



Configuring Environment Components for the Sun Business Process Manager

Sun Microsystems, Inc.
4150 Network Circle
Santa Clara, CA 95054
U.S.A.



Part No: 820-3495
April 2008

Copyright 2008 Sun Microsystems, Inc. 4150 Network Circle, Santa Clara, CA 95054 U.S.A. All rights reserved.

Sun Microsystems, Inc. has intellectual property rights relating to technology embodied in the product that is described in this document. In particular, and without limitation, these intellectual property rights may include one or more U.S. patents or pending patent applications in the U.S. and in other countries.

U.S. Government Rights – Commercial software. Government users are subject to the Sun Microsystems, Inc. standard license agreement and applicable provisions of the FAR and its supplements.

This distribution may include materials developed by third parties.

Parts of the product may be derived from Berkeley BSD systems, licensed from the University of California. UNIX is a registered trademark in the U.S. and other countries, exclusively licensed through X/Open Company, Ltd.

Sun, Sun Microsystems, the Sun logo, the Solaris logo, the Java Coffee Cup logo, docs.sun.com, Java, and Solaris are trademarks or registered trademarks of Sun Microsystems, Inc. in the U.S. and other countries. All SPARC trademarks are used under license and are trademarks or registered trademarks of SPARC International, Inc. in the U.S. and other countries. Products bearing SPARC trademarks are based upon an architecture developed by Sun Microsystems, Inc.

The OPEN LOOK and SunTM Graphical User Interface was developed by Sun Microsystems, Inc. for its users and licensees. Sun acknowledges the pioneering efforts of Xerox in researching and developing the concept of visual or graphical user interfaces for the computer industry. Sun holds a non-exclusive license from Xerox to the Xerox Graphical User Interface, which license also covers Sun's licensees who implement OPEN LOOK GUIs and otherwise comply with Sun's written license agreements.

Products covered by and information contained in this publication are controlled by U.S. Export Control laws and may be subject to the export or import laws in other countries. Nuclear, missile, chemical or biological weapons or nuclear maritime end uses or end users, whether direct or indirect, are strictly prohibited. Export or reexport to countries subject to U.S. embargo or to entities identified on U.S. export exclusion lists, including, but not limited to, the denied persons and specially designated nationals lists is strictly prohibited.

DOCUMENTATION IS PROVIDED "AS IS" AND ALL EXPRESS OR IMPLIED CONDITIONS, REPRESENTATIONS AND WARRANTIES, INCLUDING ANY IMPLIED WARRANTY OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE OR NON-INFRINGEMENT, ARE DISCLAIMED, EXCEPT TO THE EXTENT THAT SUCH DISCLAIMERS ARE HELD TO BE LEGALLY INVALID.

Contents

Configuring Environment Components for the Sun Business Process Manager	5
Monitoring and Recovery Database Overview	5
Database Scripts	5
BPM Engine and Business Process Configuration	6
Creating Environments	6
Creating an Environment for a Basic Business Process	6
Creating an Environment for a User Activity Business Process	7
Creating Environments for Web Service Business Processes	8
Configuring the BPM Engine	8
Configuring Load Balancing	11
Configuring Failover	12
Configuring the BPM Engine for Monitoring and Recovery	12
▼ To configure the BPM Engine for monitoring and recovery	12
Configuring the Worklist Manager External System	13
Configuring the WorkList Manager	13

Configuring Environment Components for the Sun Business Process Manager

The topics listed here provide links to task, conceptual and reference information for configuring environment components for the Sun Business Process Manager (BPM). If you have any questions or problems, see the Java CAPS web site at <http://goldstar.stc.com/support>.

BPM provides graphical editors to help you design and configure the models that depict your Business Processes (BPs) using simple drag and drop procedures. These topics provide the background information and instructions you need to configure BP environment components.

- “Monitoring and Recovery Database Overview” on page 5
- “Creating Environments” on page 6
- “Configuring the BPM Engine for Monitoring and Recovery” on page 12
- “Configuring the Worklist Manager External System” on page 13

Monitoring and Recovery Database Overview

When configured for persistence, Sun Business Process Manager (BPM) stores information about Business Process instances and activities for recovery and monitoring. Once the data is persisted, you can use Enterprise Manager to monitor Business Process instances. The level of detail you can view depends on whether you have configured a Business Process for reporting persistence and have configured the tables specific to that Business Process.

