



# BEA AquaLogic® Enterprise Repository

## Repository Integration with Eclipse Using WorkSpace Studio

Version 3.0 RP1  
Revised: February 2008



# Contents

## 1. Getting Started

Repository Access and Assembly Model Composition . . . . .	1-2
Example Use Cases . . . . .	1-2
Locate and Download an Asset into an Eclipse Project. . . . .	1-2
Locate an Asset and View Its Metadata in ALER . . . . .	1-2
Submit an Assembly Model to ALER . . . . .	1-2
Submit an Eclipse Project to ALER. . . . .	1-3
Advancing the Asset Registration Status . . . . .	1-3
Related Documentation . . . . .	1-3
Terminology . . . . .	1-5

## 2. Administrative Prerequisites and Establishing the ALER Connection

Configuring the ALER Plug-ins for Repository Access in WorkSpace Studio . . . . .	2-2
Uninstalling the ALER Plug-ins . . . . .	2-2
Installing AquaLogic Products After Installing ALER . . . . .	2-2
Prerequisites for Using the ALER Plug-ins With WorkSpace Studio . . . . .	2-2
Installing the AquaLogic Solution Package. . . . .	2-3
Assign Users to an ALER Project . . . . .	2-3
Enabling the Assets-in-Progress Properties . . . . .	2-3
Enabling Assembly Model Submission Properties . . . . .	2-4
SiteMinder . . . . .	2-5

Java JDK .....	2-5
XML Parsing .....	2-5
Configuring the ALER Connection .....	2-5

### 3. Viewing and Querying Assets In the Repository

Understanding the Enterprise Repository Access View .....	3-2
Searching for Assets .....	3-4
The Query Pane .....	3-4
Query Pane Toolbar .....	3-4
Categorizations Tree Viewer .....	3-5
Additional Criteria Table .....	3-5
The Results Pane .....	3-6
Results Pane Toolbar .....	3-7
Working with the Results Table .....	3-8
Viewing Assets Associated With an ALER Project .....	3-9
Overview .....	3-9
Using the Project Team Assets View .....	3-9
Viewing Asset Properties and Details .....	3-11
Viewing Asset Relationships .....	3-12
Downloading Asset Artifacts .....	3-15

### 4. Submitting Assets to ALER

Overview .....	4-2
Assembly Models .....	4-2
ALER Projects .....	4-2
Submitting Assembly Models to the Repository .....	4-3
Specifying Model Submission Information .....	4-3
Resolving External Service References .....	4-6

Guidelines for Submitting Models to ALER . . . . .	4-8
Asset Naming Algorithm . . . . .	4-8
Producing Projects . . . . .	4-8
No Support Imported WSDLs With Duplicate Assets . . . . .	4-8
Resubmitting an Existing Model . . . . .	4-9
Submitting Eclipse Project Files to ALER . . . . .	4-10
Submitting Archived Assets . . . . .	4-12

## 5. Setting Eclipse Preferences for ALER

Overview . . . . .	5-2
Configuring the ALER Connection . . . . .	5-2
Configuring Automated Usage Detection . . . . .	5-5



# Getting Started

This section contains information on the following subjects:

- [“Repository Access and Assembly Model Composition”](#)
- [“Example Use Cases”](#)
- [“Related Documentation”](#)

## Repository Access and Assembly Model Composition

The ALER plug-in for WorkSpace Studio provides repository integration within the Eclipse IDE so you can easily search for and use assets from the repository without leaving the Eclipse environment. Assets and any associated artifacts are downloaded directly to your Eclipse workspace. Integration also provides a convenient way to submit assets to ALER for use throughout the enterprise.

Repository Access within the Eclipse workspace also provides a view into ALER that enables you to save/submit assets to the repository, download artifacts and assets from the repository, query the repository, and view the contents of the repository. It also enables you to advance the asset registration status.

WorkSpace Studio also includes the Service Assembly Modeler (SAM) plug-in for WorkSpace Studio that supports the composition and visualization of composite applications. Composition occurs within an Eclipse-based development environment. In this environment, development teams can quickly assemble composite applications by re-using services that already exist in the repository, or by creating new services that can be submitted to the repository and reused by others. You can also view the assembly model for composite applications, as well as the XML definition of the composite.

## Example Use Cases

### Locate and Download an Asset into an Eclipse Project

Perform a simple keyword search in ALER to locate an asset and download its payload into an Eclipse project.

### Locate an Asset and View Its Metadata in ALER

Perform simple or complex searches in Eclipse to locate an asset in ALER in order to view the asset's metadata.

### Submit an Assembly Model to ALER

An Assembly Model is a set of related assets representing the structure of a WorkSpace Studio integration project in terms of the SCA standard. When a project is submitted from WorkSpace Studio, an Assembly Model is automatically created. The model consists of a root asset, of type *AssemblyModel*, that logically contains all the other assets in the model. The model assets

represent Components, Services, Interfaces, and Bindings defining a WorkSpace Studio project, as well as Composites that define a hierarchical composition structure. The relationships used to connect the assets in the Assembly Model are ALER system-supplied relationships.

## Submit an Eclipse Project to ALER

Submitting an entire Eclipse project to the repository.

