configuration Wizard to publish and deploy ALBPM sample projects. This option is only available if you select to create the Process Execution Engine database. |
| Create ALBPM Applications EAR Files | Generates the EAR files for each ALBPM Enterprise application. |
| BEA WebLogic Configuration | Configures WebLogic Server and deploys the ALBPM Enterprise applications. |




Directory Provider Type

This page allows you to specify how the directory provider is configured.

| Option | Description |
|---|---|
| Use a database managed by ALBPM | Uses only a database to store directory and process information. |
| Use an external directory service provider plus a database managed by ALBPM | Uses both a database and directory service provider to store directory and process information.  Note: When configuring WorkSpace Extensions, you must choose this option to select the AquaLogic Interaction Identity Service (Hydrogen). |


Directory Provider Selection

This page allows you to define the general information about the Directory Service Provider.

| Option | Description |
|------------------------------|--|
| Directory Configuration Name | Defines a label used to refer to this configuration within Admin Center. |
| Description | Provides a useful description for the new directory service configuration. |
| Directory Provider | Specifies the database driver for the database used as directory provider. |
| Organization Provider | Defines the type of external directory service provider.  Note: When configuring WorkSpace Extensions, you must select the ALI Identity Service (Hydrogen). |
| BPM Administrator User | Defines the BPM Administrator user ID.  Note: The Configuration Wizard creates this new user ID in the database.  Note: When configuring WorkSpace Extensions, this administrator ID must also be defined in the AquaLogic Integration Portal. |
| BPM Administrator Password | Defines the BPM Administrator password. |

Configure Directory Provider - Oracle

Basic Tab

| Option | Description |
|----------|---|
| Host | Specifies either the IP address or the hostname of the database server. |
| Port | Specifies the TCP port of the listener running on the database server. The default is 1521, which is the default port number when installing the Oracle database software. |
| User | Defines the user id used to connect to the database. |
| Password | Defines the password for the user id used to connect to the database. This password is case-insensitive. |
| SID | Specifies the Oracle System Identifier that refers to the instance of the Oracle database running on the server. |
| Schema | Optionally, specifies the name of the schema used.  Note: When using a schema, it is recommended that the schema name and user name be the same. |
| URL | Defines the URL format to connect to the database. |

Advanced Tab

| Option | Description |
|----------------------|---|
| Tablespace | Specifies the tablespace within the ALBPM database. |
| Temporary Tablespace | Specifies the temporary tablespace within the ALBPM database. |
| Profile | Specifies the profile for the ALBPM database. Profiles are as a way to limit which users can connect to the database. |

Properties

This tab allows you to specify additional properties that are supported by the JDBC driver. Properties are defined using name/value pairs.

Runtime

The following runtime configuration properties can be defined for this database:

| Option | Description |
|--------------------------------|--|
| Maximum Pool Size | Specifies the maximum number of connections within the connection pool. Database connections are grouped together within a connection pool to improve performance. |
| Maximum connections per user | Specifies the maximum number of connections within the connection pool for each user. |
| Connection Idle Time (minutes) | Specifies how long a connection can be idle before it times out. |
| Minimum Pool Size | Specifies the minimum number of connections with the connection pool. |
| Maximum Opened Cursors | Specifies how many queries can be created for each connection. |

These properties define the JDBC connection handling for ALBPM components when running outside a J2EE (for example, WebLogic Server). When running within a J2EE container, database connectivity information is supplied by the container itself.

Configure Directory Provider - DB2

This page defines the connection properties for the ALBPM directory database.

Basic Tab

| Option | Description |
|----------|--|
| Host | Specifies either the IP address or the hostname of the database server. For example, 122.23.15.12 or ALBPMDatabase. |
| Port | Specifies the TCP port of the listener running on the database server. The default is 1521, which is the default port number when installing the Oracle database software. |
| User | Defines the user id used to connect to the database. |
| Password | Defines the password for the user id used to connect to the database. This password is case-insensitive. |
| Database | Defines the name of the database used for the ALBPM directory service. |
| URL | Defines the URL format to connect to the database. |

Properties Tab

This tab allows you to specify additional properties that are supported by the JDBC driver. Properties are defined using name/value pairs.

