



# **AquaLogic BPM Enterprise Administration Guide**

**Version: 6.0**

# Contents

<b>Introduction.....</b>	<b>5</b>
Document Scope and Audience.....	5
BEA Documentation and Resources.....	5
<b>Getting Started.....</b>	<b>7</b>
Administration Prerequisites.....	7
ALBPM Administration Tools Overview.....	7
ALBPM Enterprise Administration Roadmap.....	7
<b>Managing the Lifecycle of an ALBPM Project.....</b>	<b>9</b>
Accessing Process Administrator.....	9
Managing Your Organization.....	9
Organization Overview.....	9
Permissions and Security in an Organization.....	13
Creating an Organization Using Process Administrator.....	15
ALBPM Directory Service.....	17
Managing Projects.....	20
Deploying and Publishing a New ALBPM Project.....	20
Undeploying a Project.....	21
Project Versioning.....	22
Managing Process Execution Engines.....	23
Engine Status.....	23
Engine Disposer Service.....	23
Creating a Process Execution Engine.....	23
Starting and Stopping Process Execution Engines.....	24
Exporting and Importing an Engine Configuration.....	24
Tuning Engine Database Runtime Performance.....	25
Determining Engine Memory Allocation (Standalone).....	26
Project Variables on ALBPM Enterprise.....	27
Creating Variables in ALBPM Enterprise.....	28
Localizing Variables Using Process Administrator.....	28
Logging.....	28
ALBPM Log Files.....	28
Log Severity Levels.....	30
Process Execution Engine Log.....	30
External Resources.....	32
Creating an External Resource Using Process Administrator.....	32
External Resource Reference.....	33
<b>Advanced Administration.....</b>	<b>52</b>
Auditing.....	52
Enabling Auditing Using Process Administrator.....	52
Which Audit Events are Generated.....	52
ALBPM Backup Guidelines.....	53

Security.....	53
Best Security Practices.....	53
ALBPM Administrator User.....	57
How a User Is Authenticated.....	60
Monitoring.....	60
Monitoring an Engine Using the HTTP Debugger.....	60
Monitoring Engine and Process Performance Using JMX.....	60
Process Instance Archiving.....	61
Archiving and Deployment.....	61
Configuring Archiving.....	61
BAM and Process Data Mart.....	62
BAM Overview.....	62
Configuring BAM.....	68
Process Data Mart Overview.....	68
Configuring Process Data Mart.....	68
Managing Enterprise Applications Using JEE.....	69
Building and Deploying Application EAR Files.....	69
Deploying Application EAR Files.....	70
Enabling Clustering.....	70
Configuring Engine Failover on Enterprise Standalone.....	70
Creating a Backup Location.....	70
Starting a Backup Location.....	71
Versioning Java Class Libraries.....	71
Referencing Processes on Other ALBPM Installations.....	71
Mapping an External Process on ALBPM Enterprise.....	71
<b>AquaLogic BPM Enterprise Overview.....</b>	<b>72</b>
ALBPM Enterprise Components.....	72
Persona.....	72
ALBM Core Components.....	73
ALBPM Supporting Applications.....	73
ALBPM Architectural Overview.....	73
ALBPM Standalone Architecture.....	73
JEE Single Node Architecture.....	74
JEE Cluster Architecture.....	75
WebLogic Portal Architecture.....	75
ALI Architecture.....	76
<b>Administration Reference.....</b>	<b>77</b>
Process Administrator Application Reference.....	77
General Actions.....	77
Organization.....	77
Engines.....	82
Projects.....	93
Variables.....	94
External Resources.....	94
External Processes.....	94

Process Monitoring.....	95
Admin Center Application Reference.....	96
Configuration.....	97
Archive Viewer Application Reference.....	100
Login Screen.....	100
Archive Viewer.....	100
ALBPM Configuration Wizard Reference.....	100
What is the ALBPM Configuration Wizard?.....	100
Running the ALBPM Configuration Wizard.....	101
ALBPM Configuration Wizard Reference.....	101
ALBPM MBean Reference.....	111
Active Process MBean.....	111
Active Processes MBean.....	112
Connection Pool MBean (Standalone).....	112
Connection Pools MBean (Standalone).....	113
Engine MBean.....	113
Execution Pool (Standalone).....	114
FDI Connection Pool MBean (Standalone).....	115
Server Session MBean.....	115
Server Sessions MBean.....	116
Engine and Directory Database Connectivity.....	116
BEA Oracle Driver Properties.....	116
BEA DB2 Driver Properties.....	117
BEA SQL Server Driver Properties.....	118

# Introduction

This section provides general information about the *AquaLogic BPM Enterprise Administration Guide*. This guide assumes that you have performed the procedures outlined in the *ALBPM Installation Guide* and the *ALBPM Enterprise Configuration Guide*.

## Document Scope and Audience

This document is written for system administrators who need to configure and manage AquaLogic BPM Enterprise. It provides conceptual and procedural information for managing ALBPM in a production environment. It also provides reference information for each of the ALBPM administration applications.

This document assumes you have installed AquaLogic BPM Enterprise as outlined in the *AquaLogic BPM Installation Guide*. It also assumes you have performed basic configuration of ALBPM for your environment according to the procedures outlined in *AquaLogic BPM Enterprise Configuration Guide*.

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 <a href="http://edocs.bea.com/albsi/docs61/index.html">http://edocs.bea.com/albsi/docs61/index.html</a> .
Release Notes	The Release Notes file includes information about new features and known issues in the release. It is available at <a href="http://edocs.bea.com/albsi/docs61/relnotes/rel_notes_AquaLogic_BPM_6_1.htm">http://edocs.bea.com/albsi/docs61/relnotes/rel_notes_AquaLogic_BPM_6_1.htm</a> .
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"> <li>In Studio, select <b>Help ► Help Contents</b> to access the complete ALBPM Studio help. Context help is also available by pressing the F1 key, or by selecting <b>Help ► Dynamic Help</b> from the menu.</li> <li>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.</li> </ul>
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><a href="http://one.bea.com/support/">http://one.bea.com/support/</a></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><a href="http://dev2dev.bea.com/albpm/">http://dev2dev.bea.com/albpm/</a></p>
ALBPM Product Center	<p>Download products, read Release Notes, access recent product documentation, and view interoperability information.</p> <p><a href="http://commerce.bea.com/products/aqualogic/bpm/albpm.jsp">http://commerce.bea.com/products/aqualogic/bpm/albpm.jsp</a></p>
BEA Education	<p>Find information about available training courses, purchase training credits, and register for upcoming classes.</p> <p><a href="http://dev2dev.bea.com/community/education/">http://dev2dev.bea.com/community/education/</a></p>
User Groups	<p>Visit the User Group section to collaborate with peers and view upcoming meetings.</p> <p><a href="http://dev2dev.bea.com/community/usergroups/">http://dev2dev.bea.com/community/usergroups/</a></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: <a href="mailto:ALBPMsupport@bea.com">ALBPMsupport@bea.com</a></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

---

This section provides a general overview of the basics of ALBPM Enterprise administration. It provides information about the applications commonly used by ALBPM administrators and provides a roadmap of ALBPM administration.

## Administration Prerequisites

This guide provides information for maintaining AquaLogic BPM Enterprise. It assumes that you have installed ALBPM Enterprise and have performed basic configuration of your environment.

The following sections outlines the administration tasks required to maintain ALBPM Enterprise. It also provides general conceptual and reference information about ALBPM.

## ALBPM Administration Tools Overview

AquaLogic BPM provides several tools for configuring and managing your enterprise environment.

ALBPM Administration Tool	Description
Admin Center	Allows you to start, stop, and configure the ALBPM applications.
Process Administrator	Allows you to publish, deploy, and manage ALBPM projects.
Log Viewer	Provides a graphical tool for viewing Process Execution Engine logs.
Archive Viewer	Allows you to view information about completed or aborted process instances stored in the archive database.
Ant Tasks	Allow you to automate many of configuration and management tasks. Ant tasks are mostly used within large enterprise installations or in situations where you do not want to use the GUI-based administration tools.

## ALBPM Enterprise Administration Roadmap

The following procedures guide you through the basic lifecycle of an ALBPM project and outline the general administrative tasks required for each stage.

The following procedures assume that you have installed and performed basic configuration of your ALBPM Enterprise environment. See the *AquaLogic BPM Enterprise Configuration Guide* for more information.

The tasks outlined in this procedure and their corresponding conceptual information is documented in [Managing the Lifecycle of an ALBPM Project](#) on page 9

1. Create your ALBPM Project
2. Export your ALBPM Project

After you have created your business process and the required resources, export your project to be deployed and published using ALBPM Enterprise. See the *ALBPM Studio User Guide* for more information.

3. Create a directory service.

If you did not create and configure a directory service when you configured ALBPM Enterprise, you must create one before you can create process execution engines and publish and deploy your ALBPM projects.

4. Create a Process Execution Engine.

See [Managing Process Execution Engines](#) on page 23 for more information.

5. Create or map project resources

Depending on your environment, you may need to create new External Resources, Roles, and Variables. Or you may need to map the existing External Resources, Roles, and Variables of your project.

- If you are mapping the existing project resources to identical resources in ALBPM Enterprise.
- If you are creating new resources for your project



**Note:** It is strongly recommended that you create and configure the necessary resources before publishing and deploying your project.

6. Configure the ALBPM Process Execution Engine log.

7. Publish your project

8. Deploy your project

9. Login to WorkSpace

To test that your project is working correctly, you should login to the WorkSpace application.



## Managing the Lifecycle of an ALBPM Project

This section outlines the basic administrative tasks required during the lifecycle of an ALBPM project. It includes information on deploying and publishing a project, information on configuring resources and logging.

After performing the procedures outline in the *ALBPM Enterprise Configuration Guide*, the information in this section takes you through the basic procedures required to run your business processes in an enterprise environment. For information on additional administrative tasks see [Advanced Administration](#) on page 52.

### Accessing Process Administrator

To access the Process Administrator, you must login using the ALBPM administrator password or as another participant who has been given administrative rights.

The following procedures show you how to access the Process Administrator application after starting it with the Admin Center.

1. Launch the Admin Center
2. Select **Launch Process Administrator**  
The Process Administrator application login screen appears in a browser window.
3. Enter your ALBPM administrator username and password
4. Click **Login**.

The main screen of the Process Administrator application appears.

### Managing Your Organization


This section provides general information about organizations and provides procedures for creating and maintaining an organization using ALBPM Enterprise. It also provides information on creating and configuring directory services.







#### Organization Overview

Business processes that require user interaction generally occur within the context of an organization. Defining an organization allows users participate in your business process once it is published and deployed. It also ensures that users can only perform activities appropriate to their role within the organization. Each ALBPM Project must have an organization defined.

Within ALBPM an organization defines a hierarchical structure that reflects the real-world organization of your business. An ALBPM organization defines the way people are grouped and defines the roles or each group and individual.

The following table lists the elements of an ALBPM organization.

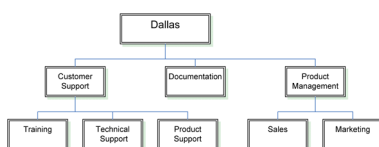
Element	Icon	Description
Organizational Units		Organizational Units are used to represent departments or divisions within the organization. Organizational Units can be defined hierarchically so that, for example, you can represent divisions within an organization, departments within a division, areas within a department, and so on. You can assign Participants, Calendars, and

Element	Icon	Description
		Business Parameters to an Organizational Unit. You can also deploy processes under an organizational unit.
Roles		Roles are used to represent functions performed by people related to the organization. Roles are assigned to participants or groups, and these assignments define the permissions the participants have when executing AquaLogic BPM tasks through WorkSpace.
Groups		Groups are collections of roles. In this way, it is possible assign multiple roles to participants in a single step. Groups may also contain other groups.
Participants		Participants are the actual people who participate in the organization, usually as end users of the BPM implementation.
Holidays		Holidays Define the organization's non-working days. These rules inform the Process Execution Engine that there is an exception to the normal calendar rules on certain days of the year.
Calendars		Calendars define the organization's work week and work schedule. Calendar rules can be assigned to organizational units.
Business Parameters		Business Parameters are used to maintain constant values defined either for the entire organization, or at the Organizational Unit level. These parameters are visible to all instances and all processes across the Organization. Although business parameters may be changed every once in a while, they are not meant to be used as variables. Rather, they provide a way of storing long-lived values, such as a sales tax rate, without having to hard-code them into PBL methods.

### About Organizational Units

Organizational units are typically departments or divisions within an organization. Organizational units can be organized in a hierarchy.

For example:



In this hierarchy, Dallas is a single top-level organizational unit which contains the Customer Support, Documentation, and Product Management organizational units, while Customer Support contains Training, Technical Support, and Product Support organizational units.

Once the organizational units have been defined, participants may be assigned to one of the organizational units in the hierarchy. Processes can be deployed for one of the organizational units defined so that only participants in that organizational unit and in lower levels within the hierarchy are able to perform tasks in a process.

Every organizational unit might have a different calendar rule associated to it. This allows the Process Execution Engine to take into account time zones and working schedules set for the organizational unit where processes are deployed and to calculate deadlines accordingly.

Studio allows you to define the organizational hierarchy and the properties of each organizational unit. Remember that all the changes introduced to the organizational structure require a `Refresh Engine Data` operation if they are to be made available to processes on a currently running Process Engine.

## Roles

A role in the organization is a title or job function which is associated to a set of activities performed by participants of the organization.

Examples of roles include Accounts Manager, Sales Clerk, or Customer. Roles are similar to job titles, but are more flexible because a participant can be assigned to several roles, and some roles, such as Customer, may not be jobs at all.

## Roles and Activities

Every interactive activity is defined under a role. This is done by placing the activity within a *swim lane* with the name of the role. Swim lanes with no role name are only used for automatic activities which require no user interaction, and are not assigned to a role.

## Roles and Participants

Participants are assigned one or more roles. This is how the process can determine which participants can execute a given activity.

See [Permissions and Security in an Organization](#) on page 13 for more information.

## Parametric Roles

A role can be defined as *parametric*. A parametric role includes a parameter which can adopt one of a set of values defined with the role.

For example, the role could be called "sales support" and the parameter could define a set of regions, such as East, West, and South. Even though there is only one role from a functional point of view, participants are assigned based on the location parameter.

Parametric roles require an instance because the parameter to be used is defined as an instance variable. This means that global activities cannot be assigned to parametric roles.



**Restriction:** Global activities cannot be assigned to parametric roles.

## Groups

Groups are collections of roles. In this way, it is possible to assign multiple roles to participants in a single step. Groups may also contain other groups.

Unlike an organizational unit, which can belong to only one parent organizational unit, a group may be included in many other groups. Groups are therefore not organized in a hierarchical structure. However, if a group is included in another group, then it cannot have as a member that group. That is, so long as group A includes group B, group B cannot include group A.

## Participants

Participants defined in the organization are all the people enabled to track and perform tasks of business processes designed and developed with Studio.

A participant might belong to an organizational unit. If so, he can only perform tasks on processes deployed in that organizational unit or any organizational units that are below it.

You can assign a set of roles to a participant. A participant who logs in to WorkSpace can perform all the tasks defined for the roles assigned to him.

You can create, edit, and delete participants from the **Project Navigator**. Participants are usually created within Studio for process design and testing purposes. When the process is implemented into production, actual participants will normally be imported from an existing company directory or will be defined within Process Administrator.

## About Holiday Rules

Holiday rules are collections of holidays that can be applied to calendar rules.

Multiple holiday rules can be created as needed for different [Calendar Rules](#) on page 12. Holiday rules affect the available work days for participants and the scheduling of activity deadlines.

## Calendar Rules

Calendar rules define the work hours, time zone, and holiday rule assignment for organizational units.

Multiple calendar rules can be created as needed for different organizational units (such as day shift, night shift, east coast, west coast, etc.). Calendar rules determine the available work days for participants and the scheduling of activity deadlines.

## Business Parameters

Business parameters are used to store long-lived information defined at the organization level.

Information suitable for storage as a business parameter includes company address and phone data, tax rates used in calculations within the process, or infrequently changed economic values such as the prime lending rate. Business parameters are visible from any process within a project and should generally be thought of as constants, though they can be changed.



**Tip:** Business parameters should be used for infrequently changed values which you do not want to include in the actual code. For example, company address data, the prime lending rate, or a sales tax rate are all good uses for business parameters.

It is strongly recommended *not* to use Business Parameters for values which will change very frequently (once a day or more). For those cases consider other options.

If you do need to change a Business Parameter from PBL code, you can change it at runtime using the component Business Parameter in the Lib category. See the Studio. component documentation.

- If you change a Business parameter from a method you must be aware that the new value is not immediately available for all instances. Even more, if this value is changed from a PBL-Method, the result may not always be the expected one and not available at the same time across the all participants.
- If the business parameter is used in a due transition expression of an activity, the business parameter value that applies is the one defined at the time the instance enters the activity. For example, let's say the business parameter "MAXTIME" is used in the due transition expression of the activity "Reply to Customer". When the instance "Request Customer 1" arrives, the due time is calculated using the value that the MAXTIME has at that moment. If another instance (in any process) changes the value of MAXTIME or you manually change it in the Process Administrator, the new value does not apply for the due time of the instance "Request Customer 1" for the activity "Reply to customer". It will apply for all instances that arrive to that activity *after* the business parameter was changed.



**Note:** If you change the Business Parameter at runtime, and you then stop and restart the Studio Process Engine, all business parameters are restored from the project definition. However, the *Enterprise* Process Engine *does* maintain Business Parameter values through a start/stop cycle, because in a production environment Business Parameter changes are assumed to be permanent.