Database Scripts

The database scripts for the monitoring and recovery database are accessed from the Java CAPS Repository, and include scripts to create, drop, purge, and archive database tables. They are located in the Sun BPM node. There are two sets of scripts you can use to create the database tables. One set of database scripts is located under Run Database Scripts, which only contains the scripts to create or drop the database tables. These scripts can be modified and run from NetBeans.

The second set of scripts is located under Download Database Scripts and are contained in .zip files specific to each database vendor. You can download these scripts, and then modify and run them from a local directory. These files contain the scripts to create or drop database tables, as well as additional purge and archive scripts that you can use to manage aging information stored in your database.

Database scripts are also generated in the Project Explorer under the Business Process node for each Business Process configured for reporting persistence. Configuring a Business Process for persistence allows you to collect and monitor more detailed data for that process than would be collected in the standard database.

BPM Engine and Business Process Configuration

The BPM Engine must be configured for persistence in order to access and store information in the monitoring and recovery database. When you configure the BPM Engine for persistence, data is stored for all Business Process instances running on the associated application server. You configure specific Business Processes to persist more detailed information for reporting and monitoring in the properties for each Business Process. When you configure a Business Process for persistence, additional database scripts are automatically created under that process. These scripts need to be run in order to enable the more detailed level of persistence.

Creating Environments

The Environment represents the physical configuration of the Business Process Project. This section outlines how to create Environments for basic Business Process Projects and for Business Processes that include a user activity. In a production environment, the Java CAPS Environment is like to be more complex than the examples shown here. Once you create an Environment for BPM, you must configure the BPM Engine properties, located on the Properties window of the application server.

Creating an Environment for a Basic Business Process

The components in a Project that contains a basic Business Process will vary depending on the external systems and other components used in the Project.

▼ To create an Environment for a basic Business Process

- 1 In NetBeans, click the Environment Explorer tab.**
- 2 Right-click the Repository name and then click New Environment.**
- 3 Name the Environment.**

- 4 Right-click the Environment, point to New, and then click Logical Host.
- 5 Name the Logical Host.
- 6 Right-click the Logical Host, point to New, and then click the type of application server on which you are deploying.
- 7 Name and configure the application server.
- 8 Add and configure any necessary external systems to the Environment.
- 9 Configure the BPM Engine properties described in [“Configuring the BPM Engine” on page 8](#).
- 10 Click Save All.

Creating an Environment for a User Activity Business Process

The components in a Project that contains a basic Business Process will vary depending on the external systems and other components used in the Project.

▼ To create an Environment for a user activity Business Process

- 1 In the Environment Explorer, expand Environment1.
- 2 Right-click the Repository name and then click New Environment.
- 3 Name the Environment.
- 4 Right-click the Environment, point to New, and then click Logical Host.
- 5 Name the Logical Host.
- 6 Right-click the Logical Host, point to New, and then click the type of application server on which you are deploying.
- 7 Name and configure the application server.
- 8 Add and configure any necessary external systems to the Environment.

Note – To access the Worklist Manager for task completion, you need to add and configure a Worklist Manager External System. If the user activity includes eVision web pages, you need to add an eVision External System as well.

- 9 Configure the Worklist Manager.
- 10 Configure the BPM Engine properties described in [“Configuring the BPM Engine” on page 8](#).
- 11 Click Save All.

Creating Environments for Web Service Business Processes

In BPM you can expose a Business Process as a web service and you can invoke web services from a Business Process. The Environment for each type of web service Business Process require different components.

Configuring the BPM Engine

The Properties window for the BPM Engine configures several aspects of the BPM Engine, including debugging, database connection, load balancing, failover and recovery, and so on. [Table 1](#) lists and describes each property.

TABLE 1 BPM Engine Properties

Property	Description
Debug	An indicator of whether the Business Process Debugger is enabled. This is not recommended for production environments because it impacts performance.
Debug Port	The port on which the Business Process Debugger starts.
Application Mode	An indicator of whether you are balancing processes across multiple BPM Engines. Select from the following options: <ul style="list-style-type: none"> ▪ <i>Single Engine</i>: All processes are handled by one BPM Engine. ▪ <i>Multiple Engine</i>: Processes are distributed across multiple engines. Persistence must also be enabled; for the Persistence Mode property, select Persist to Database - Multiple Engines.