## Advancing the Asset Registration Status

[TBD]

## Related Documentation

- [AquaLogic Service Bus \(ALSB\)](#) – An intermediary for use as a core element of distributed services networks. It enables service-oriented architecture (SOA) allowing accelerated service re-use and deployment.
  - [ALSB 3.0 documentation](#)
  - [ALSB 2.6 RP1 documentation](#)
- [AquaLogic Data Services Platform \(ALDSP\)](#) – The SOA solution for reading and writing of enterprise information through data services.
  - [ALDSP 3.0 documentation](#)
  - [ALDSP 2.5 documentation](#)
- [AquaLogic Service Registry \(ALSR\) 3.0](#) – provides a standards based mechanism for publishing and discovering Web services and related SOA resources such as WSDLs, XML Schemas, and XSL Transformations (XSLT).
- [WebLogic Integration \(WLI\) 9.2](#) – a single, comprehensive, standard-based solution for connecting applications, databases, enterprise information systems, business processes, and trading partners.
- [ALER on dev2dev](#) – Information and resources for architects, developers, and others, including an FAQ and an overview of assembly models in Service Component Architecture (SCA).
- [SOA Governance: Essential To Your Business](#) – Learn how effective SOA governance is an essential element in any enterprise transformation strategy.

## Getting Started

- [SOA Blog on dev2dev](#) – Keep on top of the latest SOA blogs.

## Terminology

[Table 1-1](#) defines the terms and acronyms used this document:

**Table 1-1 Terminology**

<b>Terms</b>	<b>Definition</b>
WorkSpace Studio	The Eclipse IDE, branded for use by BEA products. WorkSpace Studio is common to all BEA products that use an Eclipse-based IDE.
ALER Plug-in	Provides repository integration within WorkSpace Studio, so you can easily search for and use assets from the repository without leaving the Eclipse environment.
SAM Plug-in	Within WorkSpace Studio, SAM supports the composition and visualization of composite applications.
Assembly Model	A set of related assets representing the structure of a WorkSpace Studio integration project in terms of the SCA standard.
ALER Projects	Projects are the primary means of gathering metrics in ALER. ALER tracks assets produced by projects, as well as assets consumed by projects.
SFID (Software File Identification)	Allows automated usage detection for assets. SFID tags selected files within an asset with a unique ID. This SFID is then used to detect when and where an asset is used. (Available in the Advanced Edition of ALER.)

## Getting Started

# Administrative Prerequisites and Establishing the ALER Connection

This section contains information on the following subjects:

- [“Configuring the ALER Plug-ins for Repository Access in WorkSpace Studio”](#) on page 2-2
- [“Prerequisites for Using the ALER Plug-ins With WorkSpace Studio”](#) on page 2-2
- [“Configuring the ALER Connection”](#) on page 2-5

## Configuring the ALER Plug-ins for Repository Access in WorkSpace Studio

For instructions on installing the ALER plug-ins for repository access within the Eclipse-based WorkSpace Studio IDE, see the [ALER Installation Guide](#).

### Uninstalling the ALER Plug-ins

When uninstalling the ALER plug-ins, WorkSpace Studio will also be removed if it was installed by the ALER installer. This may not be desirable if other plug-ins were since added to WorkSpace Studio. To avoid uninstalling additional WorkSpace Studio plug-ins, deselect the ALER plug-ins for Eclipse option from the list of components to uninstall.

### Installing AquaLogic Products After Installing ALER

If AquaLogic Service Bus or AquaLogic Data Service Platform applications are installed after the ALER plug-in is installed, then Eclipse must be launched using the `-clean` flag.

## Prerequisites for Using the ALER Plug-ins With WorkSpace Studio

You should complete the prerequisites described in this section before using the ALER plug-ins for WorkSpace Studio.

- [“Installing the AquaLogic Solution Package”](#)
- [“Assign Users to an ALER Project”](#)
- [“Enabling the Assets-in-Progress Properties”](#)
- [“Enabling Assembly Model Submission Properties”](#)
- [“SiteMinder”](#)
- [“Java JDK”](#)
- [“XML Parsing”](#)

## Installing the AquaLogic Solution Package

In order to be able to submit AquaLogic Service Bus and AquaLogic Data Service Platform projects as assembly models to the repository, you must first import the AquaLogic Solution Package that is bundled with your installation into ALER.

1. Start the ALER Import/Export tool, as described in the [ALER Import/Export Guide](#).
2. Select the **Import** tab.
3. Navigate to the `BEA_HOME\repository30\core\tools\solutions` folder.
4. Select the `ALER30-AquaLogic-Solution-Pack.zip` as the target file to import into ALER.
5. Click **Next**, and then click **Next** again to start the import process.
6. Click **Finish** to complete the process.
7. After importing the AquaLogic solution package, you must reestablish connectivity to the ALER plug-ins by using the Eclipse Preferences page, as described in [Configuring the ALER Connection](#).

Once connectivity is established, then the ALER plug-in imports all the necessary asset types, taxonomy, relationships, and other entities for AquaLogic application integration to ALER. Once these entities are imported, they will be available whenever you connect the ALER plug-in to the enterprise repository.

## Assign Users to an ALER Project

In order to download assets from the repository, users must be assigned to at least one ALER project. An ALER project administrator can assign users to projects using the ALER Projects page.

Obtain the Eclipse integration path from the ALER administrator. (For example, `http://appserver.example.com/aler-web/eclipse`).

## Enabling the Assets-in-Progress Properties

Two system settings must be enabled in order to activate Assets-in-Progress when using the WorkSpace Studio with ALER.

This procedure is performed on the ALER Admin screen.