Runtime

The following runtime configuration properties can be defined for this database:

| Option | Description |
|--------------------------------|--|
| Maximum Pool Size | Specifies the maximum number of connections within the connection pool. Database connections are grouped together within a connection pool to improve performance. |
| Maximum connections per user | Specifies the maximum number of connections within the connection pool for each user. |
| Connection Idle Time (minutes) | Specifies how long a connection can be idle before it times out. |
| Minimum Pool Size | Specifies the minimum number of connections with the connection pool. |
| Maximum Opened Cursors | Specifies how many queries can be created for each connection. |

These properties define the JDBC connection handling for ALBPM components when running outside a J2EE (for example, WebLogic Server). When running within a J2EE container, database connectivity information is supplied by the container itself.

Configure Directory Provider - SQL Server

Basic Tab

| Option | Description |
|----------|---|
| Host | Specifies either the IP address or the hostname of the database server. |
| Port | Specifies the port of the listener running on the database server. |
| User | Defines the user id used to connect to the database. |
| Password | Defines the password for the user id used to connect to the database. |
| Database | Specifies the name of the database. |

| Option | Description |
|--------|--|
| URL | Defines the URL format to connect to the database. |

Properties Tab

This tab allows you to specify additional properties that are supported by the JDBC driver. Properties are defined using name/value pairs.

Runtime

| Option | Description |
|--------------------------------|--|
| Maximum Pool Size | Specifies the maximum number of connections within the connection pool. Database connections are grouped together within a connection pool to improve performance. |
| Maximum connections per user | Specifies the maximum number of connections within the connection pool for each user. |
| Connection Idle Time (minutes) | Specifies how long a connection can be idle before it times out. |
| Minimum Pool Size | Specifies the minimum number of connections with the connection pool. |
| Maximum Opened Cursors | Specifies how many queries can be created for each connection. |

Configure Organization Provider - Active Directory

Basic Tab

| Options | |
|-------------------------|---|
| Initial Context Factory | Specifies the JNDI class name that creates sessions with the directory service. |
| URL | Specifies the URL used to connect to the directory service. |
| Principal | Specifies the user id to connect to the directory service. |
| Credentials | Specifies the password used to connect to the directory service. |

Properties

This tab allows you to specify additional properties that are supported by the JDBC driver. Properties are defined using name/value pairs.

Runtime Tab

| Option | Description |
|--------------------------------|--|
| Maximum Pool Size | Specifies the maximum number of connections within the connection pool. Database connections are grouped together within a connection pool to improve performance. |
| Maximum connections per user | Specifies the maximum number of connections within the connection pool for each user. |
| Connection Idle Time (minutes) | Specifies how long a connection can be idle before it times out. |
| Minimum Pool Size | Specifies the minimum number of connections with the connection pool. |
| Maximum Opened Cursors | Specifies how many queries can be created for each connection. |

Configure Organization Provider - Hydrogen

This page displays configuration information used by the ALI Identity Service (Hydrogen).

Basic Tab

| Option | Description |
|-----------------------|---|
| Hydrogen Service Host | Specifies the host where the ALI Identity Service (Hydrogen) is deployed. |
| Hydrogen Service Port | Specifies the port where the ALI Identity Service is listening. |
| Service Endpoint URL | Specifies the URL of the ALI Portal Query Service. |

Runtime Tab

| Option | Description |
|--------------------------------|--|
| Maximum Pool Size | Specifies the maximum number of connections within the connection pool. Database connections are grouped together within a connection pool to improve performance. |
| Maximum connections per user | Specifies the maximum number of connections within the connection pool for each user. |
| Connection Idle Time (minutes) | Specifies how long a connection can be idle before it times out. |
| Minimum Pool Size | Specifies the minimum number of connections with the connection pool. |
| Maximum Opened Cursors | Specifies how many queries can be created for each connection. |

Configure Organization Provider - Sun LDAP

Basic Tab

| Options | |
|-------------------------|---|
| Initial Context Factory | Specifies the JNDI class name that creates sessions with the directory service. |
| URL | Specifies the URL used to connect to the directory service. |
| Principal | Specifies the user id to connect to the directory service. |
| Credentials | Specifies the password used to connect to the directory service. |

Properties

This tab allows you to specify additional properties that are supported by the JDBC driver. Properties are defined using name/value pairs.