## Permissions and Security in an Organization

### Role Permissions and Categories

#### Role Permissions

Value	Description
Execute	Allows participants to access a process instance.
Route	Allows participants to send an instance to the next activity in a process, send an instance back to the previous activity in a process, or send an instance to a specific participant assigned to the role for the next activity in a process.
Suspend	Allows participants to pause a process at the instance activity and make the process unavailable to other users until resumed.
Abort	Allows participants to abort an instance.
Delegate	Allows participants to reassign an instance to another participant with a lower rank.
Reassign	Allows participants to grab an instance from another activity in the process.
Escalate	Allows participants to reassign an instance to another participant with a higher rank.
Peer Assignment	Allows participants to reassign an instance to another participant with the same rank.

#### Role Categories

A category is a value from 1-9 that determines if a participant can delegate, escalate, or assign a process instance to a peer.

#### Viewing and Modifying the Properties of a Participant

To view or modify the properties of a participant--for example, their organizational unit, roles, or groups--launch the Process Administrator and do the following:

1. In the navigator pane, expand Organization, then select Participants.  
The Participants window appears in the right pane. It lists the participants that are entered into the system.
2. Select the participant whose properties you want to view or modify.  
The Edit Participant window appears.
  - The Properties section displays the participant's main properties.
  - The Advanced Properties section provides links enabling you to view or modify this participant's password, roles, groups, absence periods, or organizational units managed by this participant.

You can modify any of the values in the Properties section except the User ID by changing the text in the corresponding property field. When you finish, click **Save**.

To change a participant's password, or to view or modify their assigned roles, groups, absence periods, or the organizational units they manage:

1. In the Advanced Properties section, select the corresponding link.
2. Make the changes.
3. Click the Edit Participant breadcrumb at the top of the pane.

### Configuring Permissions for Participants

During runtime, the ability to perform an operation on an activity depends, in part, on a participant's roles and permissions. Two participants can have the same role, but different relations with it depending on their permissions. For example, one participant in the Shipping Clerk role may have permissions to suspend and abort activities, but not re-assign them. Another participant with that same role may have permission to re-assign activities.

Granting permissions to perform an operation is done by the administrator. To grant permissions to a user, the administrator follows these steps using Process Administrator:

1. In the navigator pane, expand Organization, then select **Participants**.  
The Participants window appears in the right pane. It lists the participants, their organizational units, and their e-mail addresses, and indicates whether they are enabled.
2. Click the link for the participant you want to specify privileges for.  
The Edit Participant window appears in the right pane.
3. In the Advanced Properties section, select **Assigned Roles**.  
The Assigned Roles window appears in the right pane. It lists the roles that have been created for the process, and, in abbreviated form, the permissions associated with that role for this user. It also tells you the user's level in the hierarchy, which, in turn, determines those lower in the hierarchy to whom they can delegate tasks, or those higher to whom they can escalate activities.
4. Click the name of the role within which you want to specify privileges.  
The Role Assignment window appears in the right pane.
5. In the Properties section, in the Permissions row, select the permissions you want to grant to this user.  
See [Role Permissions and Categories](#) on page 13 for more information.
6. When you are finished, click **Save**.

### Setting Roles for Participants

When adding a participant, the administrator can specify a role. The role indicates the participant's hierarchical rank in relation to other participants with the same role. A rank can be from 1 to 9, with 1 being the lowest and 9 being the highest. A participant's rank determines which other participants he or she can assign instances to.

To set categories for a participant follow these steps:

1. Launch Process Administrator.
2. In the navigator pane, expand Organization, then select **Participants**.  
The Participants window appears in the right pane. It lists the participants, their organizational units, and their email addresses, and indicates whether they are enabled.
3. Click the link for the participant for whom you want to specify a category.  
The Edit Participant window appears in the right pane.
4. Click **Assigned Roles** in the Advanced Properties section.  
The Assigned Roles window appears in the right pane. It lists the roles that have been created for the process, and, in abbreviated form, the permissions associated with that role for this user. It also tells you the user's category--that is, the user's level in the hierarchy. The category determines those lower in the hierarchy to whom they can delegate tasks, or those higher to whom they can escalate activities.
5. Click the name of the role within which you want to specify privileges.  
The Role Assignment window appears in the right pane.
6. In the Properties section, in the Category row, select the value of the category you want to assign to this user.
7. Click **Save** when you are finished.

### Enabling User Administration for a Participant

You can allow a participant to manage other participants, including role and group assignments.

Before performing the following procedure, ensure that you have created the participant you want to have organizational administration access.

Enabling user administration for a participant allows them to create and update participants and roles without granting general administrative privileges.

1. Launch Process Administrator
2. Click **Organization**.
3. Click **Participants**.
4. Select the participant to whom you want to grant user administration.
5. Select **Enable User Administration**
6. Click **Save**.

## Creating an Organization Using Process Administrator

The following high-level task outlines the basic procedures for creating an organization using Process Administrator.

In general, you should define your organization from the top down. You should create higher-level elements like organizational units first. Then create lower-level elements like participants.

1. Create an Organizational Unit.

See [Creating an Organizational Unit Using Process Administrator](#) on page 15. Depending on your needs, you may need to create multiple organization units. If you need a hierarchical relationship between organizational units, create the parent units first, then the child units.

2. Create Roles

See [Creating a Role](#) on page 16.

3. Create Groups

See [Creating a Group Using Process Administrator](#) on page 16.

4. Assign Roles to Groups

See [Assigning Roles to Groups Using Process Administrator](#) on page 16

5. Create Participants

See [Creating a Participant Using Process Administrator](#) on page 17

6. Assign Roles to Participants

See [Assigning a Role to a Participant Using Process Administrator](#) on page 17

7. Assign Groups to Participants

See [Assigning a Group to a Participant](#) on page 17

## Creating an Organizational Unit Using Process Administrator

1. Launch Process Administrator
2. Click **Organization ► Organizational Unit**.
3. Click **Add**.
4. Supply the following information:

Option	Description
--------	-------------

<b>Parent ID</b>	Defines the parent organizational unit of this org unit.
------------------	--



**Note:** If you are defining a hierarchy of organizational units, you must create parent units before the children.

<b>Name</b>	Specifies the name of the organizational unit
-------------	---

Option	Description
Description	Contains an optional description of the organizationl unit.
Calendar Rule	Defines the calendar rule used by this organizationl unit. If you have not already defined calendar rules, you can assign them later.

5. Click **Save**.

After creating an organizational unit, you can continue creating lower-level elements of the organization.

**Creating a Role**

- 1. Launch Process Administrator
- 2. Click **Organization ► Role**.
- 3. Click **Add**.
- 4. Enter the following:

Option	Description
Name	Specifies the name of the role.
Description	Specifies an optional description of the role.
Is Parametric	Allows you to define the role as a parametric role. If you are defining a parmetric role, you must supply values.

5. Click **Save**.

**Creating a Group Using Process Administrator**

- 1. Launch Process Administrator
- 2. Click **Organization ► Groups**.
- 3. Click **Add**.
- 4. Enter the following:

Option	Description
Group ID	
Name	
Description	
Organizational Unit for Administration Scope	

5. Click **Save**.

**Assigning Roles to Groups Using Process Administrator**

- 1. Launch Process Administrator
- 2. Click **Organization ► Groups** .
- 3. Select the group you want to assign roles to.
- 4. Click **Assigned Roles** under**Advanced Properties**.  
The list of roles assigned to this group appears.
- 5. Click**Add**.  
The **Role Assignment** page appears.
- 6. Select the role you want to add from the list of roles under**Role ID**.
- 7. If you are assigning a parametric role, select a parameter under**Parameter**.



8. Select the permissions for this role.  
See [Role Permissions and Categories](#) on page 13 for more information.
9. Define a category for this role.  
See [Role Permissions and Categories](#) on page 13 for more information.
10. Click **Save**.

### Creating a Participant Using Process Administrator

1. Launch Process Administrator
2. Select **Organization ► Participants**.
3. Click **Add**
4. Enter information for the new participant  
See [Participants](#) on page 80 for information on each field.
5. Click **Save**.  
The **Advanced Properties** section appears.
6. Assign any groups or roles to this participant.  
You can assign groups and roles later, if necessary.
7. Click **Save** to create the new participant.

The new participant appears in the list of participants within your organization.

### Assigning a Role to a Participant Using Process Administrator

- 1.

### Assigning a Group to a Participant

- 1.

## ALBPM Directory Service

The directory service is a metadata repository that stores authentication, authorization, and project information. For example, it provides the Process Execution Engine with information about which processes to run, which participants are involved, and which components to use.

The directory service stores the following information:

- Organization Information, including information about the relationships between the following:
  - Participants
  - Roles
  - Groups
  - Organizational Units
  - Calendar Rules
- Process Execution Engine configuration information
- Deployed Projects Information
  - Process Models

- Security Declarations
- Executable Project Code
- Deployment Information (engine, organizational units)
- General Configuration Settings
  - External Resource Configuration
  - BAM Configuration

Within an ALBPM Enterprise allows you to create multiple directory services. This allows you to have different directory services for different ALBPM environments. For example, you can have different directory service configurations for development, testing, and production environment. Each directory service has its own ALBPM administrator username and password.

### Types of Directory Services

In an ALBPM installation, a directory service can be either a relational database or a hybrid of both a relational database and an LDAP directory.

#### Database-Only Directory Service

This topic should provide a general description of the DB-only dir service.


#### Hybrid Directory Service

In a hybrid configuration, authentication and authorization data can be stored in the LDAP directory while the rest of the metadata resides in a transactional RDBMS. This avoids the need for replication of participants and entitlements data.

FDI (Fuego Directory Interface) is the internal API the ALBPM products use to access the Directory Service information. All changes performed in the LDAP directory become automatically visible to FDI applications without replication. FDI access to the LDAP directory is read-only.

More specifically, when using a hybrid directory service, BEA AquaLogic BPM Suite retrieves the following from the LDAP directory:


- Organizational data including users, groups, group assignments and organizational units
- Security credentials of participants (including administrators)
- A portion of authorization data. In this case, groups are assigned in the LDAP directory rather than from ALBPM Process Administrator or any other AquaLogic BPM component. Roles, however, are stored in the ALBPM directory RDBMS and are assigned to groups by using the Process Administrator.


 **Note:** All other metadata is stored in the relational database.

### 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 116.
- 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 116

**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](#) on page 20 for information on creating an organization if you are using a database-only directory service.

See [Configuring a Hybrid Directory Service](#) on page 20 for information on configuring a hybrid directory service.

## Configuring Organization using a DB-Only Directory Service

1. Create a directory service configuration.
2. Create your organizational structure using Process Administrator  
See [Creating an Organization Using Process Administrator](#) on page 15 for more information.

## Configuring a Hybrid Directory Service

This high-level task demonstrates how to configure a hybrid directory service using Process Administrator.

This task assumes that you have created your organization within your LDAP provider, including organizational units, groups, and users. It also assumes that you have created and configured a directory service using the Admin Center. ALBPM Enterprise uses this directory service to connect to your LDAP server.

See [Creating a New Directory Service](#) on page 18 for more information.

1. Launch Process Administrator
2. Verify that LDAP information is being read by Process Administrator.
  - a) Click **Organization ► Organizational Unit** .  
The organizational units defined in your LDAP server appear in the list of organizational units.
  - b) Click **Organization ► Groups**  
The groups defined in your LDAP server appear in the list of groups.
  - c) Click **Organization ► Participants**  
The participants defined in your LDAP server appear in the list of participants.

If you do not see your LDAP organization information within Process Administrator, check the configuration of your directory service.

3. Create roles using Process Administrator  
Roles are used to control access to different activities within a process.  
See [Creating a Role](#) on page 16  
After creating roles, you must assign the roles to the appropriate groups and participants.
4. Assign roles to groups  
See [Assigning Roles to Groups Using Process Administrator](#) on page 16
5. Assign roles to participants  
See [Assigning a Role to a Participant Using Process Administrator](#) on page 17

After configuring a hybrid directory service, you should deploy and publish a project. Then, you should login to the WorkSpace to ensure that access to different activities within a process are available to the correct participants in your LDAP organization.

## Managing Projects

The following section contains information on managing project in ALBPM Enterprise.

An ALBPM project contains a set of related business processes and their associated resources. At the enterprise level, projects are published and deployed within a Process Execution Engine. Once a project is deployed, its processes are available to end-users via the WorkSpace application.

## 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](#) on page 22 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.

## Undeploying a Project



**Note:** When you undeploy a project, all the project version assets are deleted, including instances, audit trails, instance attachments, and instance notes.

To undeploy a project, use Process Administrator and follows these steps:

1. In the navigator pane, click **Projects**.  
The Published Projects pane displays a list of published projects. The Deployment column tells you which projects are also deployed.
2. In the Published Projects pane, identify a project that you want to undeploy, and, in the Deployments column, click **Completely deployed**.  
The Deploy window appears in the right pane.
3. In the Deployment Topology section, click **Undeploy**.  
The Deploy window prompts you for confirmation that you want to undeploy. For each process in the project, it alerts you that the process may have instances in execution that will be lost if you undeploy it.  
  
By using the drop-down list next to each process, you can choose to:
  - Keep the project active
  - Deprecate the project
  - Undeploy the project
4. When you have made your determination, click **OK**.  
The project is undeployed. The Publish Projects window appears in the right pane. The project you undeployed is listed as not deployed.

Project Versioning

Project versioning occurs when you need to publish and deploy changes to a project. These changes can include revisions to your BPM processes or they can include minor changes to the associated resources contained within a project.

Project versioning allows you to introduce these changes without disrupting the normal flow of your existing project. Depending on the nature of the changes you are making, ALBPM provides different levels of project versioning.

ALBPM defines two different types of project versioning:

Revision Versioning	the most common type of versioning that involves changes to project elements. Changes are applied to all existing instances
Minor and Major Versioning	<p>occurs when there is a compatibility issue between the new and old version of a project. This type of versioning is also applicable when there is a need for a clear break between project versions. These generally occur as a project evolves.</p> <p>During major and minor revisioning, both the new and old version of the project run in parallel. New created instances are run using the latest version, while existing instances continue running on the old version.</p>

Creating a Project Version

1.

Revision Versioning of a Project

Some changes to an ALBPM Project do not require a major or minor revision change.

The following changes can be made to an ALBPM Project without requiring a change to the version of a project:

- Adding a role to a process.
- Deleting an empty role from a process. An empty role is a role that contains no activities.
- Moving an activity to a different role within a process.



**Note:** None of the instances of the activity being moved can be selected by a participant. You should ensure that the role where the activity is moved is assigned to all participants of the activity.

- Adding an activity to a process.
- Adding or deleting a Global or Grab activity.
- Changing the business rules of a process
- Changing the Condition or Due interval of a transition.
- Adding or deleting a transition within a process.
- Adding or deleting an instance variable within a process.
- Deleting a variable from the argument mapping of a process.
- Changing the implementation type of a process.
- Modifying a Screenflow or Procedure.

Major and Minor Versioning of a Project

Major and minor versioning of a project is required when changes create incompatibilities between the existing and new version of a project.

The following types of changes require major or minor revisioning:

- Deleting an activity from a process. Deleting an activity from a process creates incompatibilities between currently executing and waiting instances.
- Changing the type of an instance variable in a process. This can cause incompatibilities between the existing and new values of a process.
- Changing activity types within a process.

## Managing Process Execution Engines

### Engine Status

Engine Status	Description
Not Running	The process execution engine is not operating
Primary Server Backup	The engine is in BACKUP mode. This means the engine is waiting for the primary engine to fail before initiating failover.
Connections Setup	The engine is able to establish connections to data sources including the directory service, LDAP server, and engine database.
Stand By	The engine is able to do some internal processing, but, specifically, cannot accept external client requests
Ready	The engine is fully operational and can start accepting external client requests.

### Engine Disposer Service

The disposer service is a process execution engine service that deletes completed process instances from the engine database or instance data to the archive database.

The disposer service is responsible for removing information about completed or aborted process instances from the engine database.

- If archiving is enabled, the disposer moves instance data from the engine database to the archive database.
- If archiving is not enabled, the disposer deletes instance data from the engine database.

You can configure how often the disposer service runs and how long instance data is kept in the engine database before it is moved or deleted.

The disposer service is configured using the Process Administrator under **Edit Engine Properties ► Services**.


### Creating a Process Execution Engine

Using Process Administrator, you can create a new Process Execution Engine.

Before creating a new Process Execution Engine, ensure that your engine database server is configured and running. When creating a new engine, Process Administrator connects to the database and creates the required database tables.

1. Launch Process Administrator
2. Click **Engines**.
3. Click **Add**.
4. Enter the following information for your new engine:

Option	Description
Engine Name	Defines the name of the engine

Option	Description
<b>Engine Type</b>	Specifies the type of engine. Valid values are: <ul style="list-style-type: none"> <li>Enterprise</li> <li>J2EE</li> </ul> <p> <b>Note:</b> When using the ALBPM Standalone, you can only create an engine of type Enterprise. When using the J2EE version, you can create either type.</p>
<b>Engine Database Type</b>	Specifies the database driver the engine uses.

5. Click **Next**.

6. Select the database driver for you engine database vendor.

7. Enter the connectivity information for your engine database.

See [Engine and Directory Database Connectivity](#) on page 116 for information on supported database types.

8. Click **Next**.

The **Engine Configuration** page appears. From this page you can edit additional engine properties before you create the engine. See [Engines](#) on page 82 for information on additional engine properties.

9. Click **Save** to create the Process Execution Engine.

## Starting and Stopping Process Execution Engines



Using Process Administrator, you can start or stop Process Execution Engines.

1. Launch Process Administrator

2. Click **Engines**.


The list of available Process Execution Engines appears.

3. In the **Engine Actions** column select one of the following:

- Click  to start the engine.
- Click  to stop a running engine.



**Note:** If you are using the J2EE version of ALBPM Enterprise, the ALBPM Deployer must be configured and running for these options to work.

Based on the action you selected, the Process Execution Engine is started or stopped. If there are problems, click  to view the startup log.