1. Click **System Settings**.

2. Click **General Settings** in the System Settings section.
3. Enter the property **cmee.asset.in-progress** in the Enable New System Setting box and click **Enable** to reveal this hidden property.
4. Make sure the **Asset in Progress** property is set to **True**.
5. Click **Save**.
6. Enter the property **cmee.asset.in-progress.visible** in the Enable New System Setting box and click **Enable** to reveal this hidden property.
7. Make sure the **Asset in Progress** property is set to **True**.
8. Click **Save**.

The Registration Status drop-down menu will now appear in the Search section on the AquaLogic Enterprise Repository Assets screen. For more information about Assets-in-Progress, see the [ALER Registrar Guide](#).

## Enabling Assembly Model Submission Properties

In order to be able to submit AquaLogic Service Bus and AquaLogic Data Service Platform projects as assembly models to the repository, this capability must be enabled in the ALER System Settings. You can also enable the logging of asset submissions from external endpoints.

1. Click **System Settings** in the sidebar on the ALER **Admin** page.
2. Enter the property **cmee.tooling.submission.enabled** in the Enable New System Setting box and click **Enable** to reveal this hidden property.
3. Set the **Asset Submission from Integrated Endpoint** property to **True** to enable asset submissions generated through integrations with external endpoints.
4. Click **Save**.
5. Enter the property **cmee.tooling.submission.logging** in the Enable New System Setting box and click **Enable** to reveal this hidden property.
6. Set the **Submission Logging of Integrated Endpoint** property to **True** to enable the logging of import/export jobs controlling asset submissions through integrations with external endpoints.
7. Click **Save**.

For more information about System Settings, see the [ALER Administration Guide](#).

## SiteMinder

If ALER is or will be configured to be secured by Siteminder, the policy server must be configured to ignore (or unprotect) the following URL:

```
http://appserver.example.com/aler-web/eclipse/
```

## Java JDK

The Java Cryptography Extension (JCE) is required. It is provided in JDK v1.4, and is available as an optional package in JDK 1.2.x through 1.5.x. Note that ALER plug-ins for use with Eclipse 3.x require JDK v 1.5.x or later.

## XML Parsing

Since Editor and Viewer metadata is represented as CDATA-escaped XML, some XML parsers may exceed their entity expansion limit when communicating with ALER. For example, if you have defined a large number of Asset Types in ALER, then you may need to increase the Entity Expansion Limit of your XML parser.

On some popular parsers, the default entity expansion limit is set to 64,000. This limit can be increased on JAXP-compliant processors by passing a command-line parameter called `entityExpansionLimit`. The `entityExpansionLimit` can be increased by passing a VM argument on the Eclipse command-line (modify the Eclipse desktop shortcut). For example, set the target of the shortcut to the following:

```
c:\eclipse\eclipse.exe -debug -consolelog -vmargs  
-DentityExpansionLimit=1048576
```

## Configuring the ALER Connection

When you invoke an action on a repository, such as querying or publishing assets, before repository connectivity has been established, then the Connect to Enterprise Repository wizard will either be automatically displayed (in the case of querying the repository), or will be launched by an explicit user gesture.

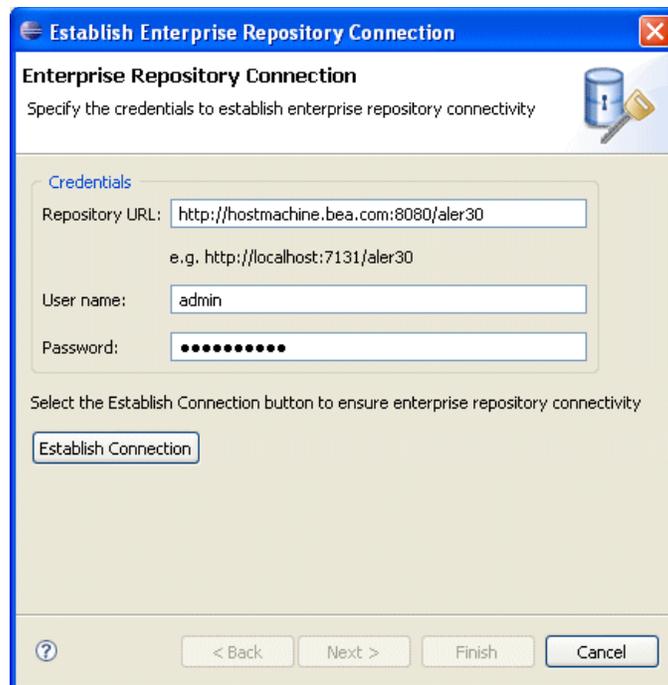
**Note:** If credential information had been specified in a previous session, the wizard will display this persisted information when it is launched.