TABLE 1 BPM Engine Properties (Continued)

Property	Description
Receive Timeout (milliseconds)	The number of milliseconds to wait to process a message that has been placed in waiting. Messages are placed in waiting for various reasons, such as when the maximum number of concurrent instances is reached.
Persistence Mode	An indicator of whether instance data is persisted to the monitoring and recovery database. <ul style="list-style-type: none"> ■ False: Data is not saved to the monitoring and recovery database. ■ <i>Persist to Database - Single Engine:</i> Data is saved to the database and processing is handled by one engine. ■ <i>Persist to Database - Multiple Engines:</i> Data is saved to the database and processing is distributed across multiple engines. If you select this, you must set the Application Mode property to Multiple Engine.
Recovery Enabled	An indicator of whether data can be recovered to a previous state in case of failure. Persistence must be enabled in the Persistence Mode and Application Mode properties for recovery to be enabled.
Engine Expiry Interval (seconds)	The number of seconds for the BPM Engine to wait to register itself as alive. For more information, see “Configuring Failover” on page 12 .
Failover Grace Period (seconds)	The elapsed time period before moving running Business Process instances from an unavailable engine to an available engine. This is used in conjunction with the Engine Expiry Interval property for configuring failover.
Recovery Batch Size	The number of records to recover at one time. Sun does not recommend setting this higher than 100.
Database	The type and version of database you are using for monitoring and recovery. If you are using an Oracle 10g database, select Oracle 9i .
Database Host	The name of the machine on which the database resides.
Database Port	The port number on which the database is listening.
Oracle Net Service Name	The name for the connect descriptor (for Oracle databases only). This is the TNS name of the database, and is required to access the database using the OCI driver to access the database. If you are not using the OCI driver, leave this property blank.
Database Instance/Schema	The name of the database. For Oracle, this is the SID name.
Database User	The login ID for the monitoring and recovery database owner. The user name is defined in the database scripts you ran when creating the database tables (by default, bpm6user).

TABLE 1 BPM Engine Properties (Continued)

Property	Description
Database Password	The password for the monitoring and recovery database owner. The password is defined in the database scripts you ran when creating the database tables (by default, bpm6user).
Database Connection Pool Size	The maximum number of physical connections the pool should keep available at all times. 0 (zero) indicates that there is no maximum. The pool size depends on the transaction volume and response time. If the pool size is too big, you may end up with too many connections with the database. Sun recommends setting this no higher than 60.
Database Connection Retries	The number of retries to establish a connection with the database.
Database Connection Retry Interval (milliseconds)	The number of milliseconds to wait between each attempt to access the database. This property is used in conjunction with Database Connection Retries .
Database Connection Steady Pool Size	The initial and minimum number of physical connections the pool should keep available at all times. 0 (zero) indicates that there should be no physical connections in the pool and the new connections should be created as needed. If the pool size is too small, you might experience longer connection times due to the existing number of physical connections.
Database Connection Max Idle Time (Seconds)	The maximum number of seconds that a physical connection will remain unused before it is closed. 0 (zero) indicates that there is no limit.
Enable Monitoring	Enables monitoring of Business Processes through the Enterprise Manager Monitor. If monitoring is enabled, persistence must also be enabled in the Application Mode and Persistence Mode properties.
Reporting Thread Sleep Time (milliseconds)	The time in milliseconds between transfers of data from the monitoring and recovery database tables to the Business Process reporting tables.
Monitoring Thread Buffer Size	The number of records at which the buffer contents is transferred to the database (if the thread buffer time lag is not expired). Monitoring data is collected in a memory buffer and is transferred to the monitoring tables based on either the buffer size or the buffer time lag, whichever occurs first.
Monitoring Thread Buffer Time Lag (seconds)	The time in seconds between transfers of data from the buffer to the monitoring table (if the buffer has not reached the thread buffer size).
Monitoring Thread Sleep Time (milliseconds)	The time in milliseconds between transfers of data from the buffer to activity monitoring table.