Additionally, you can view the engine log file located at: `<ALBPM_HOME>/albp6.0/<version> /log/ <engineName>`

## Exporting and Importing an Engine Configuration

ALBPM Enterprise allows you to export engine properties and import them to create another engine.

When you export an engine configuration, it is saved as an XML file. This file is saved as *your\_engine\_name*.engine.exp. You can edit this XML file manually to modify the engine configuration before importing it using Process Administrator.

Some of the situations where exporting and importing an engine configuration include:

- Creating a backup of an engine configuration before making configuration changes.
- Creating a new engine with an identical configuration. This is useful if you need to recreate an engine in a different environment.
- Using the exported XML as input to ALBPM Ant tasks.



## Exporting a Process Execution Engine Configuration

You can export an Process Execution Engine's configuration to an XML file. You can import this file to create another Process Execution Engine with identical configuration.

1. Launch Process Administrator
2. Select the Process Execution Engine whose properties you want to export.
3. Click **Export**
4. Click **Save File** to save the engine configuration to your local file system.

The file is saved as *your\_engine\_name.engine.exp*.

See [Importing a Process Execution Engine Configuration](#) on page 25 for information on importing an engine configuration.

## Importing a Process Execution Engine Configuration

Importing an engine configuration allows you to create a new Process Execution Engine based on a previously exported engine configuration file.

1. Launch Process Administrator
2. Select **Engines**.
3. Click **Import**
4. Browse to the location of the exported engine configuration.

The engine configuration file is an XML file with a *.exp* extension.

5. Click **Import**

Process Administrator creates a new Process Execution Engine based on the imported configuration file.

If you experience a problem when importing an engine, you can view the Process Administrator log located in: `ALBPM_HOME/version/log/webConsole.log`. Additional troubleshooting information may be provided in the Tomcat application server logs.

## Tuning Engine Database Runtime Performance

The engine database configuration can have a significant impact on engine performance as well as the overall performance of ALBPM Enterprise.

Each process execution engine must perform database transactions to persist data when a PBL method is executed or when an instance proceeds to the next activity. Data persistence is necessary to ensure the state of a process instance can be recovered after engine failure.

In addition to vendor-specific database performance tuning, the following sections provide general guidelines on tuning the engine database.

### General Database Tuning Considerations

The Maximum Pool Size parameter allows you to ensure the correct number of database connections are available. The Maximum Pool Size parameter specifies the maximum number of connections that the server can allocate to perform transactions against the server's database. If for some reason the engine database needs more connections it will be able to increase the connection pool up to the number indicated in this field.

If more connections are needed, the requests are queued until the first transaction finishes, the connection is freed for another transaction in the queue to start. This parameter should reflect the number of concurrent interactive users. This is the determining factor. Make sure the database has that many client connections configured to be consumed by FuegoBPM. This is a combined value from the interactives thread + automatic execution threads that can be concurrently active. Do not define a huge value (i.e.: 400) for this parameter since this will create also contingency in the RDBMS used by the FuegoBPM Server.

It is preferable to wait for a connection to be released after a transaction is finished than generate a big concurrency of transaction in the target RDBMS. There must be a balance between how soon the transactions can be finished without generating a bottleneck and contingency in the database. This will also depend on the hardware where your RDBMS is deployed and the dimensioning of the RDBMS used by the FuegoBPM Server. Make sure that the database is configured so that there are enough sessions to handle the number of maximum connections that the engine database may use.

### Oracle Database Tuning Considerations

Make sure you have enough sessions. Check Oracles SESSIONS parameter. For no contingency, you should have a session for each Server connection that the FuegoBPM; Server may use for a transaction.

Make sure there are enough processes on the Oracle side. Check Oracles PROCESSES parameters. This parameter depends also on the values assigned to SESSIONS and TRANSACTIONS. For no contingency, you should have a process for each FuegoBPM Server FuegoBPM Web Console 150 connection that can be executing a transaction at the same time.

Make sure the Oracle memory is properly configured for the number of concurrent transactions to be executed by a FuegoBPM Server. The dimensioning of the memory in Oracle is related to the configuration of the SGA. FuegoBPM only uses from the SGA the following sections: BUFFER CACHE, LARGE POOL, SHARED POOL

### Determining Engine Memory Allocation (Standalone)


ALBPM Standalone allows you to tune Process Execution Engine performance by setting the memory allocation for the engine and JVM.



**Note:** On the J2EE version of ALBPM memory allocation is configured at the container level. See your application server documentation for more information.

One factor affecting the performance of an engine is the amount of memory available to the the engine and the JVM running the engine. Allocating too much or too little memory can negatively influence performance. The amount of memory required for an engine is determined by three parameters:

Parameter	Description
Maximum JVM Heap Size	<p>Specifies the maximum amount of memory, in megabytes, the engine can use. If the engine's memory usage reaches this limit, the engine stops itself and restarts.</p> <p>You should always try to set the minimum value for the JVM heap size.</p> <p>The default is 256 megabytes</p>
Maximum Instance Size	<p>Specifies the maximum amount of memory a process instance can use. This property limits the size of all process instance variables used by a process instance. When any instance exceeds this limit, the engine is not able to persist instance data and the task fails and must be re-executed after increasing the size of this property.</p> <p>The default is 16 kilobytes</p>
Instance Cache Size	<p>Defines the cache size used to store recently accessed process instances. This cache is shared by all processes deployed on the engine.</p> <p>The default is 5000 instances</p> <p>The default value of 5000 instances in the cache is usually considered above average for medium size installations. If the engine has processes with thousands of instances running and these processes have a high concurrency, it is advisable to increase the size of the cache so that the engine does not need to reload instance information when new instances are accessed.</p>

 **Note:** The engine outputs warning messages to the engine log when these the engine is approaching these limits.

 **Note:** These parameters are set using the Process Administrator on the **Execution tab**.

These three parameters work together to determine engine performance. Before increasing the value of these parameters, you should check the process instance variables of your deployed processes. Larger instances variables take longer to persist and require increased engine cache sizes.

A good rule of thumb for setting these parameters is:

$$\text{Maximum Instance Size} * \text{Instance Cache Size} < \text{Maximum JVM Heap Size} / 2$$

Following this rule ensures that the instance cache will not consume more than half of the engine's allocated memory. If you need to increase the Maximum Instance Size, you should also increase the Maximum JVM Heap Size.

You should always try to set the minimum value for the JVM heap size. You should only increase the JVM heap size when you are required to increase the Maximum Instance Size or Instance Cache Size. However, before increasing the size of these parameters, you should check the instance variables of you processes. Large instance variables take longer to persist and may require more cache memory. If the size of your instance variables is too large, you may need to redesign the process. For example, using arrays in an instance variable can often cause instance variable to be too large.

If you have processes that are using instances concurrently and accessed frequently, you may want to analyze if the instances really need to be loaded back into the cache. Fetching instances from the database very often may incur in performance degradation problems. Again a redesign of your processes may be in order.

## Project Variables on ALBPM Enterprise

Can someone give me an overview of how variables are handled on Enterprise? >> I have the basic procedures of creating and localizing them, but I am >> lacking the information about why one needs to do this. > > Those variables map to the "Project Variables" (as they are called in Studio). > In short, those variables must be declared because each of them become > separate columns in the database. From Proc. Admin. you create them and then > the Engine adds new columns to one of its tables (that is, the DB structure > changes). Eduardo] Columns for Project Variables are added to the Engine PPROCINSTANCE and PPROCINSTEVENT tables. If in addition, the Project variable is qualified as "Dimension" or "Measure", these will be as well added to some tables in the BAM and DataMart Databases. Depending on the engine used at deployment time, the engine db tables that will be modified.

> Much like the Roles mapping, there's also a Variable's mapping. You declare > the variables in Process Administrator and when you publish/deploy a project > you "map" the Project's declared variables to those configured on your > Enterprise environment. The names do not need to match, but the types must be > compatible. [Eduardo] It may be worth mentioning here that when new variables are added, that the Engine DB User has enough permissions to enable the proper ALTER of the previously mentioned Engine Tables. Otherwise, the table will not be able to work correctly. In the event that the engine tables cannot be modified, a proper error with the alter SQL Statement will be printed in the Engine logs. This SQL statement can be provided to the DBA so that it is manually run and continue correctly. However, the ALTER permission should be quoted somewhere since it is of special interest to administrators and DBAs. In very restricted environments, they do not grant ALTER permissions to the Engine schema owner. > Important point: > \* You can map variables from 2 (or more) different projects to a single > variable in Enterprise. There's no conflict (they re-use the same DB column, > but each instance has it's own value) >> \* The variable as you define it in Proc Admin is just a place holder. The > actual meaning of the variable is given by each project. So, 2 (or more) > projects may map to the same variable, but each project can give that > variable a different meaning (whether this is a good idea or not is a > different question) [Eduardo] This should be consider to some extend best practices since it helps better utilize existing resources. Tables with too many fields tend to run slower. In addition, there is a limit as to HOW many columns can be defined in a Table. Oracle being the RDBMS vendor being the most generous (less than 1024 table columns).

Ideally, different projects should map to the same variable with similar semantics. But this is not necessarily true. These variables are usually filtered by the process that provides the context for their content. So if for some reason, you need to use the same physical Enterprise variable for 2 different project variables and different meaning BUT the same type,

then it is legal to do the mapping. Usually this is done to avoid an unnecessary increase in the amount of columns since this produces a performance side effect.

Creating Variables in ALBPM Enterprise

- 1. Launch Process Administrator
- 2. Click**Variables**.
- 3. Click**Add**.
- 4. Enter the following information:

Property	Description
Name	Specifies the name of the new variable.
Type	Defines the type of variable. Supported types are: <ul style="list-style-type: none"><li>• Boolean</li><li>• Integer</li><li>• Real</li><li>• Time</li><li>• Decimal</li><li>• String</li></ul>
Size	Defines the size of the variable.
Business Variable	Specifies that the new variable is a business variable.

- 5. Click **Save**.

Localizing Variables Using Process Administrator

- 1.

Logging

ALBPM Enterprise generates log files that store information about the status and behavior of ALBPM applications and processes.

ALBPM Log Files

AquaLogic BPM uses multiple log files to store information about different components. These include log files for installation, application behavior, and runtime monitoring.

The following sections describe the log files used by AquaLogic BPM. These log files may be useful in troubleshooting problems that can occur when you are installing or using ALBPM.

Runtime Logs

ALBPM Enterprise uses different runtime logs to display information about running components and applications.

Log File	Description	Default Location
Process Execution Engine Log	Displays information about the engine based on the severity level you define. Each engine contains its own log.	<code>&lt;ALBPM_HOME&gt;/alpm6.0/edition/log/engineName.log</code>
PAPI Log	Each PAPI client generates its own log file. An application that uses PAPI should configure where to store the log files, the severity level, and the date format of the messages. See the <i>AquaLogic BPM Process API Developer Guide</i> .	Depends on application
PAPI WS Log	Store log messages related to PAPI WS and can be configured using the Admin Center.	<code>&lt;ALBPM_HOME&gt;/&lt;version&gt;/log/</code>
BAM and Data Store Logs	Store log messages about the updated service and the BAM and Data Store applications. The updater service log can be configured using the following configuration file: <code>conf/WarehouseService.conf</code> . The BAM and Datastore each generate their own logs than can be configured using the Process Administrator.	The BAM and Data Store logs use the following defaults: <ul style="list-style-type: none"> <li>• <b>Updater Service Log:</b> <code>&lt;ALBPM_HOME&gt;/&lt;version&gt;/log/dwupdateservice.log</code></li> <li>• <b>BAM Log:</b> <code>&lt;ALBPM_HOME&gt;/&lt;version&gt;/log/bamdwupdate.log</code></li> <li>• <b>Data Store Log:</b> <code>&lt;ALBPM_HOME&gt;/&lt;version&gt;/log/dwupdate.log</code></li> </ul>
Feeds Log	Contains messages generated by the RSS feeds service and can be configured in the following properties file: <code>&lt;ALBPM_HOME&gt;/enterprise/apps/feed/WEB-INF/feeds.properties</code>	<code>ALBPM_HOME/enterprise/log/feeds.log</code>

## Application Logs

Displays status information about individual ALBPM applications.

Log File	Description	Default Location
Process Administrator Log	Configurable with <code>fuego.log.workspace.*</code> properties in <code>webapps/webconsole/WEB-INF/webConsole.properties</code> .	<code>.../al bpm6.0/edition/log/webConsole.log</code>
Studio Log		<code>al bpm6.0/studio/log/</code>
Eclipse Log		<code>eclipse workspace directory/.metadata/.log</code>
WorkSpace Log	Configurable with <code>fuego.log.workspace.*</code> properties in <code>webapps/workspace/WEB-INF/workspace.properties</code> . Studio default location: <code>.../al bpm6.0/studio/log/workspace.log</code> Enterprise default location: <code>.../al bpm6.0/edition/log/workspace.log</code>	
Admin Center		
Log Viewer		

Installation and Update Logs

The installation and update logs provide information that is useful for troubleshooting problems that may occur when you are installing ALBPM. These logs are usually updated only during the installation of ALBPM. After installation is complete, these logs are generally not updated.

Log File	Description	Default Location
Studio Installation Log	Contains information generated by the ALBPM Studio Installer.	ALBPM_HOME\BPA\log\BPMStudio\log
Enterprise Installation Log	Contains information generated by the ALBPM Enterprise Installer.	ALBPM_HOME\j2eews\log\filename
Enterprise Update Log	Contains information generated when applying updates and maintenance packs.	ALBPM_HOME\edition\update\updateHistory.log

Log Severity Levels

AquaLogic BPM allows you to define logging levels to specify the level of detail of the information stored in the ALBPM logs.

Log Level	Description
Fatal	Specifies a serious error that may cause the application to fail.
Severe	Specifies a serious error that may or may not cause the application to fail.
Warning	Specifies a potentially harmful situation but generally does not pose a threat to the stability of an application.
Info	<p>Specifies informational messages that highlight the progress of the application at a high level. These can include:</p> <ul style="list-style-type: none"><li>• Changes in the engine state, including: start, stop, and restart.</li><li>• Changes in state of engine services.</li><li>• Changes in engine properties.</li><li>• Changes in the state of a process deployed on the engine, including: startup, deployment, redeployment, and deprecation.</li><li>• Actions of participants</li><li>• Work executed by the engine automatically.</li></ul>
Debug	<p>Specifies informational messages that highlight the process instances at a lower level. These can include:</p> <ul style="list-style-type: none"><li>• Tracing a process instance, including: instance creation, changing activities, routing, and locks.</li><li>• Changes in the state of an instance, including: running, selection, activity completion, and exceptions.</li><li>• Actions on a process, including: executing a task, executing an activity, and executing a ToDo Item.</li></ul>

Process Execution Engine Log

The Process Exeuction Engine Log contains information about the Process Execution Engine.

Configuring Process Execution Engine Logs

1. Launch the Process Execution Engine

2. Select **Engines**
3. Select the engine whose log files you want to configure.
4. Configure the Log Directory.

If you do not want to use the default location of the Process Execution Engine logs, you can change the directory where the log files are stored.

- a) Select the **Basic Configuration** tab
- b) Specify the **Log Directory** property

This is a path to the directory containing your engine log files.

- c) Click **Save**.

5. Configure logging properties.
  - a) Review [Logging Levels and Engine Performance](#) on page 32 to determine the logging level appropriate to your environment.
  - b) Select the **Log** tab.

## Viewing Engine Log Files with the Log Viewer Application

The Log Viewer application provides a graphical interface for viewing the ALBPM engine log files.

1. Launch the Log Viewer application.
2. Open a log file

**Select File ► Open**

Allows you to browse the file system to view an engine log on the local file system.

**Select File ► Open Remote Log**

Allows you to connect to an engine remotely to download the engine log.


**Select File ► Open Recent**

Allows you to select from a list of recently viewed log files.

3. Filter the log messages
4. View an individual log message

## Process Execution Engine Log Description

In addition to the log messages, the Process Execution Engine log stores information about the log message. This information is divided into columns.

Column	Description
Severity	Displays the severity level of the log message. See <a href="#">Log Severity Levels</a> on page 30.
Message	Displays the text of the log message.
Time	Displays the time, in seconds, the log entry was created.   <b>Note:</b> Within the log file, the value is stored in GMT-0 format. The Log Viewer and Log Reader applications display the time of the log entry in local time.
Date	Displays the date the log entry was generated.
Application	Displays the application that generated the log message. The possible values for this field are: <ul style="list-style-type: none"> <li>• Default: messages generated by the Process Execution Engine.</li> <li>• CIL: messages generated by project code.</li> </ul>

Column	Description
Module	Displays the module that generated the log message.
Thread	Displays the name of the thread that generated the log entry.

**Logging Levels and Engine Performance**

You can improve Engine performance by configuring logging levels and log file size.

When the engine logging level is not configured correctly, it can reduce engine performance. The engine can distinguish between engine log messages and log messages generated by PBL methods.

**Log File Size and Number**

You can configure the maximum engine log file size as well as the maximum number of log files used by an engine. In general, it is better to have multiple small logs instead of fewer large logs. The time it takes for the file system to write large files can reduce engine performance.

By default the engine log file is set to 2KB.

**Engine Log Severity**

Setting the logging severity correctly can help engine performance. You should set the logging severity according to the needs of your environment.

In a development environment, you may want to include as much login information as possible. However, in a production environment you may only want to generate more severe log messages to reduce log files sizes.

**Logging Messages from PBL Methods**

PBL methods can be configured to send messages to the engine log files when events are encountered. To ensure that logs generated by PBL methods are correctly captured in the engine log, you should severity argument of the logMessage method. In the following example the log severity is set to DEBUG.

```
logMessage "Executing Initial Customer Info" using severity = DEBUG
```

If the severity level for PBL-generated error messages is set to DEBUG, then this log will appear in the log file. However, if it is set to a higher level, then this message will not appear.