Runtime Tab

| Option | Description |
|--------------------------------|--|
| Maximum Pool Size | Specifies the maximum number of connections within the connection pool. Database connections are grouped together within a connection pool to improve performance. |
| Maximum connections per user | Specifies the maximum number of connections within the connection pool for each user. |
| Connection Idle Time (minutes) | Specifies how long a connection can be idle before it times out. |

| Option | Description |
|------------------------|---|
| Minimum Pool Size | Specifies the minimum number of connections with the connection pool. |
| Maximum Opened Cursors | Specifies how many queries can be created for each connection. |

Enter Directory Creation Information

The page allows you to enter the database administrator username and password. The Configuration Wizard uses this information to connect to the database and create run the SQL scripts to create database schema.



Note: This page does not appear if you have chosen to generate SQL scripts.

| Option | Description |
|---------------------------|---|
| DBA User | Specifies the Database Administrator user id. |
| DBA Password | Specifies the Database Administrator password. |
| Organization Logical Name | Defines the organization logical name for this ALBPM environment. This symbolic name is used in contexts where processes in different environments communicate with each other. |

Show SQL Script

The page displays the generated SQL scripts for the ALBPM directory database. This page is displayed only if you have chosen to generate a SQL script instead of having the Configuration Wizard connect to the database. The DBA of the directory service database can use the script to create the necessary tables and schema.

| Option | Description |
|-------------------|---|
| Copy to clipboard | Allows you to copy the generated script to the clipboard. |
| Save to file | Allows you to save the generated script to a file. |

Process Engine Provider Selection

| Option | Description |
|--------------------------|---|
| Engine Database Provider | Specifies the database vendor of you Process Engine database. |

Process Engine Provider - Oracle

Basic Tab

| Option | Description |
|---|--|
| Host | Specifies either the IP address or the hostname of the database server. |
| Port | Specifies the TCP port of the listener running on the database server. The default is 1521, which is the default port number when installing the Oracle database software. |
| User | Defines the user id used to connect to the database. |
| Password | Defines the password for the user id used to connect to the database. This password is case-insensitive. |
| SID | Specifies the Oracle System Identifier that refers to the instance of the Oracle database running on the server. |
| Schema | Optionally, specifies the name of the schema used. |
| Note: When using a schema, it is recommended that the schema name and user name be the same. | |

| Option | Description |
|--------|--|
| URL | Defines the URL format to connect to the database. |

Advanced Tab

| Option | Description |
|----------------------|---|
| Tablespace | Specifies the tablespace within the ALBPM database. |
| Temporary Tablespace | Specifies the temporary tablespace within the ALBPM database. |
| Profile | Specifies the profile for the ALBPM database. Profiles are as a way to limit which users can connect to the database. |

Properties

This tab allows you to specify additional properties that are supported by the JDBC driver. Properties are defined using name/value pairs.

Runtime

| Option | Description |
|--------------------------------|--|
| Maximum Pool Size | Specifies the maximum number of connections within the connection pool. Database connections are grouped together within a connection pool to improve performance. |
| Maximum connections per user | Specifies the maximum number of connections within the connection pool for each user. |
| Connection Idle Time (minutes) | Specifies how long a connection can be idle before it times out. |
| Minimum Pool Size | Specifies the minimum number of connections with the connection pool. |
| Maximum Opened Cursors | Specifies how many queries can be created for each connection. |

Process Engine Provider - DB2

Basic Tab

| Option | Description |
|----------|--|
| Host | Specifies either the IP address or the hostname of the database server. For example, 122.23.15.12 or ALBPMDatabase. |
| Port | Specifies the TCP port of the listener running on the database server. The default is 1521, which is the default port number when installing the Oracle database software. |
| User | Defines the user id used to connect to the database. |
| Password | Defines the password for the user id used to connect to the database. This password is case-insensitive. |
| Database | Defines the name of the database used for the ALBPM directory service. |
| URL | Defines the URL format to connect to the database. |

Properties

This tab allows you to specify additional properties that are supported by the JDBC driver. Properties are defined using name/value pairs.