TABLE 1 BPM Engine Properties (Continued)

Property	Description
Work Item Submit Limit	The maximum number of work items the BPM Engine can submit to the integration server at a given time for execution. A work item is an activity or group of activities in a Business Process submitted as a single unit of work to be run on an integration server thread.
Invocation Allocation Ratio (%)	Specifies the percentage of the total Work Item Submit Limit threads that can be used to invoke activities, as opposed to other types of activities. Setting this ratio to 100% can cause a deadlock.
Automatic Execution of Database Scripts	Specifies whether database scripts will be run automatically.

Configuring Load Balancing

When a Business Process needs to be scaled to meet heavier processing needs, you can distribute the Business Process across multiple engines to increase throughput. BPM's load balancing algorithm automatically distributes processing across multiple engines; however, BPM cannot load balance correlated messages.

Note – The File Adapter is not designed to work in an BPM load-balancing scenario. Using a File Adapter will result in all instances being sent to one engine rather than being distributed.

▼ To configure load balancing

- 1 For each affected Business Process, enable persistence.
- 2 In the Environment Explorer, right-click the application server and then click Properties.
- 3 In the BPM Engine Configuration properties, do the following:
 - a. Set the Persistence Mode property to **Persist to Database - Multiple Engines**.
 - b. Set the Application Mode property to **Multiple Engine**.
 - c. Click OK.
- 4 Configure all BPM Engines to share the same database.

Configuring Failover

When the Business Process is configured for load balancing, BPM's failover capabilities ensure throughput of running Business Process instances. When Business Process instances encounter an engine failure, BPM load balances those instances across all available engines. As with load balancing, BPM's failover capabilities are limited to non-correlated messages.

▼ To configure failover

- 1 In the Environment Explorer, right-click the application server and then click Properties.
- 2 In the BPM Engine Configuration properties, set the Engine Expiry Interval property to register itself as alive frequently enough to meet the demands of your system.
- 3 In the BPM Engine Configuration properties, set the Failover Grace Period property to the optimal elapsed time period before moving running Business Process instances from an unavailable engine to an available engine.

Note – Optimizing these two property setting might require some testing. The Engine Expiry Interval property also applies to the interval for the recovery of dangling instances.

Configuring the BPM Engine for Monitoring and Recovery

A basic Java CAPS deployment with BPM Business Processes leverages the runtime monitoring and management features of Enterprise Manager Monitor. In order to monitor and manage your runtime deployments, you must configure the BPM Engine to enable persistence and to connect to the monitoring and recovery database. For more information about the BPM Engine properties, see [“Configuring the BPM Engine” on page 8](#).

▼ To configure the BPM Engine for monitoring and recovery

- 1 In the Environment Explorer, expand the Environment in which the Business Process will run, and then expand the Logical Host.
- 2 Right-click the application or integration server, and then click Properties.
- 3 Click BPM Engine Configuration.
The properties for the BPM Engine appear.

- 4 To configure the connection to the monitoring and recovery database, enter the following:
 - **Database** - The type and version of database you are using.
 - **Database Host** - The name of the machine on which the database resides.
 - **Database Port** - The port number to connect to the database.
 - **Database Instance/Schema** - The database instance, schema, or SID.
 - **Database User** - The user name for the database. This is the user name defined in the database scripts you ran when creating the database tables.
 - **Database Password** - The password for the database user. This is the password defined in the database scripts you ran when creating the database tables.
- 5 To enable persistence and recovery *without* load balancing and failover, enter the following:
 - **Persistence Mode** - Select Persist to Database - Single Engine (Recovery).
 - **Recovery Enabled** - Select true.
- 6 To enable monitoring with Enterprise Manager, set **Enable Monitoring** to true.
- 7 To enable the automatic execution of the database scripts, set **Automatic Execution of Database Scripts** to true.

Note – Until you are ready to optimize the BPM Engine for performance, scalability, and reliability, do not change the default settings for any other BPM Engine configuration properties.

- 8 Click OK.

Configuring the Worklist Manager External System

The Worklist Manager is a web-based interface that allows you to view, assign, escalate, and execute the tasks generated from user activities. The functions that can be performed in the Worklist Manager are based on user hierarchy. BPM supports the definition of organization hierarchies and user roles for task assignment. Tasks can be escalated and delegated by users from custom worklists and activity processing windows. The Worklist Manager requires an LDAP directory to define users, their roles, and their hierarchy.