**External Resources**

External Resources provide a common method for connecting to other resources in an enterprise including databases, Web Services, etc. External Resources are used to define connectivity and configuration information.

It is useful to define these separately since connectivity and configuration information is different between systems. This allows you to easily deploy a project into a new environment because you only need to redefine the External Resource without having to edit application code.

**Creating an External Resource Using Process Administrator**

- 1. Launch Process Administrator
- 2. Select **External Resources**
- 3. Click**Add**.
- 4. Enter the following information:

Option	Description
<b>Name</b>	Specifies the name of the external resrouce.



Option	Description
Type	Specifies the type of external resource. See <a href="#">External Resources</a> on page 32 for information on the types of external resources available.
Subtype	Specifies the subtype for the external resource.

5. Click **Next**.

6. Enter the information for your external resource.

The information you enter depends on the type of external resource you are creating. See [External Resource Reference](#) on page 33 for more information.

7. Click **Save**.

The external resource is created and appears in the list of available external resources.

## External Resource Reference

The following sections provide detailed information about the configuration options for each External Resource type.

### SQL Database

Provides detailed information for the SQL Database External Resource.

#### General Properties

This section defines the general properties for this External Resource:

Property	Description
Name	Defines the name of the external resource.
Type	Specifies the type of external resource.
Supported Types	Specifies the type of JDBC connection which corresponds to the database vendor and version number.

### 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 Informix Driver Properties

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

### Basic

Property	Description
Host	Specifies either the IP address or the server name (if your network supports named servers) of the primary database server. For example, 122.23.15.12 or InformixServer.
Port	Specifies the TCP port on which the database server listens for connections. The default varies depending on operating system.
User	Specifies the case-insensitive default user name used to connect to your Informix database.
Password	Specifies a case-insensitive password used to connect to your Informix database.
Database	Specifies the name of the database to which you want to connect.
Server	Specifies the name of the Informix database server to which you want to connect.
URL	Defines the URL format used to connect to your database.

### Advanced

Property	Description
Root dbspace name	Specifies the root dbspace of your Informix database. The rootdb space is the initial dbspace created by the Informix server. The root dbspace contains reserve pages and internal tables.

## 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.

**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.

Property	Description
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 Sybase Driver Properties

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

#### Basic

Property	Description
Host	
Port	
User	
Password	
Database	
Device	
Allocation Size	
URL	

### Properties Tab

The **Properties** tab allows you to define name/value pairs to configure database properties.



**Note:** All connection property names are case-insensitive.

#### Runtime Tab

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 AS/400 JDBC Properties

You can configure the following properties for your DB2 AS/400 database:

**Basic**

Property	Description
Host	
Port	
User	
Password	
Database	
Schema	
URL	

**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	
Maximum Connections Per User	
Connection Idle Time (minutes)	
Minimum Pool Size	
Maximum Opened Cursors	

**BEA DB2 OS390 Properties**

You can configure the following connectivity properties for your DB2 OS390 database:

**Basic**

Property	Description
User	
Password	
Database	
Schema	
URL	

**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.

## Derby Database Driver Properties

You can specify the following properties for your Derby database:

### Basic

Property	Description
Database Path	
Requires Authentication	
User	
Password	
Schema	
URL	

## 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.

**Generic JDBC Version 1 Properties**

You can specify the following connectivity properties for your generic JDBC driver:

**Basic**

Property	Description
JDBC Driver	
URL	
User	
Password	

**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.

**Remote JDBC Properties**

You can specify the following connectivity properties for a remote JDBC connection:

Property	Description
Database Type	
J2EE	
Lookup Name	

**SAP Service**

You can specify the following connectivity properties for an SAP Service.

**General Properties**

This section defines the general properties for this External Resources:

Property	Description
Name	Defines the name of this external resource.
Type	Specifies the external resource type.
Supported Types	

**SAP Properties**

The following properties must be for an SAP Service:



Property	Description
Service Name	
Client	
User Id	
Password	
Hostname	
System Number	
Language	
Pool Size	

### Web Service

Provides detailed information for configuring a Web Service.

### General Properties

This section defines the general Web Service properties:

Property	Description
Name	Defines the name of the external resource.
Type	Specifies the type of external resource.
Supported Types	Specifies the type of Web Service.

### Producer Web Service Properties

The following properties must be configured for a Producer Web Service External Resource:

Property	Description
Authentication Type	<p>Defines the authentication type of this web service.</p> <ul style="list-style-type: none"> <li>None - uses no authentication for the web service.</li> <li>Username Token Profile -</li> </ul>

### Consumer Web Service Properties

The following properties must be configured for a Consumer Web Service under the **Endpoint** tab:

Property	Description
Static Endpoint Binding	
UDDI Dynamic Endpoint Binding	
Transport Type	
Transport Configuration	
Use System Exceptions	

The following properties must be configured for a Consumer Web Service under the **Security** tab:

Property	Description
Send Username Token	
Send Nonce and Timestamp	
Username	
Password	
Confirm Password	

**HTTP Server Configuration**

Provides detailed information for configuring an HTTP Server as an External Resource.

**General Properties**

Property	Description
Name	Defines the name of this external resource.
Type	Specifies the external resource type.

**HTTP Server Properties**

The following properties must be defined for an HTTP Server configuration:

Property	Description
Protocol	Specifies the protocol for the HTTP server configuration. Possible values are: <ul style="list-style-type: none"><li>• HTTP</li><li>• HTTPS</li></ul>
Host	Specifies the hostname or IP address of the HTTP server.
Port	Specifies the port number where the HTTP server listens for requests. The default is 85.
Requires HTTP Basic Authentication	Specifies whether the HTTP server requires basic HTTP authentication.
User	Specifies the username used to authenticate HTTP requests.
Password	Specifies the password used to authenticate HTTP requests.

**Microsoft .Net Service**

**Mail Outgoing Service**

Provides detailed information for configuring a Mail Outgoing Service.

**General Properties**

This section defines the general properties for this External Resource:

Property	Description
Name	Defines the name of this external resource.
Type	Specifies the external resource type.

Property	Description
Supported Types	

### SMTP Properties

The following properties must be for an SMTP Mail Outgoing Service:

Property	Description
Server Host	
Server Port (optional)	
User	
Password	
Secure Connection	<p>Defines the security protocol used. Valid values are:</p> <ul style="list-style-type: none"> <li>No - No security protocol is used.</li> <li>TLS, if available</li> <li>TLS</li> <li>SSL - Uses the Secure Sockets Layer (default).</li> </ul>

### J2EE Application Server

Provides detailed information on configuring a J2EE Application Server as an External Resource.

To create Enterprise JavaBeans components, you must define a J2EE Application Server as an External Resource.

### General Properties

The following table defines the general properties for this External Resource:

Property	Description
Name	Defines the name of the external resource.
Type	Specifies the type of external resource.
Supported Types	Specifies the type of J2EE server

### Local J2EE Application Server Properties

This type of J2EE application server is used when the process is deployed in a J2EE environment, and the resources are located in the same Application Server

The following properties must be configured for a Remote JDBC Connection:

Property	Description
User Transaction Lookup Name	

### Generic J2EE Application Server Properties

The following properties must be configured for a J2EE Application Server using the **Basic** tab:

Property	Description
Initial Context Factory	
URL	

Property	Description
Principal	
Credentials	

The following properties must be configured for a J2EE Application Server under the **Advance** tab:

Property	Description
User Transaction Lookup Name	

The following properties must be configured for a J2EE Application Server using the **Properties** tab:

Property	Description
Initial Context Factory	

The following properties must be configured for a J2EE Application Server using the **Runtime** tab:

Property	Description
Maximum Pool Size	
Maximum Connections Per User	
Connection Idle Time (minutes)	
Minimum Pool Size	

**Enterprise JavaBean (EJB)**

Provides detailed information for configuring an Enterprise JavaBean as an External Resource.

**General Properties**

This section defines the general properties for this External Resource:

Property	Description
Name	Defines the name of this external resource.
Type	Specifies the external resource type.
Supported Types	

**EJB Properties**

The following properties must be defined for an EJB:

Property	Description
J2EE	
Lookup Name	

**Java Class Library**

Provides detailed information for configuring a Java Class Library as an External Resource.

**General Properties**

This section defines the general properties for this External Resource:

Property	Description
Name	Defines the name of this external resource.
Type	Specifies the external resource type.
Supported Types	

### Java Class Library Properties

The following properties must be defined for a Java Class Library:

Property	Description
property	
property	

### AquaLogic Service Bus

Provides detailed information for configuring an AquaLogic Service Bus as an External Resource.

#### General Properties

This section defines the general External Resource properties:

Property	Description
Name	Defines the name of the external resource.
Type	Specifies the type of external resource.
Supported Types	Specifies the type of AquaLogic Service Bus connection.

### Management Host Properties

The following properties must be configured for a Management Host:

Property	Description
Host	
Port	
User	
Password	

### Proxy Service Properties

The following properties must be configured for a Proxy Service:

Property	Description
Host	
Port	
User	
Password	

### Process Deployment Properties

The following properties must be configured for a Process Deployment:

Property	Description
Management Configuration	
Project Name	
WSDL Folder	
Business Services Folder	
WS-Security Account	
Transport	
Host	
Port	

**Mail Incoming Service**

Provides detailed information for configuring a Mail Incoming Service as an External Resource.

**General Properties**

This section defines the general properties for this External Resource:

Property	Description
Name	Defines the name of this external resource.
Type	Specifies the external resource type.
Supported Types	

**IMAP and POP3 Properties**

The following properties must be when using IMAP or POP3:

Property	Description
Service Host	
Server Port (optional)	
User	
Password	
Secure Connection	
Secure Authentication	

**Microsoft .NET Service**

Provides detailed information for configuring a Microsoft .NET Service.

**General Properties**

This section defines the general properties for this External Resources:

Property	Description
Name	Defines the name of this external resource.
Type	Specifies the external resource type.

## Microsoft .NET Service Properties

The following properties must be defined for a Microsoft .NET Service:

Property	Description
Host	Defines the location of the .NET Bridge host
Port	Defines the port used by the .NET Bridge host

## JMX Service

AquaLogic BPM supports the following JMX service providers:

- BEA Weblogic
- IBM WebSphere
- JBoss
- JSR-160
- MX4J

## Basic

Property	Description
Host	

## Advanced

### CORBA Service

Provides information on configuring a CORBA Service as an External Resource.

Studio allows you to catalog CORBA objects that reside in an Interface Repository. Once cataloged, you can manipulate the components of the CORBA object in your Method tasks in a process design. To catalog a CORBA object, you need a configuration of CORBA type. Note that creating a configuration allows you to reuse it each time you need to add new components or to reintrospect existing ones

## General Properties

This section defines the general SQL Database properties:

Property	Description
Name	Defines the name of the external resource.
Type	Specifies the type of external resource.
Supported Types	

## CORBA Service Properties

The following properties must be configured for a CORBA Service under the **Basic Settings** tab:

Property	Description
Use Application Server Orb	Determines whether BEA Systems should lookup the default ORB when running on an application server. If unchecked, the default ORB is used.

The following properties must be configured for a CORBA Service under the **Naming Service** tab:

Property	Description
Read IOR from URL	if you have the IOR exported to some service (for example, a web server) and a URL exists that can be used to fetch it
Use This IOR	Allows you to explicitly define an IOR. (recommended)
Resolve Initial Reference	to request the ORB to get the reference of the service trying to resolve its reference. Note that this option is not recommended since it has interoperability problems when using different ORB
Do Not Use Naming Service	select this option if you do not need this service. However, remember that objects used in methods are found using the Naming Service. Hence, you will need to use the IOR for any object to be used in a method - passing it as a parameter in its constructor

The following properties must be configured for a CORBA Service under the **Interface Repository** tab:

Property	Description
Read IOR from URL	if you have the IOR exported to some service (for example, a web server) and a URL exists that can be used to fetch i
Use This IOR	Allows you to explicitly define an IOR (recommended)
Resolve Initial Reference	to request the ORB to get the reference of the service trying to resolve its reference. Note that this option is not recommended since it has interop problems when using different ORB
Use Interface Repository	this service can be used when an ORB does not have an implementation of the Interface Repository service. Note that this service must be launched separately
Host	
Port	

### JMS Messaging Service

Provides detailed information for configuring a JMS Messaging Service as an External Resource.

#### General Properties

This section defines the general properties for this External Resources:

Property	Description
Name	Defines the name of this external resource.
Type	Specifies the external resource type.
Supported Types	

#### JMS Messaging Service Properties

The following properties must be configured for a JMS Messaging Service:

Property	Description
J2EE	
Destination Type	<ul style="list-style-type: none"> <li>Queue</li> </ul>



Property	Description
	<ul style="list-style-type: none"> <li>• Topic</li> </ul>
Connection Factory Lookup	
JMS Listener Port (WebSphere only)s	

## JNDI Directory Server

Provides detailed information on configuring a JNDI Directory Server as an External Resource.

### General Properties

This section defines the general properties for this External Resource:

Property	Description
Name	Defines the name of the external resource.
Type	Specifies the type of external resource.
Supported Types	Specifies the type of JNDI Directory Server.

### Active Directory Properties

The following properties must be configured for an Active Directory JNDI Directory Server using the **Basic** tab:

Property	Description
Initial Context Factory	Defines the name of the initial context factory you want to use.
URL	Defines the URL you want to use to connect to the directory service.
Principle	Defines the root distinguished name for the directory service.
Credentials	Specifies the password for the directory service.
Referrals	<ul style="list-style-type: none"> <li>• follow: the entry will be looked for directly.</li> <li>• ignore: the entry is not looked for.</li> <li>• throw: you must catch and manage any exceptions.</li> </ul>

The following properties must be configured for an Active Directory JNDI Directory Server using the **Properties** tab:

Property	Description
Properties	Define any name/value pair properties that need to be passed to the directory service.

The following properties must be configured for an Active Directory JNDI Directory Server using the **Runtime** tab:

Property	Description
Maximum Pool Size	
Maximum Connections Per User	
Connection Idle Time (minutes)	
Minimum Pool Size	

## Sun One LDAP Properties

The following properties must be configured for a Sun One LDAP JNDI Directory Server using the **Basic** tab:

Property	Description
Initial Context Factory	Defines the name of the initial context factory you want to use, for example: com.sun.jndi.ldap.LdapCtxFactory.
URL	Defines the URL you want to use to connect to the directory service.
Principle	Defines the root distinguished name for the directory service.
Credentials	Specifies the password for the directory service.
Referrals	<ul style="list-style-type: none"> <li>follow: the entry will be looked for directly.</li> <li>ignore: the entry is not looked for.</li> <li>throw: you must catch and manage any exceptions.</li> </ul>

The following properties must be configured for a Sun One LDAP JNDI Directory Server using the **Properties** tab:

Property	Description
Properties	Define any name/value pair properties that need to be passed to the directory service.

The following properties must be configured for a Sun One LDAP JNDI Directory Server using the **Runtime** tab:

Property	Description
Maximum Pool Size	
Maximum Connections Per User	
Connection Idle Time (minutes)	
Minimum Pool Size	

## Java Process Definition (JPD)

Provides detailed information for configuring a Java Process Definition as an External Resource.

### General Properties

This section defines the general properties for this External Resources:

Property	Description
Name	Defines the name of this external resource.
Type	Specifies the external resource type.
Supported Types	

### Java Process Definition Properties

The following properties must be configured for a Java Process Definition:

Property	Description
HTTP Server Configuration	

Property	Description
Path	

# Advanced Administration

---

## Auditing

ALBPM Processes provide auditing capabilities by recording information about the occurrence of sensible events during the execution of the process.

Audit Events allow you to keep track of the events that occur while a process instance is flowing through the process. An event is registered each time the instance enters or exits an activity (simple activity, group, process). The Process Execution Engine generates one event per action each time an even enters or exits an activity. By default events are only generated for interactive activities.

### Enabling Auditing

Auditing properties are set in two locations within ALBPM:

- As a property of an activity within a business process.
- Globally for all published projects using Process Administrator.

### Enabling Auditing Using Process Administrator

You can enable auditing globally for all published project using Process Administrator.

1. Launch the Process Administrator
2. Click the Process Execution Engine where you want to configure auditing.
3. Click the **Others** tab.
4. Select one of the following options under **Store Events**

Option	Description
<b>Depends on Process</b>	Generates events only for those processes configured to be archived in Studio. This is the default.
<b>Never</b>	No audit records are generated. Overrides the setting specified within the process.
<b>Always</b>	Generates audit records for all events. Overrides the settings specified within the process.



**Note:** Setting **Store Events** to this option means that audit records are kept for all activities in all processes. This can lengthen transaction times and decrease performance.

5. Click **Save**.

Audit records are generated according to your selection for every process deployed on the Process Execution Engine.

### Which Audit Events are Generated

The following auditing events are generated:

- All the activities generate the same events (IN, OUT, EXECUTE, SELECT, UNSELECT, among others.)
- The Begin activity has no Activity IN event as the instance is created in it.
- The End activity has no Activity OUT event as the instance terminates there.
- The Join activity generates events only if the Split associated activity generates events.
- When an instance is created, a CREATION event is generated instead of an Enter event. This event is always automatically generated if the Engine stores events. All original instances (copy 0) have the CREATION event.

- When an instance is finished, an END event is generated. This event is always automatically generated if the Engine stores events. All terminated original instances (copy 0) have the END event.
- Interactive activities have additional events that occur between the Enter and End events.

If you have any Generates events check box selected, the Audit Trail in WorkSpace is enabled. The Audit Trail displays all events that have occurred for an instance.