Runtime

| Option | Description |
|--------------------------------|--|
| Maximum Pool Size | Specifies the maximum number of connections within the connection pool. Database connections are grouped together within a connection pool to improve performance. |
| Maximum connections per user | Specifies the maximum number of connections within the connection pool for each user. |
| Connection Idle Time (minutes) | Specifies how long a connection can be idle before it times out. |
| Minimum Pool Size | Specifies the minimum number of connections with the connection pool. |
| Maximum Opened Cursors | Specifies how many queries can be created for each connection. |

Process Engine Provider - SQL Server

Basic Tab

| Option | Description |
|----------|---|
| Host | Specifies either the IP address or the hostname of the database server. |
| Port | Specifies the port of the listener running on the database server. |
| User | Defines the user id used to connect to the database. |
| Password | Defines the password for the user id used to connect to the database. |
| Database | Specifies the name of the database. |
| URL | Defines the URL format to connect to the database. |

Properties

This tab allows you to specify additional properties that are supported by the JDBC driver. Properties are defined using name/value pairs.

Runtime

| Option | Description |
|--------------------------------|--|
| Maximum Pool Size | Specifies the maximum number of connections within the connection pool. Database connections are grouped together within a connection pool to improve performance. |
| Maximum connections per user | Specifies the maximum number of connections within the connection pool for each user. |
| Connection Idle Time (minutes) | Specifies how long a connection can be idle before it times out. |
| Minimum Pool Size | Specifies the minimum number of connections with the connection pool. |
| Maximum Opened Cursors | Specifies how many queries can be created for each connection. |

Enter Process Engine Creation Information

| Option | Description |
|--------------|--|
| DBA User | Specifies the Database Administrator user id. |
| DBA Password | Specifies the Database Administrator password. |

Show SQL Script

This page displays the generated SQL script which is used by your database administrator to create the necessary tables and schema.


| Option | Description |
|-------------------|---|
| Copy to clipboard | Allows you to copy the generated script to the clipboard. |
| Save to file | Allows you to save the generated script to a file. |

Select EAR Files to Create - WebLogic Server

This page allows you to select which application EAR files you want to create and deploy within your WebLogic Server installation.

| Option | Description |
|-----------------------------|--|
| EAR Files Folder | Specifies the directory where the generated WAR files are saved. |
| Process Engine EAR | |
| WorkSpace EAR | |
| WorkSpace Administrator EAR | |
| RSS Feeds EAR | |
| PAPI Web Services EAR | |
| ALSB/ALBPM Transport EAR | |

Configure WebLogic Server

| Option | Description |
|---------------------------|--|
| Create New Domain | Causes the configuration wizard to create a new WebLogic Server domain. |
| Modify an Existing Domain | Causes the configuration wizard to configure and deploy to an existing WebLogic Server domain.  Note: If you are using the configuration wizard to deploy to a clustered domain, you must select this option. |
| WebLogic Home | Specifies the root directory of your WebLogic Server installation. |
| Hostname | Specifies the hostname or IP address of the server where you want to deploy the ALBPM applications. |
| Port | Specifies the port number of the server where you want to deploy the ALBPM applications. |
| Server Name | |
| Domain Name | |
| Domain Folder | |
| WebLogic Admin | Specifies the WebLogic Server administrator ID. |
| WebLogic Password | Specifies the WebLogic Server administrator password. |

Configuring ALBPM

This page appears while the ALBPM Configuration Wizard is running. It may take several minutes for this to complete.

ALBPM Configuration Complete

This page appears after the ALBPM Configuration Wizard is complete. Click **Finish** to close the configuration wizard.

Example web.xml for WebLogic Portal

When configuring AquaLogic BPM to work with WebLogic Portal you must use the following web.xml as part of your WebLogic Portal configuration.