1. In the Credentials area, enter the server location and login credentials, as follows:

## Administrative Prerequisites and Establishing the ALER Connection

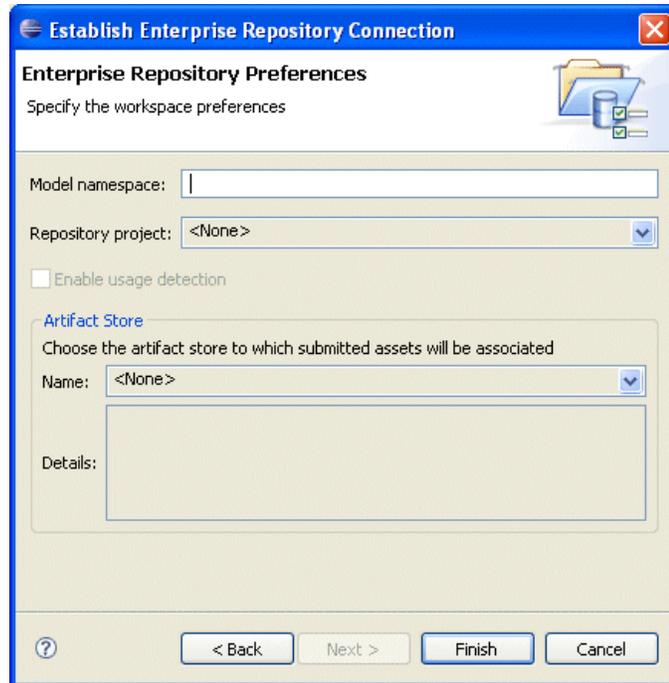
- **Repository URL** – the URL of the repository server. The URL must include the host, port, and ALER server name. For example, `http://localhost:7001/aler30`.
- **User Name** – user name to gain access to the repository.
- **Password** – password to gain access to the repository.

**Figure 2-1 Establish Enterprise Repository Connection**



2. Click the **Establish Connection** button to ensure enterprise repository connectivity.  
If a connection cannot be established, an appropriate error message will be displayed.
3. Once connectivity is established, you can either:
  - Click **Finish** to exit.
  - Click **Next** to select your workspace preferences (skip to Step 4).

Figure 2-2 Specify Workspace Preferences



4. Once connectivity is established, you can specify your workspace preferences:
  - Enter a **Model Namespace** to use as a default for your all of your projects. The Namespace provides a means to organize your models, with the Namespace pre-pended to the names of all the assets in the model in the repository. However, you can change the Namespace on a project-by-project basis (such as when submitting assets), and the new Namespace will only be saved for that project, but will not affect the Workspace Preference name.
  - Select a **Repository project** in ALER that the submitted model will be associated with. Asset usage is tracked in the repository and attributed to repository projects, which typically represent software development programs, business initiatives, etc.
  - **Enable usage detection** – If you selected an ALER project as the workspace default, usage detection will be enabled for the default ALER project. For more information about workspace preferences, see [Chapter 5, “Setting Eclipse Preferences for ALER.”](#)

## Administrative Prerequisites and Establishing the ALER Connection

5. The **Artifact Store** area displays the name of a preconfigured Artifact Store that the submitted assets will be associated with. Artifact Stores contain the files relevant to assets in ALER and are configured in the ALER console by an administrator. The **Details** box may also display some additional information about the Artifact Store.
6. Click **Finish** to exit.

# Viewing and Querying Assets In the Repository

This section contains information on the following subjects:

- [“Understanding the Enterprise Repository Access View”](#)
- [“Searching for Assets”](#)
- [“Viewing Assets Associated With an ALER Project”](#)
- [“Viewing Asset Properties and Details”](#)
- [“Viewing Asset Relationships”](#)
- [“Downloading Asset Artifacts”](#)

## Understanding the Enterprise Repository Access View

The Enterprise Repository Access View view provides access to assets and artifacts in the Enterprise Repository. You can search for assets matching various criteria or view assets that may be of interest to your project. For selected assets, you can view details and relationships, and can also download associated artifacts into your workspace.

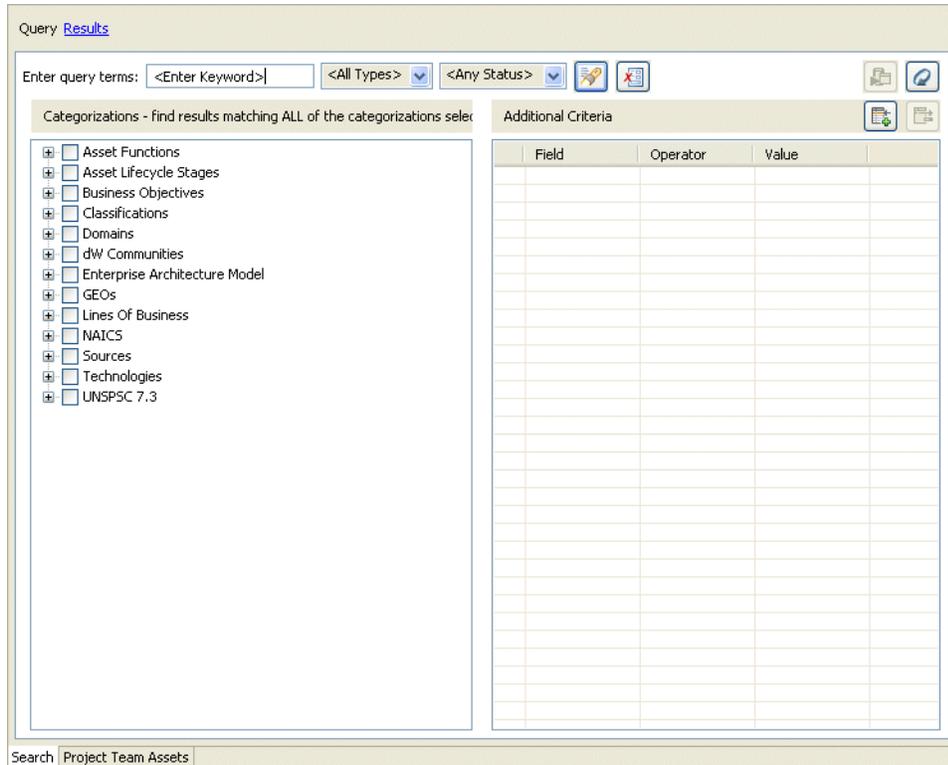
## Displaying the Enterprise Repository Access View

To display the Enterprise Repository Access view:

1. Open the Window menu.
2. Select **Show View**.
3. Select **Other**.
4. Open the **AquaLogic** folder.
5. Select the **Enterprise Repository Access** option.