## ALBPM Backup Guidelines

### ALBPM Files

File Type	Description	Location
Configuration Files		\$INST_DIR/enterprise/conf
Non-versionable JAR Files	Java libraries that are not part of a project file.	<ul style="list-style-type: none"> <li>• <i>INSTALL_DIR</i>/webapps/webconsole/WEB-INF/lib</li> <li>• <i>INSTALL_DIR</i>/ext</li> <li>• <i>INSTALL_DIR</i>/webapps/portal/WEB-INF/lib</li> </ul>
Properties Files		../directory.properties
Web Applications	Customized application files, including CSS files, custom JSPs, and image files.	
WorkSpace	Custom JSP used by WorkSpace	\$INST_DIR/webapps/portal/WEB-INF/portal.properties
WorkSpace Administrator	Workspace Administrator properties file	\$INST_DIR/webapps/portaladmin/WEB-INF/portaladmin.properties
Archive Viewer	Property and configuration files for the Archive Viewer application.	<ul style="list-style-type: none"> <li>• <i>\$INST_DIR</i>/webapps/archive/WEB-INF/archive.properties</li> <li>• <i>\$INST_DIR</i>/webapps/archive/WEB-INF/conf/Default</li> </ul>

### Databases

Database	Description
Engine Database	
Archive Database	
BAM Database	

## Security

### Best Security Practices

This section summarizes security concerns for AquaLogic BPM deployments. Note, however, that it is not intended to replace the services of a qualified security professional.

#### Determine Your Security Needs

This section describes best practices for determining the security needs of your AquaLogic BPM deployment.

## Understand Your Environment

To better understand your security needs, ask yourself the following questions:

- Which resources am I protecting?

Many resources in the production environment can be protected, including information in databases accessed by AquaLogic BPM and the availability, performance, applications, and the integrity of the website. When deciding the level of security to provide, consider the resources you want to protect .

- From whom am I protecting the resources?

For most websites, resources must be protected from everyone on the Internet. But should the website be protected from the employees on the intranet in your enterprise? Should your employees have access to all resources within the AquaLogic BPM environment? Should system administrators have access to all AquaLogic BPM resources? Should system administrators have access all data? You might consider giving access to highly confidential data or strategic resources to only a few well trusted system administrators. Perhaps it would be best to allow no system administrators access to the data or resources.

- What will happen if protections on strategic resources fail?

In some cases, a fault in your security scheme is easily detected and considered nothing more than an inconvenience. In other cases, a fault might cause great damage to companies or individual clients that use the website. Understanding the security ramifications of each resource will help you protect it properly.

## Hire Security Consultants or Use Diagnostic Software

Whether you deploy AquaLogic BPM on the Internet or on an intranet, it is a good idea to hire an independent security expert to go over your security plan and procedures, audit your installed systems, and recommend improvements. BEA partners offer services and products that can help you to secure an AquaLogic BPM production environment. For details, see the BEA Partner's Page at <http://www.bea.com/partners>.

## Read Security Publications

For the latest information about securing web servers, BEA recommends the 'Security Practices & Evaluations' information available from the CERT™ Coordination Center operated by Carnegie Mellon University.

Report possible security issues in AquaLogic BPM by contacting AquaLogic BPM technical support. For technical support contact information, see [BEA Documentation and Resources](#).

## Some Common Tasks and the Components to Perform Them

The following table lists some of the more common security-related tasks of an administrator and the component you use to perform each one.

Task	Component
<b>Before You Begin</b>	
Configuring databases for the Directory Service and Process Execution Engine	Admin Center
Configuring the archive database and logging Into Archive Viewer	Archive Viewer
<b>Auditing</b>	
Configuring the generation of audit records for a process execution engine	Process Administrator
Modifying the generation of audit records for a process execution engine	Process Administrator
Using Log Viewer	Log Viewer
<b>Administering Participants</b>	
Creating a Role during Runtime by Using BEA AquaLogic BPM Process Administrator	Process Administrator
Adding a Participant During Runtime by Using Process Administrator	Process Administrator

Task	Component
Viewing and Modifying the Properties of a Participant During Runtime	Process Administrator
Assigning Roles to Participants	
Setting Permissions for Participants	
Setting Categories for Participants	
Specifying parametric roles	
Configuring Permissions for Participants	Process Administrator
<b>Administering Process Execution Engines</b>	
Adding a Process Execution Engine	Process Administrator
Enabling and Disabling a Process Execution Engine	Process Administrator
Publishing and Deploying a Project	Process Administrator
Deploying a Project after Publishing	Process Administrator
Undeploying a Project	Process Administrator
Querying audit records	WorkSpace
<b>Logging In</b>	
Logging in to Process Administrator	Process Administrator

## Ensuring the Security of Your Production Environment

This section provides high-level descriptions of the security measures that can be employed to secure your AquaLogic BPM environment.

### Securing the AquaLogic BPM Hosts

An AquaLogic BPM production environment is only as secure as the machines on which it is running. It is important that you secure the physical machine, the operating system, and all other software that is installed on the host machine. The following are suggestions for securing your AquaLogic BPM host in a production environment. Also check with the manufacturer of the machine and operating system for recommended security measures.

Security Action	Description
Physically secure the hardware.	Keep your hardware in a secured area to prevent unauthorized operating system users from tampering with the deployment machine or its network connections.
Secure the networking services that the operating system provides.	<p>Have an expert review network services such as e-mail programs or directory services to ensure that a malicious attacker cannot access the operating system or system-level commands. The way you do this depends on the operating system you use.</p> <p>Sharing a file system with other machines in the enterprise network imposes risks of a remote attack on the file system. Be certain that the remote machines and the network are secure before sharing the file systems from the machine that hosts AquaLogic BPM components.</p>
Use a file system that can prevent unauthorized access.	Make sure the file system on each AquaLogic BPM component host can prevent unauthorized access to protected resources. For example, on a Windows computer, use only NTFS.

Security Action	Description
Set file access permissions for data stored on disk.	<p>Set operating system file access permissions to restrict access to data stored on disk. This data includes, but is not limited to, the following:</p> <ul style="list-style-type: none"> <li>• Third-party authentication directories</li> <li>• Portal configuration files</li> </ul> <p>For example, operating systems such as Unix and Linux provide utilities such as <code>umask</code> and <code>chmod</code> to set the file access permissions. At a minimum, consider using <code>'umask 066'</code>, which denies read and write permissions to Group and Others.</p>
Set file access permissions for data stored in the portal database.	Set operating system file access permissions to restrict access to data stored in the portal database.
Safeguard passwords.	The passwords for user accounts on production machines should be difficult to guess and should be guarded carefully. Set a policy to expire passwords periodically. Never code passwords in client applications.
Do not develop on a production machine.	Develop first on a development machine and then move code to the production machine when it is completed and tested. This process prevents bugs in the development environment from affecting the security of the production environment.
Do not install development and sample software on a production machine.	Do not install development tools on production machines. Keeping development tools off the production machine reduces the leverage intruders have should they get partial access to an AquaLogic BPM production machine. Do not install the AquaLogic BPM sample applications on production machines.
Enable security auditing.	Configure security auditing to enable monitoring of sensitive portal functions using the Audit Manager function.
Consider using additional software to secure your operating system.	<p>Most operating system can run additional software to secure a production environment. For example, and Intrusion Detection System (IDS) can detect attempts to modify the production environment.</p> <p>Refer to the vendor of your operating system for information about available software.</p>
Apply operation-system service packs and security patches.	Refer to the vendor of your operating system for a list of recommended service packs and security-related patches.
Apply the latest AquaLogic BPM maintenance packs and implement the latest security advisories.	<p>If you are responsible for security related issues on your site, register on the AquaLogic BPM Support Center page, <a href="http://support.plumtree.com">http://support.plumtree.com</a>, and subscribe to Support Alerts via email.</p> <p>In addition, you are advised to apply each maintenance pack as it is released. Maintenance packs are a roll-up of all bug fixes for each version of the product. You can download maintenance packs</p>





## Authentication of Administrators

Both ALBPM system administrators and user administrators must enter a username and password to access Process Administrator. Where ALBPM stores the passwords and permission information depend on the type of administrative user and directory service configuration.

Administrator Type	Location of password
ALBPM System Administrator	Is stored as part of the directory service configuration.  If you are using a database-only directory service this password can be changed using the Admin Center. If you are using a hybrid directory service, this password must be changed for the ALBPM Administrator user on your LDAP server.
ALBPM System Administrator (participant)	Is stored as part of the participant information.  If you are using a database-only directory service, this password can be changed using Process Administrator. If you are using a hybrid directory service, this password must be changed for this participant on your LDAP server.
User Administrator	Permissions and password are are stored as part of the participant information.  If you are using a database-only directory service, this password can be changed using Process Administrator. If you are using a hybrid directory service, this password must be changed for this participant on your LDAP server.

## Changing the ALBPM Administrator Password

1.

## Assigning Administrative Privileges to a Participant

You can assign administrative privileges to a participant within an organization.

1. Launch Process Administrator  
You must login as the ALBPM Administrator User
2. Select **Organization ► Participants**.
3. Select the participant you want to grant administrative privileges.
4. Click the **Administrator** checkbox.
5. Click **Save**.

## Assigning Administrative Privileges to LDAP Groups

Because ALBPM has read-only access to LDAP information, permissions for the ALBPM Administrator are not stored as part of the group or participant data within LDAP. To grant ALBPM Administration privileges to LDAP groups, you must add the permission information directly to the directory database.



**Note:** The following task describes procedures for adding ALBPM administrative access to LDAP groups by editing the directory service database. In general, however, you should never update the directory service or engine databases manually unless directed by ALBPM technical support.

1. Ensure that you have created an LDAP group containing participants who will have ALBPM administrator privileges.

2. Add a row to the FUEGO\_STRPROPS table of the directory service database.



**Note:** You must add one row to the FUEGO\_STRPROPS table for each LDAP group you want to have ALBPM administrator privileges.

The specific columns you must define are:

Column	Value
owner	null
category	FDI_PREFS
key	FUEGO_ADMINISTRATORS_PARTICIPANTS_GROUPS_DN.0 If you are adding multiple groups, change the trailing 0 to 1 ... n depending on the number of groups you are adding.
value	DN The distinguished name (DN) is the value of the group in your LDAP server. For example, CN=Gests,CN=Builtin,DC=server,DC=vmware,DC=fuegolabs,DC=com

After you have added this row to the directory database,

#### SQL Statement for Adding ALBPM Administrator Access in a Hybrid Directory Service

```
insert into DIR6WMAD2.FUEGO_STRPROPS (FUEGO_MODIFIER,
FUEGO_CRTIME,
FUEGO_STRVALUE,
FUEGO_CATEGORY,
FUEGO_KEY,
FUEGO_OWNER,
FUEGO_CREATOR,
FUEGO_MOTIME) values (
'Administrator',
SYSDATE,
'CN=group1,OU=groups,DC=server,
DC=vmware,DC=fuegolabs,DC=com',
'FDI_PREFS',
'FUEGO_ADMINISTRATORS_PARTICIPANTS_GROUPS_DN.0',
'dir6wmad2.fuego_participant-fuego_in-1',
'Administrator',
SYSDATE);
```

### Enabling User Administration for a Participant

You can allow a participant to manage other participants, including role and group assignments.

Before performing the following procedure, ensure that you have created the participant you want to have organizational administration access.

Enabling user administration for a participant allows them to create and update participants and roles without granting general administrative privileges.

1. Launch Process Administrator
2. Click **Organization**.
3. Click **Participants**.
4. Select the participant to whom you want to grant user administration.
5. Select **Enable User Administration**

6. Click **Save**.

## How a User Is Authenticated

The first time a user accesses a runtime component—for example, Process Administrator, WorkSpace, Archive Viewer--the Process Execution Engine authenticates that user as follows:

1. The Engine compares the security credentials provided in the component interface by the participant with those stored in the the directory service.
2. If the credentials match, then the Engine authenticates the user. This authentication is valid for the remainder of the session.
3. The Engine sends the message of authentication to the component.

## Monitoring

### Monitoring an Engine Using the HTTP Debugger

The HTTP debugger is an engine service that displays runtime information about your ALBPM Enterprise environment. The information provided by the HTTP debugger service is useful to troubleshoot problems

#### Starting the HTTP Debugger

Because the HTTP debugger service consumes system resources when running, it is turned off by default.

1. Launch Process Administrator
2. Click **Engines**.
3. Select the engine where you want to start the HTTP Debugger.
4. Click **Runtime Info**.
5. Enter a port number to access the HTTP Debugger.

The default port is 8080.



**Note:** 8080 is a commonly used port. If the HTTP debugger service does not start, try setting a different port.

You can start different instances of the HTTP Debugger for different engines, but you must specify a unique port for each instance.

6. Click **Start the HTTP Service**

## Monitoring Engine and Process Performance Using JMX

#### Enabling JMX (Standalone)

ALBPM Enterprise Standalone provides a basic JMX to view information about engine and process performance via MBeans.

1. Launch Process Administrator
2. Click **Engines**.
3. Select the engine where you want to enable JMX.
4. Select the **Services** tab.
5. Click the checkbox next to **HTTP Adaptor Enabled**.
6. Specify a port for the HTTP adaptor.



**Note:** You can enable the JMX HTTP adaptor for multiple engines. You must specify a unique port for each engine.

7. Click **Save and Apply** to save your changes.

You can connect to the HTTP-based JMX client with the URL: `http://engine_hostname:port`.

## Viewing Engine Information Using JMX (J2EE)

## 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 61 for more information.
3. Create the archive database
 

See [Creating the Archive Database](#) on page 62
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 62

## 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 62.

### Creating the Archive Database

Before performing the following procedures, ensure that you have enabled archiving. See [Enabling Archiving](#) on page 61 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 a 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	<p>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.</p> <p> <b>Note:</b> BAM Dashboards require the Flash Plugin.</p>

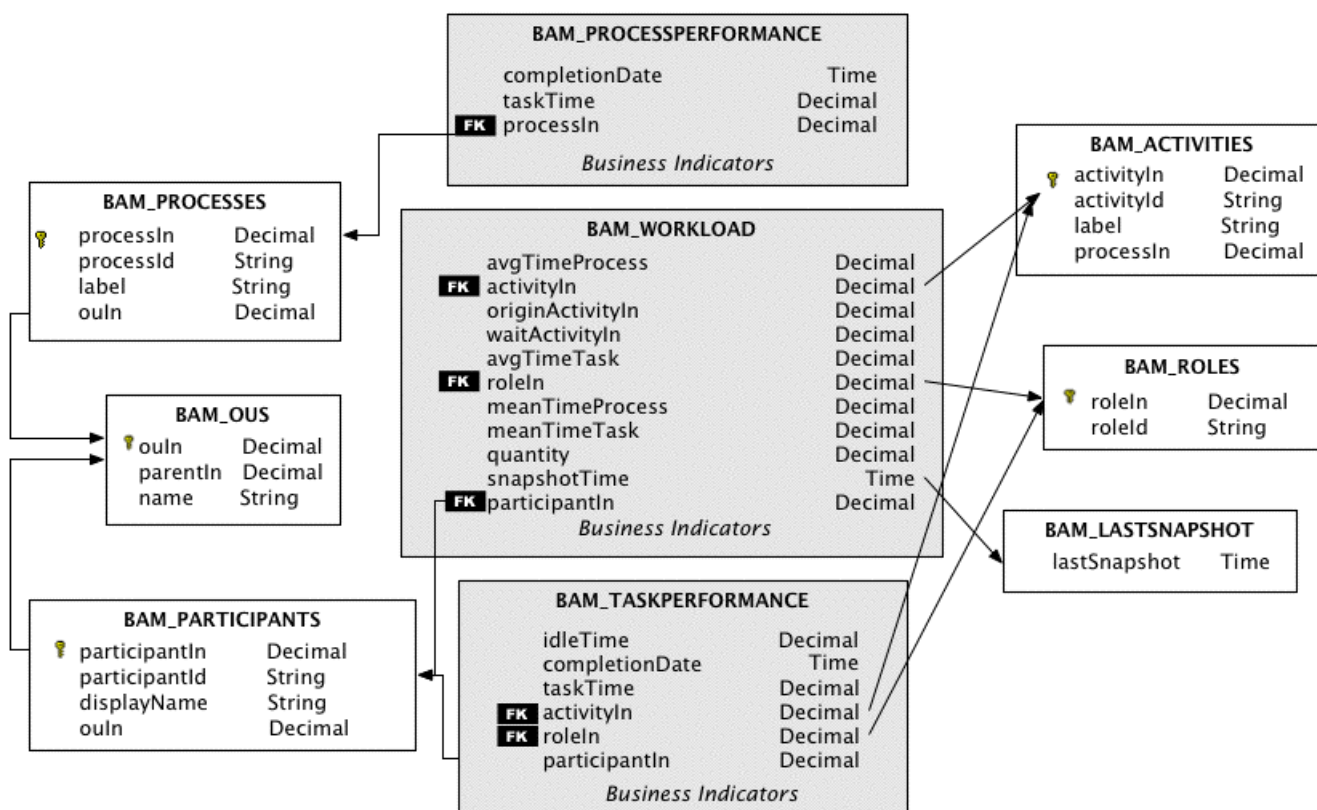
### BAM Database

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
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

Row Name	Value	NULL Value
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.
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	
<code>activityIn</code>	DECIMAL(10)	NOT NULL	
<code>roleIn</code>	DECIMAL(10)	NOT NULL	
<code>participantIn</code>	DECIMAL(10)	NOT NULL	
<code>completionDate</code>	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 <code>completionDate</code> is stored in GMT-0, as there might be different servers running with different hours.
<code>taskTime</code>	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
<code>processIn</code>	DECIMAL(10)	NOT NULL	
<code>completionDate</code>	TIMESTAMP	NOT NULL	Date when the instance reached the End activity of the process. To maintain the coherence between the data, the <code>completionDate</code> is stored as GMT-0 since there might be different servers running with different hours.
<code>taskTime</code>	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
<code>lastshapshot</code>	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`

## Managing Enterprise Applications Using JEE

### 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 Application EAR Files

- 1.

## 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.

## Configuring Engine Failover on Enterprise Standalone

ALBPM Enterprise allows you to .... what is failover in this case? What data is preserved, etc?