This file can be downloaded at: http://edocs.bea.com/albsi/docs60/resources/workspace_wlp/web.xml. See *WebLogic Portal Configuration* for more information.

```
<?xml version="1.0" encoding="UTF-8"?>
<web-app id="WebApp_ID" version="2.4" xmlns="http://java.sun.com/xml/ns/j2ee"
xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
xsi:schemaLocation="http://java.sun.com/xml/ns/j2ee
http://java.sun.com/xml/ns/j2ee/web-app_2_4.xsd">
  <display-name>WorkSpaceWebProject</display-name>
  <welcome-file-list>
    <welcome-file>index.html</welcome-file>
    <welcome-file>index.htm</welcome-file>
    <welcome-file>index.jsp</welcome-file>
    <welcome-file>default.html</welcome-file>
    <welcome-file>default.htm</welcome-file>
    <welcome-file>default.jsp</welcome-file>
  </welcome-file-list>
  <context-param>
    <param-name>javax.faces.STATE_SAVING_METHOD</param-name>
    <param-value>server</param-value>
  </context-param>
  <context-param>
    <param-name>com.sun.faces.validateXml</param-name>
    <param-value>true</param-value>
  </context-param>
  <context-param>
    <param-name>com.sun.faces.verifyObjects</param-name>
    <param-value>true</param-value>
  </context-param>
  <servlet>
    <servlet-name>Faces Servlet</servlet-name>
    <servlet-class>javax.faces.webapp.FacesServlet</servlet-class>
  </servlet>
  <filter>
    <filter-name>PageFlowJspFilter</filter-name>
    <filter-class>org.apache.beehive.netui.pageflow.PageFlowJspFilter</filter-class>
  </filter>
  <filter>
    <filter-name>PageFlowForbiddenFilter</filter-name>
    <filter-class>org.apache.beehive.netui.pageflow.PageFlowForbiddenFilter</filter-class>

    <init-param>
      <param-name>response-code</param-name>
      <param-value>404</param-value>
    </init-param>
  </filter>
  <filter>
    <filter-name>Character Encoding</filter-name>
    <filter-class>fuego.web.filter.CharsetFilter</filter-class>
  </filter>
  <filter>
    <filter-name>SingleThreadPerSessionFilter</filter-name>
    <filter-class>fuego.web.filter.SingleThreadPerSessionFilter</filter-class>
    <init-param>
      <param-name>enabled</param-name>
      <param-value>true</param-value>
    </init-param>
  </filter>
</web-app>
```

```

<filter>
  <filter-name>CacheResourcesFilter</filter-name>
  <filter-class>fuego.web.filter.CacheResourceFilter</filter-class>
  <init-param>
    <param-name>enabled</param-name>
    <param-value>true</param-value>
  </init-param>
</filter>
<filter-mapping>
  <filter-name>CacheResourcesFilter</filter-name>
  <url-pattern>*.js</url-pattern>
</filter-mapping>
<filter-mapping>
  <filter-name>CacheResourcesFilter</filter-name>
  <url-pattern>*.css</url-pattern>
</filter-mapping>
<filter-mapping>
  <filter-name>CacheResourcesFilter</filter-name>
  <url-pattern>*.gif</url-pattern>
</filter-mapping>
<filter-mapping>
  <filter-name>CacheResourcesFilter</filter-name>
  <url-pattern>*.png</url-pattern>
</filter-mapping>
<filter-mapping>
  <filter-name>SingleThreadPerSessionFilter</filter-name>
  <url-pattern>*.xhtml</url-pattern>
</filter-mapping>
<filter-mapping>
  <filter-name>SingleThreadPerSessionFilter</filter-name>
  <url-pattern>/servlet/*</url-pattern>
</filter-mapping>
<filter-mapping>
  <filter-name>Character Encoding</filter-name>
  <url-pattern>/*</url-pattern>
</filter-mapping>
<filter-mapping>
  <filter-name>PageFlowJspFilter</filter-name>
  <url-pattern>*.jsp</url-pattern>
  <dispatcher>FORWARD</dispatcher>
  <dispatcher>REQUEST</dispatcher>
  <dispatcher>INCLUDE</dispatcher>
</filter-mapping>
<filter-mapping>
  <filter-name>PageFlowJspFilter</filter-name>
  <url-pattern>*.jspx</url-pattern>
  <dispatcher>FORWARD</dispatcher>
  <dispatcher>REQUEST</dispatcher>
  <dispatcher>INCLUDE</dispatcher>
</filter-mapping>
<filter-mapping>
  <filter-name>PageFlowForbiddenFilter</filter-name>
  <url-pattern>*.java</url-pattern>
  <dispatcher>REQUEST</dispatcher>
</filter-mapping>
<filter-mapping>
  <filter-name>PageFlowForbiddenFilter</filter-name>
  <url-pattern>*.jsfb</url-pattern>
  <dispatcher>REQUEST</dispatcher>
</filter-mapping>
<filter-mapping>
  <filter-name>PageFlowForbiddenFilter</filter-name>
  <url-pattern>*.jpfs</url-pattern>
  <dispatcher>REQUEST</dispatcher>
</filter-mapping>
<listener>
<listener-class>org.apache.beehive.netui.pageflow.PageFlowContextListener</listener-class>

</listener>
</listener>