Configuring the WorkList Manager

When you create the Environment for a user activity Business Process, you need to create and configure a Worklist Manager External System. The properties you configure for the external

system define the Worklist Manager database connectivity, LDAP server and directory information, and custom labels for flex attributes. Perform the following steps to configure the Worklist Manager.

- “Creating the Worklist Manager External System” on page 14
- “Defining Configuration Properties” on page 14
- “Customizing Flex Attribute Labels” on page 16
- “Defining Email Notification Properties” on page 16

In addition, you must perform one of the following tasks:

- “Configuring an OpenLDAP Connection” on page 17
- “Configuring a Sun Java System Directory Server Connection” on page 19
- “Configuring a Microsoft Active Directory Connection” on page 22

Creating the Worklist Manager External System

The Worklist Manager External System is created from the Environment Explorer.

▼ To create the Worklist Manager External System

- 1 In the Environment Explorer, right-click the Environment for the user activity Project.
- 2 Point to New, and then click Worklist Manager.
- 3 Enter a name for the Worklist Manager External System, and then click OK.
The Properties window appears.
- 4 Configure the Worklist Manager, as described in the following sections.

Defining Configuration Properties

The configuration properties of the Worklist Manager define information about the Worklist Manager database and application.

▼ To define configuration properties

- 1 On the Worklist Manager Properties window, click WLMConnector External System Configuration.
- 2 Enter values for the properties described in the following table.
- 3 Click OK.

Property	Description
Database Type	The type of database you are using. Select from oracle , sybase , sqlserver , or db2 .
Database Host	The name of the database server.
Database Port	The port number on which the database is listening.
Database SID	The SID name of the database.
Database User ID	The login ID of the WLM user for the database. This should be the same user as was created by the WLM database installation scripts.
Database Password	The password for the WLM user.
Database Connection Retry Count	The number of times the driver will try to connect to the database after an unsuccessful attempt.
Database Connection Retry Delay	The number of seconds the driver waits before retrying to connect to the database server.
Database Initial Pool Size	The number of physical connections with which the database will be initialized.
Database Pool Minimum Pool Size	The minimum number of physical connections in the database pool.
Database Pool Maximum Pool Size	The maximum number of physical connections in the database pool.
Database Pool Property Cycle	The frequency in seconds that the database pool manager should check the pool.
Database Pool Idle Time	The length of time a physical connection can be inactive.
Auto Commit	An indicator of whether to enable the auto-commit feature of the database.
Connection Parameters	The type of LDAP directory you are using. Select from <i>Sun Java System Directory Server</i> , ActiveDirectoryConnection , or OpenLdapConnection .

Property	Description
WLM Application Name	The name of the Worklist Manager application that is generated when you deploy the Project. This name is part of the URL used to log into the Worklist Manager and should be unique for each user activity Business Process. You can deploy multiple Business Processes with user activities to the same Environment, but each WLM application must have a unique name. In that case, the Environment must include multiple Worklist Manager External Systems.
WLM Session Timeout (minutes)	The number of minutes the Worklist Manager can remain idle before timing out.
WLM Initial Order	The order of the fields on the Worklist Manager. When you click the ellipses next to this property, the Ordering Sequence dialog box appears, where you can select the fields to display on the Worklist Manager, and the order in which they appear.

Customizing Flex Attribute Labels

Flex attributes are customizable attributes that aid in task assignment. The attributes appear in the Business Rule Designer as well as in columns of the Worklist Manager.

You can map values to these attributes in the Business Rule Designer so the values appear in the Worklist Manager. You can also label the attributes to make them easy to identify in the Worklist Manager.

▼ To customize flex attribute labels

- 1 From the Environment Explorer, right-click the Worklist Manager External System.
- 2 Click Properties.
The Properties dialog box appears.
- 3 Click Custom Attribute Labels.
- 4 Define labels for as many attributes as necessary.
- 5 Click OK.

Defining Email Notification Properties

If you defined email notifications on the Worklist Manager window for a user activity, you need to define the connection properties for the email server in the Worklist Manager External

System properties. You also need to modify the LDAP properties for the directory server you are using by specifying the name of the attribute that contains the users' email addresses.

▼ To define email notification properties

- 1 On the Worklist Manager Properties window, click **Email Server Connection Parameters**.
- 2 Enter values for the properties.
- 3 Click **OK**.