Primary and Backup Engines

- you can configure multiple engines to serve as the backup - when the primary engine fails, any of the back up engines will take over as the primary - when the failed engine is restarted, it joins the pool of backup engines - each backup engine communicates with the primary engine to see if it is still running - This communication occurs every 5 seconds - The primary engine to function as the primary engine as long as the following conditions are true: - The primary engine is running (what is the message sent?) - The primary engine returns the EngineBusyException - The primary engine status cannot be obtained because the host cannot be contacted - This can be caused by - an engine configuration problem - a java.net.UnknownHostException error - a java.net.MalformedURLException error - a java.net.NoRouteToHostException error

## Creating a Backup Location

1. Login to Process Administrator
2. Create a new Process Execution Engine
3. Select the Process Execution Engine you want to configure as a backup.
4. Select the **Basic Configuration** tab
5. Click **Locations**
6. Click **Add**
7. Enter connectivity information for the backup Process Execution Engine.



**Note:** You should use the actual hostname or IP address of the backup engine. You should not use localhost or a loopback IP address (for example, 127.0.0.1) because these will not work in a failover situation.

## Starting a Backup Location

1.

## Versioning Java Class Libraries

AquaLogic BPM Enterprise allows you to include Java libraries as part of the resources of an ALBPM project.

AquaLogic BPM has two ways of handling Java class library versions.

Versionable Libraries	<p>Libraries that may change after a process has been deployed in a production environment. When using this type of library, each version of a process must be associated with a specific version of a library. Library versioning is useful when you want to ensure that upgrades to a Java library do not change the behavior of an existing version of a process.</p> <p>Versionable class libraries are included as part of an exported project file. They are an integral piece of a project similar to other project resources.</p>
Non-versionable Libraries	<p>Non-versionable libraries are libraries that generally do not change after a process has been deployed. When changes are required they are propagated to all deployed process in the engine.</p> <p>Non-versionable libraries must be copied manually into the correct directories. They are stored in the following directories:</p> <ul style="list-style-type: none"> <li>• <code>INSTALL_DIR/webapps/webconsole/WEB-INF/lib</code></li> <li>• <code>INSTALL_DIR/ext</code></li> <li>• <code>INSTALL_DIR/webapps/portal/WEB-INF/lib</code></li> </ul>

## Referencing Processes on Other ALBPM Installations

### Mapping an External Process on ALBPM Enterprise

1.

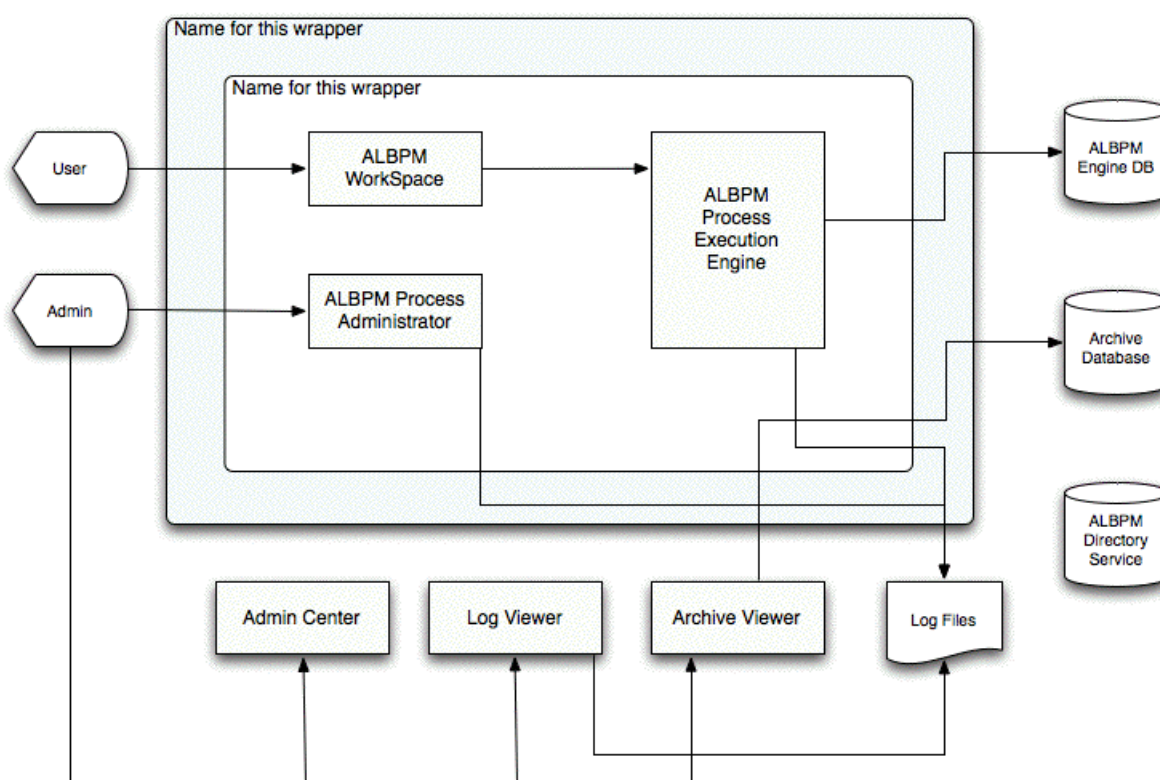
# AquaLogic BPM Enterprise Overview

## 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:

- 

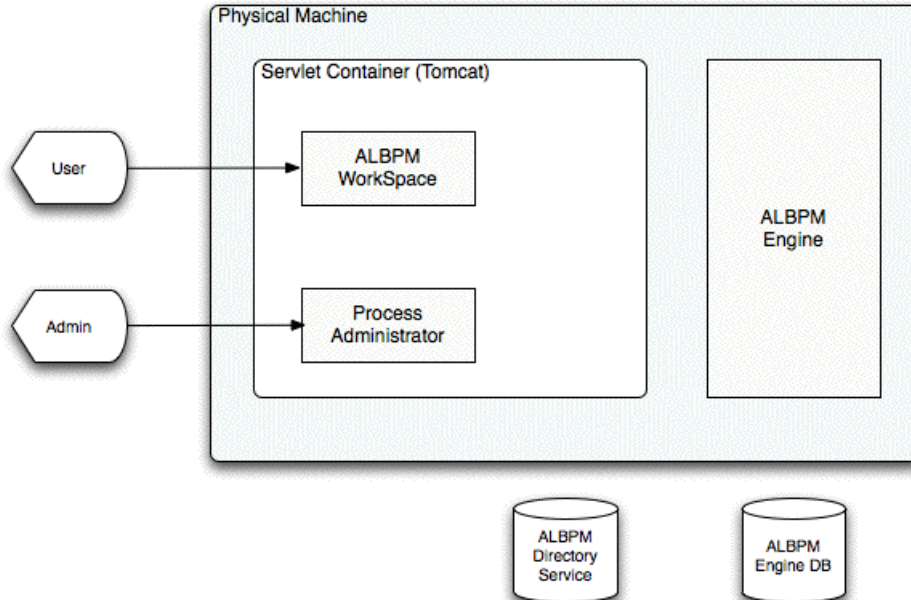
## ALBPM Supporting Applications

### Log Viewer Application

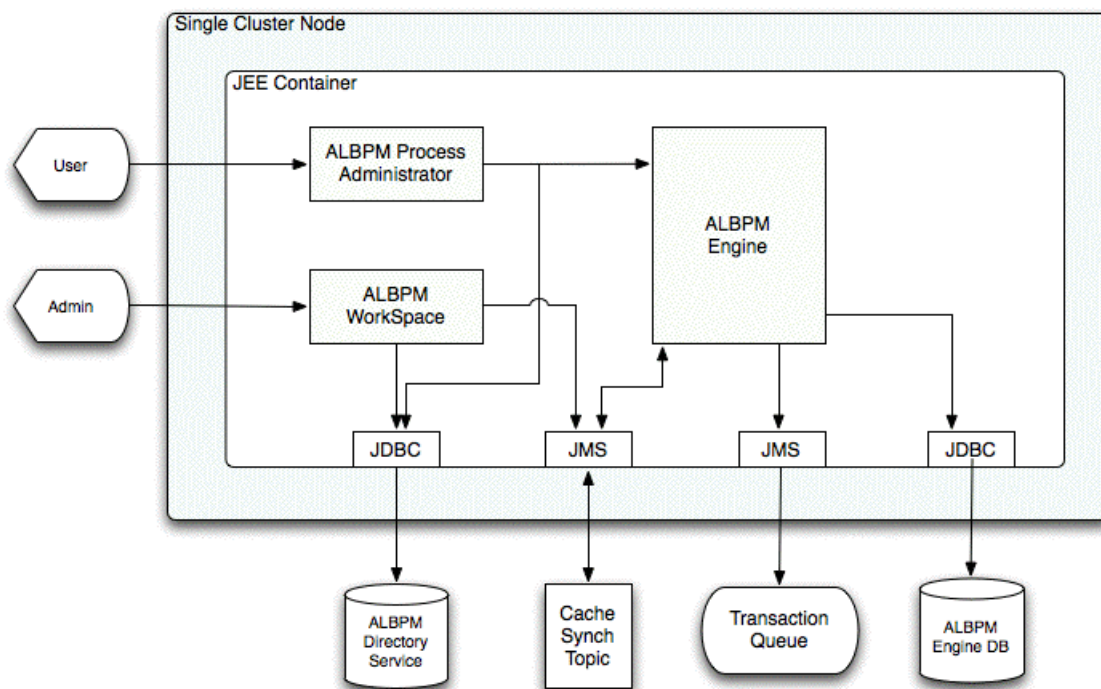
The Log Viewer is a Java-base utility for viewing the Process Execution Engine log files. It retrieves log records. It loads logs from the default log location.

## 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.

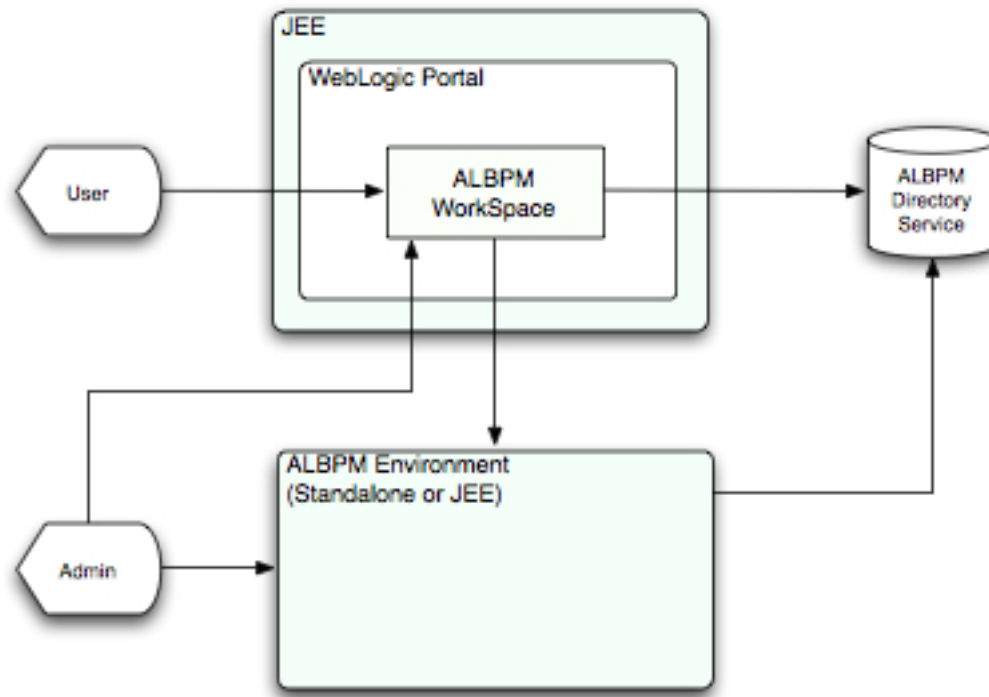
1. Schedules Asynchronous Transactions
2. Spreads Loads Across Multiple Nodes



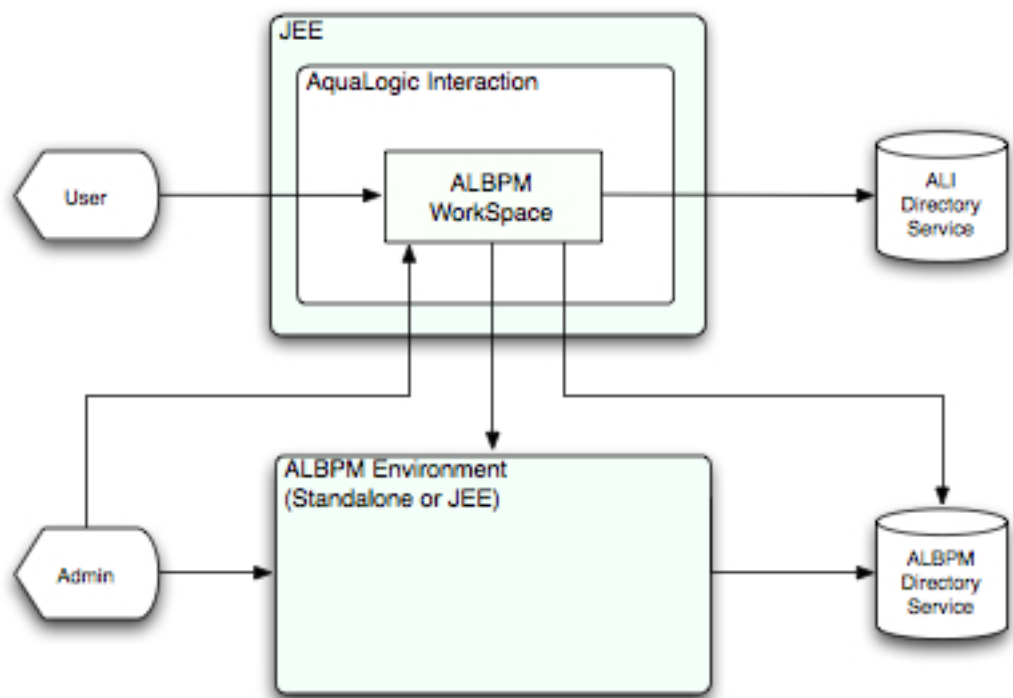
**Note:** The ALBPM JMS queue is not used to integrate with other applications. It is only used internally by the Process Execution Engine. External applications can not use this queue.

## JEE Cluster Architecture

### WebLogic Portal Architecture



ALI Architecture



# Administration Reference

The following sections provide reference information for ALBPM Enterprise administration, including application references and the ALBPM Configuration Wizard.

## Process Administrator Application Reference

The following topics provide general reference information for each screen within the Process Administrator application.

### General Actions

The following actions are available in the toolbar of some pages of the Process Administrator application.

Action	Description
Launch Log Viewer	Starts the Log Viewer application.
Reload	Refreshes the current page.
Help	Displays the HTML version of the <i>ALBPM Enterprise Administration Guide</i> .
Filter	Allows you to filter information displayed within Process Administrator.
Preferences	Defines application preferences.

## Organization

### Organization Tab

UI Element
Calendar Rule
Save
Cancel
Reset

### Import Tab

The **Import** tab allows you to import an existing organization into ALBPM Enterprise.

UI Element	Description
Category	Determines the element of an ALBPM Project you want to import.
From File	Allows you to specify the file you want to import.

### Category

Allows you to specify the specific elements of an organization that you want to import.

Element	Description
Organization	

Element	Description
Views	
External Resources	
Organization, Views, and Properties	

From File

Option	Description
Choose a file	
Password file output directory	

Export Tab

The **Export** tab allows you to export an existing organization from ALBPM Enterprise.

Category

Allows you to specify the specific elements of an organization that you want to import.

Element	Description
Organization	
Views	
External Resources	
Organization, Views, and Properties	

Advanced Properties

Option	Description
Include disabled participants	


Organizational Unit

The **Organizational Unit** tab displays existing organizational units and allows you to create new ones.

Action	Description
Add	Creates a new organizational unit
Delete	Deletes the selected organizational unit.

Add / Edit Organizational Unit

This page displays when you add or edit an organizational unit.

Property	Description
Parent ID	Specifies the parent for this organizational unit.   <b>Note:</b> You can only specify a parent when creating an organizational unit. After the organizational unit is created, you cannot modify this value.  If you need to define a hierarchy of organizational units, you must create higher-level units first. Then you can assign the parent units when creating lower level organizational units.
Name	Specifies the name of the organizational unit.
Description	Specifies a description for the organizational unit.
Calendar Rule	Allows you to associate a calendar rule with the organization.

## Roles

The **Roles** tab displays existing roles and allows you to create new ones.

Action	Description
Add	Creates a new role
Delete	Deletes the selected role.

## Add/Edit Role

This page appears when you add or edit a role.

Property	Description
Name	
Description	
Is Parametric	
Parametric Values	

## Advanced Properties

Property	Description
Calendar Rules	

## Groups

The **Group** tab displays existing groups and allows you to create new ones.

Action	Description
Add	Creates a new group
Delete	Deletes the selected group.

## Add/Edit Groups

This page appears when you add or edit a group.

Property	Description
Group ID	
Name	Specifies the name of the group.
Description	Specifies a description for the group.
Organization Unit for Administration Scope	Allows you to associate an organizational unit for this group. Only ALBPM User Administrators with access to this organizational unit can edit this group.

### Advanced Properties

Property	Description
Assigned Participants	Allows you to assign participants to this group.
Assigned Roles	Allows you to assign roles to this group.
Assigned Groups	Allows you to assign other groups to this group.


### Participants

The **Participants** tab displays existing groups and allows you to create new ones.


Action	Description
Add	Creates a new participant
Delete	Deletes the selected participant.

### Add/Edit Participants

This page appears when you add or edit a participant.