```



```

<listener-class>org.apache.beehive.netui.pageflow.HttpSessionMutexListener</listener-class>

</listener>
</listener>

<listener-class>fuego.workspace.execution.WorkspaceExecutionSessionListener</listener-class>

</listener>
<listener>
  <listener-class>fuego.workspace.context.WorkspaceContextListener</listener-class>
</listener>
<listener>
  <listener-class>com.sun.faces.config.ConfigureListener</listener-class>
</listener>
<servlet>
  <servlet-name>action</servlet-name>
  <servlet-class>org.apache.beehive.netui.pageflow.PageFlowActionServlet</servlet-class>

  <init-param>
    <param-name>config</param-name>
    <param-value>/_pageflow/struts-config.xml</param-value>
  </init-param>
  <init-param>
    <param-name>debug</param-name>
    <param-value>2</param-value>
  </init-param>
  <init-param>
    <param-name>detail</param-name>
    <param-value>2</param-value>
  </init-param>
  <load-on-startup>2</load-on-startup>
</servlet>
<servlet>
  <servlet-name>XmlHttpRequestServlet</servlet-name>

<servlet-class>org.apache.beehive.netui.pageflow.xmlhttprequest.XmlHttpRequestServlet</servlet-class>

</servlet>
<!-- Executor Servlet -->
<servlet>
  <servlet-name>ExecutorServlet</servlet-name>
  <servlet-class>fuego.workspace.servlet.ExecutorServlet</servlet-class>
  <load-on-startup>1</load-on-startup>
</servlet>
<!-- Image Servlet -->
<servlet>
  <servlet-name>ImageViewer</servlet-name>
  <servlet-class>fuego.workspace.servlet.ImageViewer</servlet-class>
  <load-on-startup>1</load-on-startup>
</servlet>
<!-- Instance Detail Servlet -->
<servlet>
  <servlet-name>Controller</servlet-name>
  <servlet-class>fuego.workspace.servlet.Controller</servlet-class>
  <load-on-startup>1</load-on-startup>
</servlet>
<!-- Download Attachment Servlet -->
<servlet>
  <servlet-name>DownloadAttachmentServlet</servlet-name>
  <servlet-class>fuego.workspace.servlet.DownloadAttachmentServlet</servlet-class>
  <load-on-startup>1</load-on-startup>
</servlet>
<servlet-mapping>
  <servlet-name>action</servlet-name>
  <url-pattern>*.jpg</url-pattern>
</servlet-mapping>
<servlet-mapping>
  <servlet-name>action</servlet-name>
  <url-pattern>*.do</url-pattern>
</servlet-mapping>