The Enterprise Repository Access view is displayed as a tabbed pane containing Search and Project Team Assets panes.

Figure 3-1 Two-Tabbed Enterprise Repository Access View



The **Search** tab enables querying of assets and displays results based upon specified criteria, as discussed in [“Searching for Assets.”](#)

The **Project Team Assets** tab allows you to view all assets associated with a specified ALER project, as discussed in [“Viewing Assets Associated With an ALER Project.”](#)

## Searching for Assets

The Search tab displays a toolbar at the top that is visible whether the active view is the Query pane or the Results pane. You can toggle between the two displays by clicking either the **Query** link or the **Results** link, depending on which pane is active at the time.

### The Query Pane

The Query pane, as shown in [Figure 3-1](#), allows you to easily construct asset query filtering to view assets that may be of interest to your project, including Assembly Model assets imported from an integrated external endpoint, such as an AquaLogic Service Bus or an AquaLogic DSP project, and common ALER assets already defined in your ALER instance.

### Query Pane Toolbar

[Table 3-1](#) describes the options that are available to query assets.

**Table 3-1 Asset Query Options**

Icon	What it means...
<b>Enter Keyword</b> box	When the query is invoked, all results will contain this keyword (if one has been specified) as part of a well-defined set of fields (e.g. name, description, categories, etc.)
<b>Asset Type</b> menu	Displays all asset types, which allows you to filter assets by a specified selection. When a selection occurs, the specific fields associated with this asset type are added to the Additional Criteria table. Three important asset types to note are: <ul style="list-style-type: none"> <li>• &lt;Assembly Model Assets&gt; – assets that are submitted from an integrated external endpoint, such AquaLogic DSP and AquaLogic Service Bus.</li> <li>• &lt;Common Assets&gt; – legacy assets from an existing ALER instance.</li> <li>• &lt;Consumable Assets&gt; – integrated endpoint services that are consumable by Workspace Studio plug-ins, which restricts them to services exposed or promoted as public services in the assembly model.</li> </ul>
<b>Registration Status</b> menu	Displays all registration status values, which allows you to filter assets by a specified selection.
	<b>Connect to enterprise repository</b> – If no connection to a repository has been established, this button invokes the Connect to Enterprise Repository wizard, which will prompt you for connection credentials.

**Table 3-1 Asset Query Options (Continued)**

Icon	What it means...
	<b>Perform query</b> – If a repository connection has been established, this button queries the enterprise repository based upon the specified criteria. Once the results are obtained, the Results pane will automatically become visible.
	<b>Clear all query criteria</b> – Clears all query criteria fields to prepare for another query.
	<b>Refresh enterprise repository information</b> – Queries the enterprise repository for its metadata, and repopulates the appropriate fields based upon the results of this query.

## Categorizations Tree Viewer

The Categorizations tree viewer provides a convenient way to query for assets associated with a set of categories, with most option containing appropriate subcategories. If a category is selected all its existing subcategories are also selected. If a subcategory has a parent, the parent checkbox state is also selected. Conversely, if a category is deselected all its subcategories are deselected.

**Note:** If more than one categorization type is used in the criteria, the selected types are ANDed together. There is no OR option.

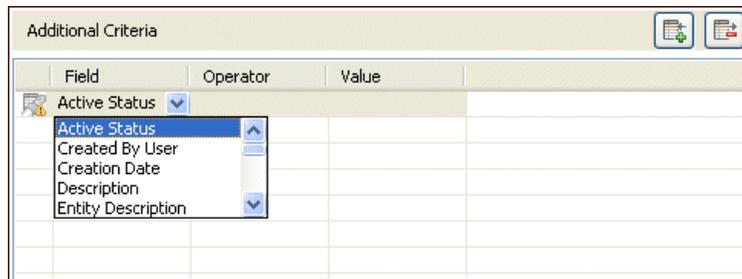
## Additional Criteria Table

The Additional Criteria table contains an additional set of filters to impose upon a query. These criterion are joined together to formulate the query constraint.

1. Click the **Add Criteria** button to activate a new criteria row.

The Field column presents a drop-down menu that allows you to select from a list of fields for the query filter. The list of fields based upon the selected asset type. If no asset type is selected, the list of fields are the fields that are common to all asset types.