Property	Description
Email Outgoing server	The name of the email server on which the Worklist Manager email notifications are sent.
Email User Name	The login ID for the email account used by the Worklist Manager.
Email Password	The password for the email account.
Sender of Email	The name that should appear in the email as the sender. This property is used to create a URL, so it cannot contain any spaces.
First footer of Email	A footer for the email notifications.
Second footer/Disclaimer of Email	A second footer or disclaimer for the email notifications.

Configuring an OpenLDAP Connection

To use OpenLDAP with the Worklist Manager, you must specify certain information about the LDAP directory structure so the Worklist Manager knows where to find the user information defined in the directory. You can use your existing directory structure as long as there is a mechanism for defining a user reporting hierarchy.

The Worklist Manager uses an anonymous bind with OpenLDAP, so you do not need to specify credentials for the security principal.

▼ To configure an OpenLDAP connection

- 1 From the Environment Explorer tab, right-click the Worklist Manager External System, and then click **Properties**.
- 2 On the properties page that appears, verify that the **Connection Parameters** property is set to **OpenLdapConnection**.

3 Expand WLMConnector External System Configuration, and then click Open Ldap Parameters.

The Properties dialog box appears.

4 Enter values for the properties.

Note – The default values for these properties are based on the values for the user activity sample and the and audit processing tutorial. Modify these values to suit your existing directory structure and attributes.

5 Click OK.

Property	Description
LDAP Initial Context Factory	The fully qualified name of the factory class that creates the initial context. The initial context is the starting point for JNDI naming operations.
LDAP Provider URL	The URL of the LDAP server. The format of the URL is ldap://<host>:<port> , where <host> is the name of the computer hosting the LDAP server, and <port> is the port number on which the LDAP server is listening for requests.
Open LDAP RootName	The name of the root node in the LDAP directory. For example, if the root node is dc=Sun,dc=com , the value for this property would be Sun .
Open LDAP Version	The version of OpenLDAP you are running.
Open LDAP Search Filter	A search filter used by the Worklist Manager to search for users. The Worklist Manager will only find those users described by the filter. For example, to use an organization name as the search filter where all Worklist Managers are assigned to the Sun organization, the value for this property would be (o=Sun) .
Open LDAP Search DN	The DN of the root entry of the portion of the LDAP directory where the Worklist Manager will start the search for users. For example, if users are all defined in an organizational unit named People , the value for this property would be similar to ou=People,dc=sun,dc=com .

Property	Description
Open LDAP Attribute Role	The name of the role attribute in the LDAP directory to which Worklist Manager users belong. Only enter a value for this property if you have defined a role to which all Worklist Manager users are assigned.
Open LDAP Attribute Manager	The name of the attribute that specifies reporting hierarchy in the LDAP directory. This is the attribute assigned to a user that specifies who they report to. You can use <i>Manager</i> , the predefined attribute for OpenLDAP, or you can create a custom attribute. The Worklist Manager uses this entry to create the list of users to whom a supervisor can reassign tasks, and to specify the supervisor when a task is escalated.
Open LDAP Group	The name of the group attribute in the LDAP directory to which Worklist Manager users belong. Only enter a value for this property if you have defined a group to which all Worklist Manager users are assigned.
Open LDAP Email	The name of the attribute that contains a user's email addresses. This is used in email notification.
Open LDAP Attribute Given Name	The name of the attribute that contains a user's first name. This is used in email notification.

Configuring a Sun Java System Directory Server Connection

To use the Sun Java System Directory Server with the Worklist Manager, you must specify certain information about the LDAP directory structure so the Worklist Manager knows where to find the user information defined in the directory. You can use your existing directory structure as long as there is a mechanism for defining a user reporting hierarchy.

▼ To configure a Sun Java System Directory Server connection

- 1 From the Environment Explorer tab, right-click the Worklist Manager External System, and then click Properties.
- 2 On the properties page that appears, verify that the Connection Parameters property is set to Sun Java System Directory Server.
- 3 Expand WLMConnector External System Configuration, and then click Sun Java System Directory Server/ADS.
The Properties dialog box appears.
- 4 Enter the property values for the properties described in the following table.

Note – Depending on how your LDAP directory is set up, not all of these fields are required. The default configuration is not necessarily illustrative of an actual implementation.