```

```

<servlet-mapping>
  <servlet-name>XmlHttpRequestServlet</servlet-name>
  <url-pattern>*.xhr</url-pattern>
</servlet-mapping>
<servlet-mapping>
  <servlet-name>XmlHttpRequestServlet</servlet-name>
  <url-pattern>*.render</url-pattern>
</servlet-mapping>
<!-- extension mapping -->
<servlet-mapping>
  <servlet-name>ExecutorServlet</servlet-name>
  <url-pattern>/servlet/executor</url-pattern>
</servlet-mapping>
<servlet-mapping>
  <servlet-name>ImageViewer</servlet-name>
  <url-pattern>/servlet/image</url-pattern>
</servlet-mapping>
<servlet-mapping>
  <servlet-name>Controller</servlet-name>
  <url-pattern>/servlet/controller</url-pattern>
</servlet-mapping>
<servlet-mapping>
  <servlet-name>DownloadAttachmentServlet</servlet-name>
  <url-pattern>/servlet/download</url-pattern>
</servlet-mapping>
<!-- Use Documents Saved as *.xhtml
-->
<context-param>
  <param-name>javax.faces.DEFAULT_SUFFIX</param-name>
  <param-value>.xhtml</param-value>
</context-param>
<context-param>
  <param-name>com.sun.faces.NUMBER_OF_VIEWS_IN_SESSION</param-name>
  <param-value>15</param-value>
</context-param>
<context-param>
  <param-name>fuego.upload.FILE_MAX_SIZE</param-name>
  <param-value>1000000</param-value>
</context-param>
<context-param>
  <param-name>facelets.DEVELOPMENT</param-name>
  <param-value>>false</param-value>
</context-param>
<context-param>
  <param-name>facelets.VIEW_MAPPINGS</param-name>
  <param-value>*.xhtml</param-value>
</context-param>
<context-param>
  <param-name>facelets.REFRESH_PERIOD</param-name>
  <param-value>-1</param-value>
</context-param>
<context-param>
  <param-name>facelets.LIBRARIES</param-name>
  <param-value>
    /WEB-INF/facelets/bpmWorkspace.taglib.xml;
    /WEB-INF/facelets/bpmWorkspaceLibrary.taglib.xml;
    /WEB-INF/facelets/fuegojsf.taglib.xml;
    /WEB-INF/facelets/htmlib.taglib.xml;
  </param-value>
</context-param>
<context-param>
  <param-name>javax.faces.CONFIG_FILES</param-name>
  <param-value>
    /WEB-INF/application.xml,/WEB-INF/navigation.xml,/WEB-INF/managed-beans.xml,/WEB-INF/components.xml

  </param-value>
</context-param>
<context-param>
  <param-name>com.bea.opencontrols.RESOURCE_SUFFIX</param-name>
  <param-value>.resource</param-value>

```

```

</context-param>
<context-param>
  <param-name>com.bea.opencontrols.WLP</param-name>
  <param-value>true</param-value>
</context-param>
<filter>
  <filter-name>PageFlowFacesFilter</filter-name>
  <filter-class>org.apache.beehive.netui.pageflow.PageFlowFacesFilter</filter-class>
</filter>
<filter-mapping>
  <filter-name>PageFlowFacesFilter</filter-name>
  <url-pattern>*.faces</url-pattern>
  <dispatcher>FORWARD</dispatcher>
  <dispatcher>REQUEST</dispatcher>
  <dispatcher>INCLUDE</dispatcher>
</filter-mapping>
<filter-mapping>
  <filter-name>PageFlowFacesFilter</filter-name>
  <url-pattern>*.jsf</url-pattern>
  <dispatcher>FORWARD</dispatcher>
  <dispatcher>REQUEST</dispatcher>
  <dispatcher>INCLUDE</dispatcher>
</filter-mapping>
<servlet-mapping>
  <servlet-name>Faces Servlet</servlet-name>
  <url-pattern>*.faces</url-pattern>
</servlet-mapping>
<servlet-mapping>
  <servlet-name>Faces Servlet</servlet-name>
  <url-pattern>*.resource</url-pattern>
</servlet-mapping>
<servlet-mapping>
  <servlet-name>Faces Servlet</servlet-name>
  <url-pattern>/faces/*</url-pattern>
</servlet-mapping>
<servlet-mapping>
  <servlet-name>Faces Servlet</servlet-name>
  <url-pattern>*.xhtml</url-pattern>
</servlet-mapping>
<error-page>
  <exception-type>java.lang.Exception</exception-type>
  <location>/jsp/errorPage.jsp</location>
</error-page>
<error-page>
  <error-code>500</error-code>
  <location>/jsp/errorPage.jsp</location>
</error-page>
<jsp-config>
  <jsp-property-group>
    <url-pattern>*.jsf</url-pattern>
    <is-xml>false</is-xml>
  </jsp-property-group>
  <jsp-property-group>
    <url-pattern>*.jsp</url-pattern>
    <url-pattern>*.jspx</url-pattern>
    <el-ignored>true</el-ignored>
  </jsp-property-group>
</jsp-config>
<mime-mapping>
  <extension>png</extension>
  <mime-type>image/png</mime-type>
</mime-mapping>
</web-app>

```