**Figure 3-2 Additional Criteria Table**



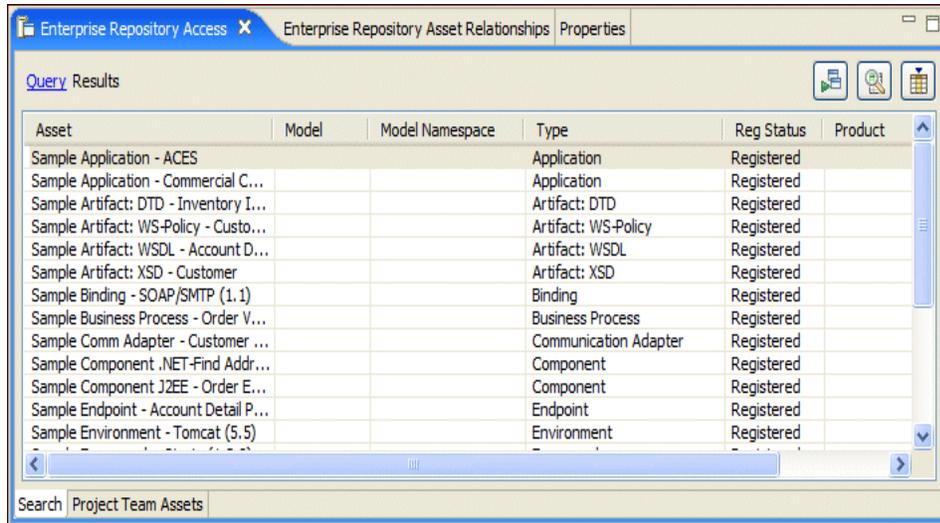
2. Select a field from Field column menu. As a result, the Operation and Value cells are updated to reflect the possible operations and pre-defined set of values associated with the selected field.
3. From the Operation column's menu, select the operation (equals, not equals, etc.) for the query filter row.
4. If the specified field has a pre-defined set of values, then a drop-down menu is displayed to specific a comparison value. If the specified field does not have a pre-defined set of values, then a text field is displayed to specify a comparison value.
5. If necessary, use the **Remove Criteria** button to remove selected rows from the table.

If the query criterion specified in the row is invalid or incomplete, the **Info** (i) icon displays in the left-most status column.

## The Results Pane

The Results pane shows all assets that satisfied the search criteria established in the Query pane. It displays information for all query results, such as the asset name, asset type and registration status.

Figure 3-3 Results Pane



## Results Pane Toolbar

Table 3-1 describes the options that are available on the Results pane.

Table 3-2 Results Pan Icons

Icon	What it means...
	<b>Toggle the display of the asset detail view</b> – Displays a selected asset’s details in a web browser view, or to not launch the web-based view when an asset is selected.
	<b>Display the asset properties and relationships view</b> – Enables you to view the details of a selected asset in the Results pane.
	<b>Customize ALER query result columns</b> – Opens the Customize Results Columns dialog, which enables you to rearrange the order of the results columns.

## Working with the Results Table

[Table 3-3](#) describes the columns in the Query Results table.

**Table 3-3 Query Results Table**

Row	What it means...
Asset	The simple name of the asset.
Model	The name for Assembly Model assets imported into ALER.
Model Namespace	The group the model is a member of. The Namespace is pre-pended to the names of all the assets in the Assembly Model in the repository.
Version	The asset version number.
Type	The asset type.
Reg Status	The asset registration status.
Product	The product where the asset originated.
# of Artifacts	The number of artifacts associated with the asset. If the number is 1 or higher the <b>Download artifacts</b> option is enabled.

- When you select a results row, the asset's properties, details, and relationships are displayed in the Properties View, Details View, and Enterprise Repository Relationships View respectively, if the relevant view is visible.
- Right-click an asset to access these options from the menu:
  - **Download artifacts** – if the value in the **# of Artifacts** column is 1 or higher, you can download an asset's artifacts and its dependencies into your Eclipse project (if the SCM information was properly set up when the assembly model was submitted). For more information, see [“Downloading Asset Artifacts.”](#)
  - **Subscribe/Unsubscribe** – subscribes to the selected asset if currently unsubscribed, and visa versa.
  - **Show in asset details view** – opens the Asset Details view.
- You can sort the asset results by clicking any column header.

## Viewing Assets Associated With an ALER Project

The Project Team Assets view displays a list of assets that have been prescribed to your project, as well as assets that are already in use in the project.

### Overview

Through the Enterprise Repository, analysts, architects, technical leads, and others that are involved in the design stages of a project, can create a list of assets that might fulfill a project's requirements. The lists of assets are captured in compliance templates in the repository, and the compliance templates are associated with an ALER project. For more information on compliance templates, refer to the *ALER Compliance Templates Guide*.

**Note:** Compliance Templates are only available in the ALER Advanced Edition.

From the Enterprise Repository Access viewer in Workspace Studio you can view a list of assets appearing in all of the Compliance Templates assigned to your project. The viewer will indicate which of the assets have been used by you and/or other project members. The viewer will also display other assets that are already in use in the project.

### Using the Project Team Assets View

The Project Team Assets view is decoupled from a workspace ALER project. Therefore you can view assets for any ALER project without having to set or alter the workspace default. In addition, there are also ancillary views to display asset details without having to launch an external Web browser to view an asset's details.

1. From the **Repository Project** menu, select a project from the set of all enterprise repository projects associated with the current repository connection. The items in the list are prefixed with the name of the repository in which the associated project resides.



2. Click the **Query** button to query the enterprise repository for all assets associated with the specified repository project.

If a repository connection has been established, the Project Team Assets table is populated with the set of assets associated with the specified project. Each of the columns in the table identifies how the asset relates to the project in the repository. You can also resort each column using the column header.