Property	Description
Enable	Enables a participant to login to WorkSpace and access any assigned process instances.   <b>Note:</b> When you want to restrict access to a participant, you should disable rather than delete a participant. Deleting a participant from an organization can have adverse effects on in-flight process instances.
User ID	Defines the user ID for this participant.
First Name	Specifies the participant's first name.
Last Name	Specifies the participant's last name.
Display Name	Specifies the participant's display name.
Telephone (optional)	Specifies the participant's telephone number.
Fax (optional)	Specifies the participant's fax number.
Notify by e-mail when new instances arrive	Allows the engine to send an email to the participant when a new instance arrives.
Mail	Specifies the participant's email address.
Password	Specifies the participant's initial password. Participants should change their password as soon as possible.



Property	Description
Confirm Password	Allows you to reconfirm the participant's password.
Organizational Unit	Defines the organizational unit for the participant. It is recommended that you specify an organizational unit for each participant. If you do not specify an organizational unit for a participant, the participant is a member of all organizational units.
Enable User Administration	Allows the participant to manage participants and role assignment. The participant is give permission to manage other participants in the same organizational unit.
Administrator	<p>Grants administrator access to the participant. The participant will have the same access as the ALBPM administrator. However, participants given administrative access can be assigned roles and groups.</p> <p> <b>Note:</b> If the ALBPM Administrator disables administrator privileges for a participant that is currently logged in, the participant will continue to have these permissions until they log out.</p>

### Advanced Properties

Property	Description
Change the Password	Allows you to change the password for the participant.
Assigned Roles	Allows you to assign roles to the participant.
Assigned Groups	Allows you to assign groups to the participant.
Absence Periods	Allows you to define absense periods for the participant.

### Holiday Rules

The **Holiday Rules** tab displays existing holiday rules and allows you to create new ones.

Action	Description
Add	Creates a new holiday rule
Delete	Deletes the selected holiday rule.

### Add/Edit Holiday Rules

This page appears when you add or edit a holiday rule.

Property	Description
Name	
Description	
Type	
Date	

### Calendar Rules

The **Calendar Rules** tab displays existing calendar rules and allows you to create new ones.

Action	Description
Add	Creates a new calendar rule
Delete	Deletes the selected calendar rule.

**Add/Edit Calendar Rules**

This page appears when you add or edit a calendar rule.

Property	Description
Name	Specifies the name of the calendar rule.
Time Zone	Specifies the time zone for the calendar rule.
Holiday Rule	Specifies the holiday rule for this calendar rule.
Work Schedule	Allows you to define the work schedule for this calendar rule. You can the day, start time, and finish time. Time values should be entered in 24 hour format.

**Business Parameters**

Action	Description
Add	Creates a new business parameter
Delete	Deletes the selected business parameter.

**Add/Edit Business Parameters**

Property	Description
Name	
Type	
Value	
Organizational Unit	

**Engines**

The **Engines** page allows you to create, configure, and manage Process Execution Engines.

**Engine Actions**













The following actions can be performed from the **Engines** page:

Action	Description
Add	Allows you to create and configure a new Process Execution Engine.
Delete	Deletes the selected engine.
Refresh Status	Refreshes the status for each engine as displayed in the status column.
Re-Load Informaton From the Directory	Refreshes information stored in the directory service.

Action	Description
Import	Allows you to import a previously exported Process Execution Engine configuration file.

## Engine Information

The **Engines** page displays a table listing all of the Process Execution Engines that have been created. This table displays the following information about each engine.

Row	Description										
Name	Displays the name of the Process Execution Engine. Select this name to edit the engine's properties.										
Status	Displays the current status of the engine. See <a href="#">Engine Status</a> on page 23 for more information.										
Type	Lists the version of ALBPM Enterprise for the engine. The possible values are: <ul style="list-style-type: none"> <li><b>weblogic:</b> The JEE version running on WebLogic Server</li> <li><b>websphere:</b> The JEE version running on WebSphere Application Server</li> <li><b>enterprise:</b> The standalone version of ALBPM.</li> </ul>										
Engine Actions	Lists the actions available for each engine. These are: <table> <tr> <th>Action</th><th>Description</th></tr> <tr> <td></td><td>Start the process execution engine. After selecting this action, the status column is updated to reflect the new engine status. If a problem occurs you can view the engine log for details.</td></tr> <tr> <td></td><td>Stop the process execution engine. After selecting this action, the status column is updated to reflect the new engine status.</td></tr> <tr> <td></td><td>View a thread dump for the process execution engine.</td></tr> <tr> <td></td><td>View the engine startup log.</td></tr> </table>	Action	Description		Start the process execution engine. After selecting this action, the status column is updated to reflect the new engine status. If a problem occurs you can view the engine log for details.		Stop the process execution engine. After selecting this action, the status column is updated to reflect the new engine status.		View a thread dump for the process execution engine.		View the engine startup log.
Action	Description										
	Start the process execution engine. After selecting this action, the status column is updated to reflect the new engine status. If a problem occurs you can view the engine log for details.										
	Stop the process execution engine. After selecting this action, the status column is updated to reflect the new engine status.										
	View a thread dump for the process execution engine.										
	View the engine startup log.										

## Basic Configuration Tab

The **Basic Configuration** page allows you to view and edit basic configuration properties for a Process Execution Engine. It also provides links to configure advance engine properties.

## Properties

You can view or define the following properties for a Process Execution Engine.

Property	Description
Name	Displays the name of the engine.
Type	Displays the version of ALBPM Enterprise for this engine.  When using a JEE version of Enterprise, the Cluster checkbox is visible. Selecting this checkbox enables the engine to support

Property	Description
	clustering. When clustering is enabled, the <b>Cluster</b> tab is displayed.
Home Directory	Defines the home directory for the engine. This directory contains resources used by the engine.
Log Directory	Defines the directory where the engine log files are stored.

### Edit Engine Database Configuration

The **Edit Engine Database Configuration** page allows you to define connectivity and runtime properties for the engine database.

The contents of this page depend on your database vendor. See [Engine and Directory Database Connectivity](#) on page 116 for a description of the connectivity and runtime properties for supported databases.

### J2EE Application Server Files

#### Export

Allows you to export the configuration properties for the process execution engine.

#### Manage Databases

The **Manage Databases** page allows you to perform operations on the engine and archive database.

### Database Creation

The options under **Database Creation** allow you to perform operations on the engine database.

Option	Description
Drop the database	Generates SQL commands to drop an existing engine database. You should select this option only if you want to remove an existing engine database before creating a new one.
Create the database	Generates SQL commands to create a new engine database.
Create the data structure	Generates SQL commands to create the required schema for the engine database.
Database Administrator User	Specifies the username of the administrator user.
Database Administrator Password	Specifies the password of the administrator user.

### Archiving Database

The options under **Archive Database Creation** allow you to perform operations on the engine database.



**Note:** This section is only visible if you have enabled archiving. See [Enabling Archiving](#) on page 61.

Option	Description
Drop the database	Generates SQL commands to drop an existing engine database. You should select this option only if you want to remove an existing engine database before creating a new one.
Create the database	Generates SQL commands to create a new engine database.
Create the data structure	Generates SQL commands to create the required schema for the engine database.
Database Administrator User	Specifies the username of the administrator user.
Database Administrator Password	Specifies the password of the administrator user.

## Manage Database Actions

Action	Description
Ok	Performs the database operation based on the options you have selected.
Cancel	Cancels the selected operations and returns to the <b>Edit Engine</b> page.
Show SQL Statements	Generates a .sql file based on the options you enter. If you do not have DBA privileges, you can give this file to your DBA to create the engine or archive database.

## Log Viewer

The **Log Viewer** displays the contents of the engine log withing Process Administrator.

This is similar to the information displayed by the log viewer application. See [Process Execution Engine Log Description](#) on page 31 for information on each of the columns displayed.

## Runtime Info (Standalone)

The **Runtime Info** page allows you to monitor Process Execution Engine usage.

Option	Description
Connected Clients	Displays a list of all users currently logged into the WorkSpace application.
Active Sessions	Displays the active sessions running on the engine.

## HTTP Debugger

The HTTP Debugger allows you to view information about the engine at runtime. After starting the HTTP Debugger service, you can access it by connectng to the URL of the service.

Option	Description
Status	
Last Login From	
Port	

## Engine Nodes (Standalone)

Add

## Basic Configuration

Option	Description
Host	Content
Home Directory	
Log Directory	
Protocol	<p>Specifies the protocol for the engine node. The following protocols are supported:</p> <ul style="list-style-type: none"> <li>TCP</li> <li>SSL</li> </ul>

Option	Description
Port	<ul style="list-style-type: none"> <li>• MMP</li> </ul>
Process Administrator Protocol	<p>Specifies the protocol used by the process administrator. The following protocols are supported:</p> <ul style="list-style-type: none"> <li>• HTTP</li> <li>• HTTPS</li> </ul>
Process Administrator Port	

### Advanced Properties

col1	col2
Maximum number of connections per server	
Maximum number of connections per external agent	
Handshake Timeout	
Additional Protocol Parameters	

### Log Tab

The **Log** tab allows you to configure properties for the Process Execution Engine logs.

The following engine log properties can be configured from the **Log** tab.



**Note:** It is important to set the logging levels and log files sizes to the levels required for your environment. Unnecessary log messages and large log files can cause decreased overall engine performance.

Property	Description
Messages Logged from Server	Specifies the severity level of log messages generated by the engine.
Messages Logged from BP-Methods	Specifies the severity level of log messages from PBL methods.
Messages Sent by Email	Specifies which log entries are sent via email to the ALBPM administrator.
Maximum Size of Log File	Specifies the maximum size of each engine log file.
Maximum Number of Log Files	Specifies the maximum number of engine log files that are created. When the maximum number of log files is created, the engine overwrites the oldest log file in the sequence.

### Execution Tab

The **Execution** tab allows you to specify the runtime properties of a Process Execution Engine.

### Startup (Standalone)

The **Startup** section allows you to configure any startup properties for the engine.

Property	Description
Start automatically during Process Administrator initialization	Determines if the engine is started automatically when Process Administrator is launched.

Property	Description								
Additional arguments used in startup	<p>Defines any startup arguments passed to the engine on startup. Available engine arguments are:</p> <table> <tr> <th>Option</th><th>Description</th></tr> <tr> <td><b>-c</b></td><td>Rebuilds instance counters displayed in WorkSpace.</td></tr> <tr> <td><b>-p <i>process_id</i></b></td><td>Rebuilds instance counters for the process specified.</td></tr> <tr> <td><b>-t</b></td><td>Specifies the delay, in seconds, the engine waits before shutting down when after the shutdown command is issued.</td></tr> </table>	Option	Description	<b>-c</b>	Rebuilds instance counters displayed in WorkSpace.	<b>-p <i>process_id</i></b>	Rebuilds instance counters for the process specified.	<b>-t</b>	Specifies the delay, in seconds, the engine waits before shutting down when after the shutdown command is issued.
Option	Description								
<b>-c</b>	Rebuilds instance counters displayed in WorkSpace.								
<b>-p <i>process_id</i></b>	Rebuilds instance counters for the process specified.								
<b>-t</b>	Specifies the delay, in seconds, the engine waits before shutting down when after the shutdown command is issued.								
Additional java arguments used in startup	Defines any arguments passed to the JVM during engine startup.								

## Memory

These properties allow you to set memory allocation for the Process Execution Engine and JVM. See [Determining Engine Memory Allocation \(Standalone\)](#) on page 26 for information on setting these properties to improve engine performance.

Property	Description
Maximum JVM Heap Size	<p>Specifies the maximum amount of memory, in megabytes, the engine can use. If the engine's memory usage reaches this limit, the engine stops itself and restarts.</p> <p>You should always try to set the minimum value for the JVM heap size.</p> <p>The default is 256 megabytes</p>
Maximum Instance Size	<p>Specifies the maximum amount of memory a process instance can use. This property limits the size of all process instance variables used by a process instance. When any instance exceeds this limit, the engine is not able to persist instance data and the task fails and must be re-executed after increasing the size of this property.</p> <p>The default is 16 kilobytes</p>
Participant Cache	
Instance Cache Size	<p>Defines the cache size used to store recently accessed process instances. This cache is shared by all processes deployed on the engine.</p> <p>The default is 5000 instances</p> <p>The default value of 5000 instances in the cache is usually considered above average for medium size installations. If the engine has processes with thousands of instances running and these processes have a high concurrency, it is advisable to increase the size of the cache so that the engine does not need to reload instance information when new instances are accessed.</p>

Execution Threads (Standalone)

This section allows you to configure thread handling behavior of ALBPM Standalone. These properties should be configured to reflect the load on your deployed processes.

Property	Description
Maximum number of execution threads used for interactive executions	Determines the size of the thread pool available for user requests. These requests general originate from interactive activities within a process.  The recommended value for this property depends on the size of your process, number of concurrent users, and the number of interactive activities that must be processed.  The default value is 50.
Maximum number of execution threads used for automatic tasks	Determines the size of the thread pool available for automatic tasks.  The default value is 5.
Priority of Automatic Execution Threads	
Automatic Items Queue Size	
Retry Times	
Retry Interval	
Persist instance data	
Maximum Number of Rogue Component Executions	
Maximum amount of acquired instances in a single execution context.	
Request Queue Size	
Request Queue Timeout	

Automatic Extension (J2EE)

Property	Description

JMS (J2EE)

Property	Description

Timeouts

Property	Description
Maximum BP-Methods Timeout	
Interactive Component Timeout	
Maximum Process Web Service Session Timeout	



## Debugger

Property	Description
Trace Components	

## Services Tab

The **Services** tab allows you to configure different Process Execution Engine services.

## Disposer

The Process Execution Engine uses the disposer service to remove completed or aborted process instances and disabled participants from the engine database. If you have configured archiving, the process instance data is transferred to the archive database. If archiving is not configured, the process instance data is removed from the engine database.

Property	Description
Disposer Latency	Specifies, in days, how often the disposer service should run.
Disposer Starting Time	Specifies when the disposer service runs for the first time.
Instance Expiration	Determines how long instance information should remain after reaching the End Activity.
Dispose Disabled Participants	Determines whether information about disabled participants is deleted from the engine database. If enabled, the engine will periodically delete information about disabled participants as specified by the value of the <b>Participant Removal After</b> property.
Archiving	Enables archiving of process instance data. When this option is enabled, you must select the external resource of your archive database. Enabling this option allows you to create the archive database. See <a href="#">Creating the Archive Database</a> on page 62 for more information.

## IPC

The Inter-process Communication (IPC) service allows processes deployed on different Process Execution Engines using different directory services to function as if they were deployed on the same engine. IPC allows these processes to create instances, call subflows, and send notifications using another process.

Property	Description
Enable IPC Service	Enables the IPC Service.
IPC Service Port	Specifies the port the engine uses to handle IPC communication.
Maximum Incoming Connections	Specifies the number of simultaneous external connections an engine can accept. The default value is 5.

## Socket Factory

Option	Description
Enable Socket Factory Service	

## Web Services

Option	Description
Web Services Listener Port	
Max Incoming Connections	
Enable SSL	
Protocol	
Keystore file	
Keystore password	

## JMX

Option	Description
JRMP Adaptor Enabled	
Port	
HTTP Adaptor Enabled	
HTTP Adaptor Port	

## SNMP

The Simple Network Management Protocol (SNMP) is a protocol for performing network management and for monitoring network devices and their functions. SNMP allows you to monitor the status and performance of an engine.

Property	Description
Enable SNMP Service	Enables the SNMP service.
SNMP Agent Port	Specifies the port the SNMP agent uses.
SNMP Manager Host	Specifies the host the SNMP client used to view engine information.
SNMP Manager Port	Specifies the port of the SNMP client used to view engine information.
SNMP Log Level	
SNMP Read Community	
SNMP Write Community	

## Networking Tab

The **Networking** tab allows you to specify general information for connecting to different subsystems and components of ALBPM.

Property	Description
SMTP Mail Server Name	Specifies the mail server the engine uses to send email notifications.
Administrator Email	Specifies the email address of the ALBPM Administrator. Replies to notification emails are sent to this address. Also, log notifications are sent to this address.
WorkSpace URL	Specifies the base URL used to connect to the WorkSpace application.

Property	Description
WorkSpace End URL	Specifies an additional part of the URL used to connect to the WorkSpace application.

## Others Tab


The **Others** tab allows you to specify miscellaneous properties for an engine.

## Directory

Property	Description
Directory Polling Interval	<p>Defines how often, in minutes, the engine metadata is refreshed from the engine database. If you are using a hybrid directory service, this property also defines how frequently LDAP information is refreshed.</p> <p>The default is 1 minute.</p>

## Events

**Store Event** allows you to globally configure how audit events are generated for all processes deployed on the Process Execution Engine.

Property	Description
Depends on process	Generates events only for those processes configured to be archived in Studio. This is the default.
Never	No audit records are generated. Overrides the setting specified within the process.
Always	<p>Generates audit records for all events. Overrides the settings specified within the process.</p> <p> <b>Note:</b> Setting <b>Store Events</b> to this option means that audit records are kept for all activities in all processes. This can lengthen transaction times and decrease performance.</p>

## Round Trip

Property	Description
Enable Roundtrip Simulation	
Roundtrip Simulation Granularity	
Roundtrip Simulation Starting Time	

## PAPI

Property	Description
Instance Retrieval Time	
Do Not Load Instance Task When Execute Filters	
Notify Thread Priority	
Latency Between Notification	

### Application Server Tab (J2EE)

The **Application Server** tab allows you to specify application server properties including including JMS, JMX, and Deployer connectivity information.



**Note:** This tab appears only in the JEE versions of AquaLogic BPM Enterprise.

### Configuration Summary

Property	Description
Engine Runtime Datasource Lookup Name	
Engine DI Datasource Lookup Name	
JMS Queue Name	
JMS Queue Connection Factory	
JMS Queue authentication	
JMS Topic Name for Server synchronization information	
JMS Topic Connection Factory	
JMS Topic authentication	

### JMX Engine Management Configuration

Property	Description
Host	
Port	
Credentials	

### BPM Application Deployer

You can specify the following JMS connectivity information.