5 Click OK to close the Properties dialog box.

Property	Description
Java Naming Provider URL	The URL of the LDAP server. The format of the URL is ldap://<host>:<port> , where <host> is the name of the computer hosting the LDAP server, and <port> is the port number on which the LDAP server is listening for requests.
Java Naming Factory Initial	The fully qualified name of the factory class that creates the initial context. The initial context is the starting point for JNDI naming operations.
Java Naming Security Authentication	The security level to use in JNDI naming operations. Enter one of the following values: <ul style="list-style-type: none"> ▪ none: Authentication is not required. Use this for anonymous access. ▪ simple: Authentication requires a user name and password. You must enter the security principal and credentials below.
Java Naming Security Principal	The DN of the security principal used for connecting to the LDAP server.
Java Naming Security Credentials	The password of the naming security principal.
Users ParentDN	The parent DN of the user entries. This property specifies the root entry of the Users portion of the LDAP directory; for example, ou=People,dc=sun,dc=com .
UserDN Attribute Name In User	The name of the attribute in a user entry where the user's DN is defined. The default value is <i>entrydn</i> , which is the default name for the Sun Java System Directory Server attribute.
UserId Attribute Name In User	The name of the naming attribute in each user entry. The default name for this attribute in the Sun Java System Directory Server is <i>uid</i> , but can also be cn .
Roles Parent DN	The parent DN of the role entries. This property specifies the root entry of the Roles portion of the LDAP directory; for example, dc=sun,dc=com .

Property	Description
Role Name Field In RoleDN	The name of the attribute in a role entry that specifies the name of the role. The default name for this attribute in the Sun Java System Directory Server is cn .
Role Name Attribute Name In User	The name of the attribute in a user entry that specifies the DN of the roles to which a user is assigned. The default value is <i>nsroledn</i> , which is the default attribute name in the Sun Java System Directory Server.
Group ParentDN	The parent DN of the group entries. This property specifies the root entry of the Groups portion of the LDAP directory; for example ou=Groups,dc=sun,dc=com .
Group DN Attribute Name In Group	The name of the attribute in a group entry that specifies the name of the group. The default value is <i>entrydn</i> , which is the default name for the Sun Java System Directory Server attribute.
Group Name Field In Group DN	The name of the attribute in a group entry that specifies the name of the group. The default name for this attribute in the Sun Java System Directory Server is cn .
Group Of User Filter Under Groups ParentDN	The LDAP search filter used to retrieve all of a user's groups. This property follows the syntax supported by the java.text.MessageFormat class with {1} marking where the user's DN should be inserted. For example, uniquemember={1} .
UserId Attribute Manager	<p>The name of the attribute in a user entry that specifies the person the user reports to. The default value is <i>manager</i>, which is the attribute the Sun Java System Directory Server provides for this purpose. You can also use the <i>entrydn</i> for this purpose, or you can create custom attributes to define an upward reporting structure.</p> <p><i>Note: To enable task escalation and re-assignment, this value must be manager and each user entry in the LDAP directory must include a manager attribute that specifies the supervisor by their entrydn.</i></p>

Property	Description
UserId Attribute Subordinate	<p>The name of the attribute in a user entry that specifies the people who report to the user. The default value is <i>directReports</i>, which is not used in the Sun Java System Directory Server. You can use a default attribute for Sun, such as <i>secretary</i> or entrydn, or you can create custom attributes to define the downward reporting structure.</p> <p><i>Note: To enable task escalation and re-assignment, this value must be ent rydn and the reporting structure for each user must be defined in their entrydn attribute in the LDAP directory.</i></p>
UserId Attribute Email	<p>The name of the attribute in a user entry that specifies an email address. The default value is <i>mail</i>, which is the attribute the Sun Java System Directory Server provides for this purpose. This is used for email notifications (as defined in the Worklist Manager window for a user activity in the Business Process).</p>
UserId Attribute Given Name	<p>The name of the attribute in a user entry that specifies a user's first name. The default value is <i>givenName</i>, which is the attribute the Sun Java System Directory Server provides for this purpose. This is used during email notification.</p>

Configuring a Microsoft Active Directory Connection

To use Microsoft Active Directory with the Worklist Manager, you must specify certain information about the LDAP directory structure so the Worklist Manager knows where to find the user information defined in the directory. You can use your existing directory structure as long as there is a mechanism for defining a user reporting hierarchy.