**Figure 3-4 Project Team Assets Pane**

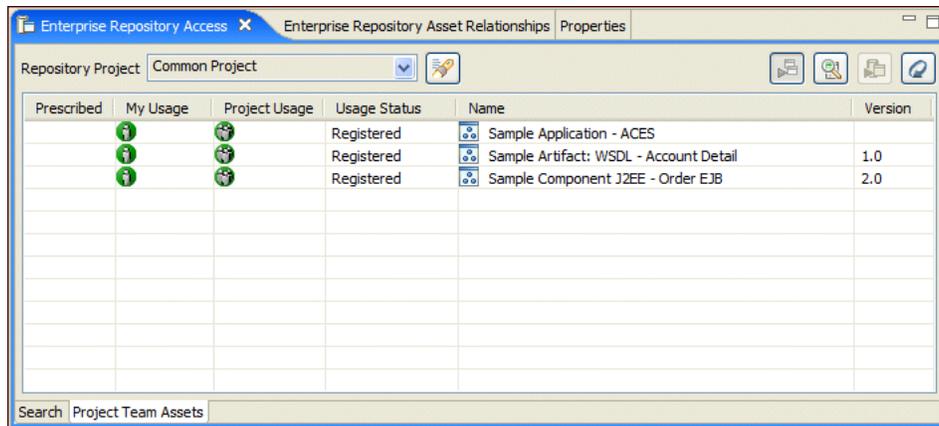


Table 3-4 describes the other icons and columns that are available in the Project Team Assets view.

**Table 3-4 Project Team Assets Table**

Icon/Column Name	What it means...
	<b>Toggle the display of the asset detail view</b> – Displays a selected asset’s details in a web browser view, or to not launch the web-based view when an asset is selected.
	<b>Display the asset properties and relationships view</b> – Allows you to view the details of a selected asset in the Results pane.
	<b>Connect to enterprise repository</b> – If no connection to a repository has been established, this button invokes the Connect to Enterprise Repository wizard, which will prompt you for connection credentials.
	<b>Refresh enterprise repository information</b> – Queries the enterprise repository for its metadata, and repopulates the appropriate fields based upon the results of this query.
Prescribed	Indicates whether or not an asset is prescribed and has been assigned to a project using a Compliance Template. A Compliance Template may be used to prescribe specific assets for use in any projects to which the compliance template is applied.

**Table 3-4 Project Team Assets Table (Continued)**

Icon/Column Name	What it means...
My Usage	Identifies assets you have used/downloaded for the project.
Project Usage	Identifies assets that have been used/downloaded by any member of the project.
Usage Status	Identifies the status of the asset as it relates to the project: Prescribed – The asset has been assigned to the project but has not yet been used in or downloaded to the project. In Process – The asset has been used in/downloaded to the project. Rejected – The asset has been rejected for use in the project. (Not shown in example.) Deployed – The asset was used for a project that has been closed.
Name	Displays the asset name.
Version	Displays the asset version.

3. If necessary, click the **Refresh** button to repopulate the table with the results of this query.

## Viewing Asset Properties and Details

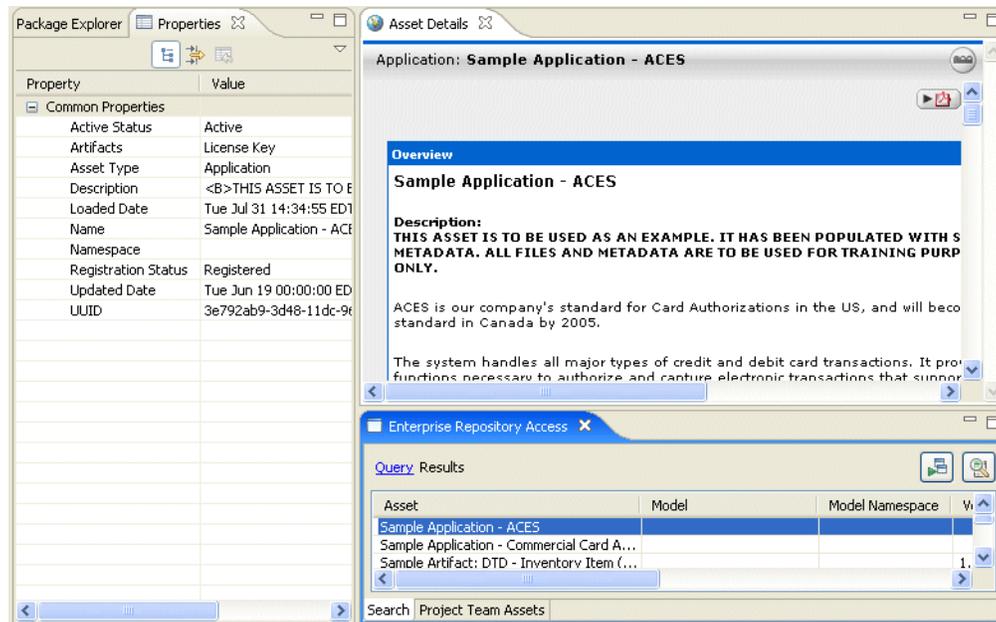
You display asset properties and values in the standard Eclipse Properties view. To launch the Asset Properties view, you can either:

- Select the **Enterprise Repository Assets Relationships** option under the AquaLogic Enterprise Repository view category, as described in [“Understanding the Enterprise Repository Access View.”](#)
- Click the **Display the asset properties and relationship views** button on either the Enterprise Repository Access tab’s Results or Project Team Assets toolbars, as described in [“The Results Pane”](#) and [“Viewing Assets Associated With an ALER Project.”](#)

The Asset Details view is another view that provides asset details in an embedded web-based browser view, which when enabled, calls out the enterprise repository application for details associated with the selected asset. See [“Searching for Assets”](#) for instructions on how to enable or disable rendering of the asset details view.