Property	Description
BPM Application Deployer URL	
WebLogic Server or Cluster Name	

### Cluster Tab (J2EE)

The **Cluster** tab displays information about the channels used by the engine in cluster communications.



**Note:** This tab only appears in the J2EE versions of ALBPM when clustering is enabled.

### Channel Properties

Property	Description
Distributed Participant Cache Multicast Address	
Distributed Role Cache Multicast Address	
Protocol Stack Configuration	

## Projects

The **Published Projects** page displays the projects that have been published and deployed in all Process Execution Engines.

### Information About Published Projects

Column	Description
Name	
History	
Deployment	
Round Trip Process	

### Project Actions

Project Action	Description
Publish	
Unpublish	
Import Custom Views and Presentations	
AquaLogic Service Bus Registration	

### Publication Source

The **Publication Source** page of the Process Administrator allows you to specify publication and deployment behavior for your project.

### Publication Source

Option	Description
Project at Web Server host	
Exported Project	

### Publication Properties

Property	Description
Smart Publish	
Increase the Project Version Number	
Republish a Previous Version	

### Deployment Properties

Option	Description
Deploy processes after publishing them.	
Import the project's custom views and presentations after the deployment.	

Project Revision Information

Variables

Action	Description
Add	
Delete	

Variable Properties

Property	Description
Name	
Type	Defines the variable type. The possible values are: <ul style="list-style-type: none"><li>• Boolean</li><li>• Integer</li><li>• Real</li><li>• Time</li><li>• Decimal</li><li>• String</li></ul>
Size	
Business Variable	

External Resources

See [External Resource Reference](#) on page 33 for information on each external resource type.

Action	Description
Add	
Delete	

External Resource Properties

Property	Description
Name	
Type	
Subtype	

External Processes

Action	Description
Add	
Delete	

## External Process Properties

Property	Description
Organization	
Organizational Unit	
Process	
URL	
User	
Password	

## Process Monitoring

### Process Data Mart Tab

Property	Description
Enable Automatic Update	
Runtime Database Configuration	
Data Detail Level	
Snapshot Time	
Update Daily Time	
Log Directory	
Messages Logged from Data Store Updater	
Language	
Generate Performance Metrics	
Generate Workload Metrics	
Generate O3 Cubes	

### BAM Tab

Property	Description
Enable Automatic Update	
Updater Database Configuration	
Updater Frequency	Specifies the frequency, in minutes, that BAM data is updated.
Data Expiration Time	

### O3 Tab

O3 is a tool available as part of the ALBPM Enterprise product suite. O3 provides multi-dimensional analysis of process data.

O3 uses Online Analytical Processing (OLAP) protocols to generate business reports on data generated by the Process Execution Engine. When you have configured O3, the Process Execution Engine stores data in a separate database. This database has the same structure as the BAM database.



**Note:** O3 is not included in the basic ALBPM product suite. You must purchase an additional license to use O3.

Property	Description
Enable O3 Cube Generation	
Generate Cubes Daily At	
Host	
Port	


## Admin Center Application Reference

The Admin Center is a java-based application that allows you to start and configure different components of AquaLogic BPM Enterprise.



Action	Description
Help	Displays the HTML version of the <i>ALBPM Enterprise Administration Guide</i> .
About	<p>Displays the <b>About</b> window which contains the following information:</p> <ul style="list-style-type: none"> <li>About: Displays version information for ALBPM Enterprise, including the version number and build number.</li> <li>Virtual Machine Properties: Displays information about the virtual machine running the Process Administrator application.</li> </ul>

### Admin Center Main Window

The main Admin Center window contains allows you to control and configure the ALBPM Enterprise applications. This window also allows you to view console messages for these applications.

Action	Description
Start BPM Applications	<p>Launches the embedded Tomcat application server that runs the ALBPM applications. The Tomcat application server outputs log messages to the <b>Servlet Engine</b> console window.</p> <ul style="list-style-type: none"> <li>On the Standalone version, You can specify which applications are started in <b>Configuration ► BPM Web Applications</b>.</li> <li>On the J2EE version, this option launches Process Administrator. All other ALBPM applications are controlled from the process administrator.</li> </ul>
Stop BPM Applications	Stops the running ALBPM Enterprise applications.
Launch Process Administrator	<p>Starts the Process Administrator application. The default location is: <code>http://localhost:8686/webconsole</code></p> <p> <b>Note:</b> When you start the ALBPM applications using the Admin Center, the Process Administrator is always started by default.</p>
Launch WorkSpace (Standalone)	<p>Starts the WorkSpace application. The default location is: <code>http://localhost:8686/workspace</code></p>



Action	Description
Launch WorkSpace Administrator (Standalone)	Starts the WorkSpace Administrator applications. The default location of this application is: <code>http://localhost:8686/portaladmin</code>
Launch Archive Viewer (Standalone)	
Launch PAPI Web Services Console (Standalone)	
Configuration	Displays the <b>Configuration</b> window which allows you to configure the behavior of the ALBPM Enterprise applications.   <b>Note:</b> When the applications are running, some configuration options are disabled.
Exit	Exits the Admin Center application.   <b>Note:</b> Exiting the Admin Center does not stop any running ALBPM Enterprise applications.

### Admin Center Console Window

The console window is a resizable window that displays runtime messages and startup information for the following ALBPM components:

BPM Component	Console Window Description
Servlet Engine	
BPM Web Applications	
Process Administrator	
WorkSpace	
Feeds Service	
WorkSpace Administrator	
PAPI Web Services	

## Configuration

The **Configuration** window allows you to configure the behavior of different components of ALBPM Enterprise.

### Directory Tab

The **Directory** tab allows you to create, configure, and edit directory services.

This tab also displays a list of the currently configured directory services. You can perform the following actions related to directory services:

Action	Description
Add	Create a new directory service configuration. Clicking <b>Add</b> launches the ALBPM Configuration Wizard. See <a href="#">ALBPM Configuration Wizard Reference</a> on page 100.
Remove	Delete the selected directory service configuration.
Edit	Edit the configuration properties for the selected directory service configuration.


Action	Description
Set as Current	Sets the selected directory service configuration as the default.

### Edit Directory Configuration

The **Edit Directory Configuration** window allows you to configure connectivity properties of the directory database.

### BPM Web Applications Tab

The **BPM Web Applications** tab allows you to configure general preferences for the ALBPM web applications. This tab also allows you to install the ALBPM web applications as a Windows Service.

Preference	Description
JDK Home Folder	Specifies the location of the JDK the embedded Tomcat application server and ALBPM applications use. The default is the location of the JDK installed with ALBPM Enterprise. However, you can specify the path to another JDK installation.
Port	Specifies the port for the ALBPM applications. Although the URL for each application is different, each application uses the same port number. The default port is: 8686.
Shutdown Port	Specifies the port the Tomcat embedded server uses to listen to start and stop requests. The default is: 8886.   <b>Note:</b> This port can be a potential security risk. You should ensure that it is properly locked down to avoid inadvertent shutdown of the application server.
BPM Application Startup	Allows you to select the ALBPM applications that are started when selecting <b>Start BPM Web Applications</b> from the Admin Center main window.
Request confirmation when exiting.	Causes the Admin Center to require confirmation on exit if the BPM Web Applications are running.
Install as Service	Allows you to install the BPM Applications as a Windows Service.

### Process Administrator Tab

The **Process Administrator** tab allows you to configure the Process Administrator application log and enable Single Sign On (SSO).

Property	Description
Log Folder	Specifies the location of the Process Administrator application log file.
Log Message Severity Level	Defined the severity level of the messages written to the application log.
Enable Single Sign On (SSO)	Allows access to SSO to the Process Administrator. When enabled, you must specify the name of the Java class used to implement SSO. This class utilizes the applications server's security configuration. The default class is: <code>fuego.web.SSOUserLogin</code> .

### WorkSpace Tab

The **WorkSpace** tab allows you to configure the WorkSpace application log, enable Single Sign On (SSO), and generate the application WAR file.

Property	Description
Log Folder	Specifies the location of the WorkSpace application log file.
Log Message Severity Level	Defined the severity level of the messages written to the application log.
Enable Single Sign On (SSO)	Allows access to SSO to the Process Administrator. When enabled, you must specify the name of the Java class used to implement SSO. This class utilizes the applications server's security configuration. The default class is: <code>fuego.workspace.security.SSOWorkSpaceLogin</code> .
Generate Web Archive (WAR) File	Generates the WAR file for the WorkSpace application and allows you to save it to the file system.

### WorkSpace Admin Tab

The **WorkSpace Administrator** tab allows you to configure the WorkSpace Administrator application log, enable Single Sign On (SSO), and generate the application WAR file.

Property	Description
Log Folder	Specifies the location of the WorkSpace Administrator application log file.
Log Message Severity Level	Defined the severity level of the messages written to the application log.
Enable Single Sign On (SSO)	Allows access to SSO to the Process Administrator. When enabled, you must specify the name of the Java class used to implement SSO. This class utilizes the applications server's security configuration.  The default class is: <code>fuego.web.SSOUserLogin</code> .
Generate Web Archive (WAR) File	Generates the WAR file for the WorkSpace Administrator application and allows you to save it to the file system.

### PAPI Web Services Tab

The **PAPI Web Services** tab allows you to configure the PAPI Web Services log, enable Single Sign On (SSO), and generate the application WAR file.

Property	Description
Log Folder	Specifies the location of the PAPI Web Services log file.
Log Message Severity Level	Defined the severity level of the messages written to the application log.
Enable Single Sign On	Allows PAPI Web Services to use SSO. You must create a class that implements the <code>fuego.sso.SSOUserLoginInterface</code> . See the <i>ALBPM Process API Developer's Guide</i> for more information.
Enable HTTP Basic Authentication	Enables HTTP authentication which forces the browser request to provide a username and password for authentication.
Enable Username Token Profile Authentication	Causes PAPI WS to require a username and password for authentication. This is the default security mechanism used by PAPI WS.
Generate Web Archive (WAR) File	Generates the WAR file for PAPI Web Services and allows you to save it to the file system.

### Service Pack Updates Tab

The **Service Pack Updates** tab allows you to install ALBPM Enterprise service packs and roll back previously installed service packs and hotfixes.

Property	Description
Local Update	
Rollback Service Pack	
Rollback Hotfix	

## Archive Viewer Application Reference

The archive viewer is a web based application that allows you view and search archive process instance information.

### Login Screen

### Archive Viewer

## 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:

- 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 101.

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

See [Directory Provider Type](#) on page 102.

3. Enter information about your directory provider type.

See [Directory Provider Selection](#) on page 102. If you have chosen to implement the

4. Enter connectivity information about your database.

Provider	More information
Oracle	<a href="#">Configure Directory Provider - Oracle</a> on page 103
DB2	<a href="#">Configure Directory Provider - DB2</a> on page 104
SQL Server	<a href="#">Configure Directory Provider - SQL Server</a> on page 105

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.  <b>Note:</b> If you choose to use an existing directory service, it must be created using the current version of AquaLogic BPM.
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.  <b>Note:</b> 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.  <b>Note:</b> When configuring WorkSpace Extensions, you must select the ALI Identity Service (Hydrogen).

Option	Description
BPM Administrator User	<p>Defines the BPM Administrator user ID.</p> <p> <b>Note:</b> The Configuration Wizard creates this new user ID in the database.</p> <p> <b>Note:</b> When configuring WorkSpace Extensions, this administrator ID must also be defined in the AquaLogic Integration Portal.</p>
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	<p>Optionally, specifies the name of the schema used.</p> <p> <b>Note:</b> When using a schema, it is recommended that the schema name and user name be the same.</p>
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.

Option	Description
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.
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.
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.  <b>Note:</b> 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	
Temporary Tablespace	
Profile	

### 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.   <b>Note:</b> 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.

# ALBPM MBean Reference

## Active Process MBean

### Attributes

Name	Type	Description
Id	java.lang.String	The process Id.
In	int	The native identifier of the process.
CatalogIn	int	The IN of the associated catalog for this process.
Organization	java.lang.String	The name of the organization where this process is deployed.
OrganizationalUnit	java.lang.String	The name of the organization unit where this process is deployed.
IsDeprecated	boolean	Indicates if the process is deprecated

Name	Type	Description
IsArchiving	boolean	Indicates if archiving is enabled for the engine running this process.
CommittedTransactions	int	Total number transactions committed by this process.
RollbackedTransactions	int	Total number of transactions rolled-back by this process.
RunningTransactions	int	Number transactions currently running in this process.
EventsCount	int	Total number of events generated by this process.
SentMailsCount	int	Total number of emails sent by this process.
SentNewsCount	int	Total number of news notifications sent by this process.
LastNewsTimeStamp	java.lang.String	The Time when the last news was sent. Returns an ISO standard string representation of a time value.
PendingNewsCount	int	Number of pending news items to be sent.

### Operations

Operation	Description
resetCounters(int CommittedTransactions, int RolledbackTransactions, int RunningTransactions, int SentMailsCount, int SentNewsCount, int EventsCount)	Resets the transactions counters for this process.

## Active Processes MBean

### Attributes

Name	Type	Description
Count	int	Number of active processes running on the Engine.

## Connection Pool MBean (Standalone)



**Note:** This MBean is only available on ALBPM Enterprise Standalone.

### Attributes

Name	Type	Description
ConfigurationName	java.lang.String	The name of the configuration.
Size	int	The size of this configuration.
IdleTime	int	TimeToLive used for a connection inside this configuration.



Name	Type	Description
LockedConnections	int	Number of currently locked connections for this configuration.
NotLockedConnections	int	Number of currently not locked connections for this configuration.
TotalLocked	int	Total number of locked connections for this configuration.
TotalUnLocked	int	Total number of unlocked connections for this configuration.
EmptySlots	int	Number of empty slots in this configuration.
MinNonDisposable	int	Minimum number of non disposable entries for this configuration.
MissCount	int	Number of misses for this configuration. A miss is when someone needed a connection and there was none available.

## Connection Pools MBean (Standalone)



**Note:** This MBean is only available on ALBPM Enterprise Standalone.

### Attributes

Name	Type	Description
Count	int	Number of configurations in the Engine.

## Engine MBean

### Attributes

Name	Type	Description
Name	java.lang.String	The name of the Engine
Host	java.lang.String	The hostname where the Engine is running. This can be either the DNS hostname or the IP address of the host.
Status	java.lang.String	The current status of the Engine. See <a href="#">Engine Status</a> on page 23 for a description of possible engine states.
RunLevel (editable)	int	The current runLevel of the Engine. The Engine RunLevel can be changed through this attribute.
TotalMemory	long	The total memory available to the engine, specified in megabytes.
UsedMemory	long	The total memory used by the engine, specified in megabytes.

Name	Type	Description
FreeMemory	long	The amount of free memory available to the engine, specified in megabytes.
CommittedTransactions	int	The total number of committed database transactions.
RollbackedTransactions	int	The total number of transactions that have been rolled-back.
RunningTransactions	int	The number of database transactions currently running for this engine.
EngineExecutedStatements	java.util.Map	A map containing a key/value pair of all engine executed statements (key) and the corresponding number of executions (value).
FDIExecutedStatements	java.util.Map	A map containing a key/value pair of all FDI executed statements (key) and the corresponding number of executions (value).
EngineMostExecutedStatements	java.util.List	A sorted list containing the most Engine executed statements at the beginning.
FDIMostExecutedStatements	java.util.List	A sorted list containing the most FDI executed statements at the beginning.

## Execution Pool (Standalone)




**Note:** This MBean is only available on ALBPM Enterprise Standalone.

### Attributes

Name	Type	Description
Size	int	The size of the thread pool.
Busy	int	Number of busy threads in this pool
Free	int	Number of free threads in this pool
Unused	int	Number of unused threads.
AutomaticPoolSize	int	Size of the threads pool for automatic executions.
FblPoolSize	int	Size of the threads pool for FBL executions.
HttpPoolSize	int	Size of the threads pool for HTTP executions.
InteractivePoolSize	int	Size of the threads pool for Interactive executions.
AutomaticCount	int	Number of current automatic execution.
FblCount	int	Number of current PBL methods being run
InteractiveCount	int	Number of current interactive execution
HttpCount	int	Number of current HTTP execution

Name	Type	Description
MaxRequestLatency	long	Max time in millis a request can be executed
RequestQueueSize	int	Size of the request queue.
RogueCounter	int	Number of rogue threads in the pool
RogueLimit	int	Max number of rogue threads allowed
TimeStamp	java.lang.String	Time when this pool was created.

## FDI Connection Pool MBean (Standalone)

 **Note:** This MBean is only available on ALBPM Enterprise Standalone.

### Attributes

Name	Type	Description
Capacity	int	Maximum number of FDI connections.
Empty	int	Number of empty slots.
EntryCapacity	int	Maximum number of entries
Locked	int	Number of FDI locked connections.
NotLocked	int	Number of not locked FDI connections
TotalLocked	int	Total number of locked FDI connections
TotaUnlLocked	int	Total number of unlocked FDI connections
MinNoDisposable	int	The Minimum number of disposable entries.
MissCount	int	Number of misses for this pool
TTL	int	TimeToLive used for FDI connections

## Server Session MBean

### Attributes

Name	Type	Description
Participant	java.lang.String	The Participant name.
Id	int	The ServerSession IN
CreationTime	java.lang.String	TimeStamp when the ServerSession was created.
Host	java.lang.String	The name of the host where the client is connected from
LastOperationTime	java.lang.String	TimeStamp of the last operation for this ServerSession.

## Server Sessions MBean

### Attributes

Name	Type	Description
Count	int	Number of client sessions active in this engine.

## 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	

Property	Description
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.