▼ To configure a Microsoft Active Directory connection

- 1 From the Environment Explorer tab, right-click the Worklist Manager External System, and then click Properties.
- 2 On the properties page that appears, verify that the Connection Parameters property is set to ActiveDirectoryConnection.
- 3 Expand WLMConnector External System Configuration, and then click Sun Java System Directory Server/ADS.
The Properties dialog box appears.
- 4 Enter the property values for the properties described in the following table.

Note – Depending on how your LDAP directory is set up, not all of these fields are required. The default configuration is not necessarily illustrative of an actual implementation.

5 Click OK to close the Properties dialog box.

Property	Description
Java Naming Provider URL	The URL of the LDAP server. The format of the URL is ldap://<host>:<port> , where <host> is the name of the computer hosting the LDAP server, and <port> is the port number on which the LDAP server is listening for requests.
Java Naming Factory Initial	The fully qualified name of the factory class that creates the initial context. The initial context is the starting point for JNDI naming operations.
Java Naming Security Authentication	The security level to use in JNDI naming operations. Enter one of the following values: <ul style="list-style-type: none"> ■ none: Authentication is not required. Use this for anonymous access. ■ simple: Authentication requires a user name and password. You must enter the security principal and credentials below.
Java Naming Security Principal	The DN of the security principal used for connecting to the LDAP server.
Java Naming Security Credentials	The password of the naming security principal.
Users ParentDN	The parent DN of the user entries. This property specifies the root entry of the Users portion of the LDAP directory; for example, cn=Users,dc=sun,dc=com .
UserDN Attribute Name In User	The name of the attribute in a user entry where the user's DN is defined. If you are using the default schema for Active Directory, enter <i>distinguishedName</i> for this property.
UserId Attribute Name In User	The name of the attribute in a user entry that defines the user's login ID. The default name for this attribute in Active Directory is <i>sAMAccountName</i> .
Roles ParentDN	The parent DN of the role entries. This property specifies the root entry of the Roles portion of the LDAP directory; for example, ou=SunRoles,dc=sun,dc=com .

Property	Description
Role Name Field In RoleDN	The name of the attribute in a role entry that specifies the name of the role. The default name for this attribute in Active Directory is <i>cn</i> .
Role Name Attribute Name In User	The name of the attribute in a user entry that specifies the roles to which a user is assigned. The default value, <i>nsroledn</i> , does not apply to Active Directory. The default attribute used by Active Directory is <i>memberOf</i> .
Group ParentDN	The parent DN of the group entries. This property specifies the root entry of the Groups portion of the LDAP directory; for example cn=users,dc=sun,dc=com .
Group DN Attribute Name In Group	The name of the attribute in a group entry that specifies the DN of the group. If you are using the default schema for Active Directory, enter <i>distinguishedName</i> for this property.
Group Name Field In Group DN	The name of the attribute in a group entry that specifies the name of the group. The default name for this attribute in Active Directory is <i>cn</i> .
Group Of User Filter Under Groups ParentDN	The LDAP search filter used to retrieve all of a user's groups. This property follows the syntax supported by the java.text.MessageFormat class with {1} marking where the user's DN should be inserted. For example (for Active Directory only), (&(member={1})(objectclass=group)) .
UserId Attribute Manager	The name of the attribute in a user entry that specifies the person a user reports to. The default value is <i>manager</i> , which is the attribute that Active Directory provides for this purpose. You can also create custom attributes to define a reporting structure.
UserId Attribute Subordinate	The name of the attribute in a user entry that specifies the people who report to the user. The default value is <i>directReports</i> , which is the attribute that Active Directory provides for this purpose. In Active Directory, directReports is linked referentially to manager , above.

Property	Description
UserId Attribute Email	The name of the user attribute that specifies an email address. The default value is <i>mail</i> , which is the attribute that Active Directory provides for this purpose. This is used for email notifications (as defined in the Worklist Manager window for a user activity in the Business Process).
UserId Attribute Given Name	The name of the user attribute that specifies a user's first name. The default value is <i>givenName</i> , which is the attribute that Active Directory provides for this purpose. This is used during email notification.