[Figure 3-5](#) shows the Properties view on the left side of the workspace being populated based upon the selection in the Results Pane. It also depicts the Asset Details view on the right, with its contents based upon the selected asset in the Results pane.

Figure 3-5 Asset Properties and Details Views



## Viewing Asset Relationships

The Enterprise Repository Asset Relationships view displays the relationships for a selected asset in a graphical format. The Asset Relationships view listens for asset selection, either within the Results pane or the Project Team Assets view, and updates itself based upon the relationships of the currently selected asset. If there are no assets currently selected, then this view will be blank.

You can launch the view using either of these methods:

- Select the **Enterprise Repository Assets Relationships** option under the AquaLogic Enterprise Repository view category, as described in [“Understanding the Enterprise Repository Access View.”](#)
- Click the **Display the asset properties and relationship views** button on either the Enterprise Repository Access tab’s Search or Project Team Assets toolbars, as described in [“Searching for Assets”](#) and [“Viewing Assets Associated With an ALER Project.”](#)

Figure 3-6 depicts the Asset Relationship view in the upper-right corner of the workspace. Its content is based upon the current selection in the Results pane. The view shows the selected asset in the middle of the graph as a stand-alone node, and shows all relationships that the asset is involved in, such as bi-directional and one-way relationships.

Those assets that are a *source* of a relationship with the selected asset are shown above the selected asset and have links pointing *to* the selected assets. Those assets that are a *target* of a relationship with the selected asset are shown above the selected asset and have links pointing *from* the selected assets.

Figure 3-6 Asset Relationships View

The screenshot displays the 'Enterprise Repository Asset Relationships' window. The main graph shows the selected asset, 'Sample Application - Commercial Card Authorization System', at the center. Above it, two other assets are shown: 'Sample Environm...' and 'Sample Application - A...'. Arrows labeled 'Runs' point from these two assets down to the central asset. Below the central asset, another asset 'Sample Comm Ad...' is shown with an arrow labeled 'Depends on' pointing from the central asset down to it. The right-hand pane shows the 'Asset Details' for the selected asset, including an overview, description, and metadata. The bottom pane shows 'Enterprise Repository Access' with a table of query results.

Asset	Model	Model Nam
Sample Application - ACES		
Sample Application - Commercial Card A...		
Sample Artifact: DTD - Inventory Item (...)		
Sample Artifact: WSC-Policy - Customer I		

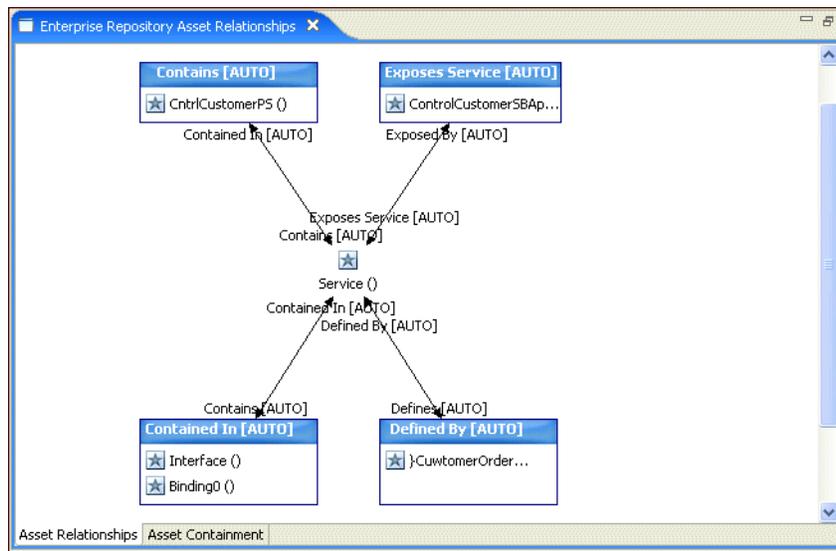
To fully understand the bi-directionality of the Asset Relationships view, read the graph in Figure 3-6 as follows. (Note that the selected asset is: Sample Application - Commercial Credit Authorization System.)

## Viewing and Querying Assets In the Repository

- Relationship 1: *Runs* – Sample Environment - Tomcat (source asset) *Runs* Sample Application - Commercial Credit Authorization System (target asset)
- Relationship 2: *Functional Equivalent Of* – Sample Application - ACES (source asset) is *Functional Equivalent of* Sample Application - Commercial Credit Authorization System (target asset)
- Relationship 3: *Depends On* – Sample Application - Commercial Credit Authorization System (source asset) *Depends On* Sample Comm Adapter - Customer Credit Information (target asset)

There are numerous instances where a particular relationship for the selected asset may be satisfied by more than one asset. For such cases, the assets that satisfy this relationship are grouped together where the relationship type is shown in the node label. [Figure 3-7](#) shows a selected asset that has three *Asset Prescribed By* relationships and how those assets are grouped to satisfy that relationship.

**Figure 3-7 Asset Relationships View With Multiple Assets Satisfying a Relationship**

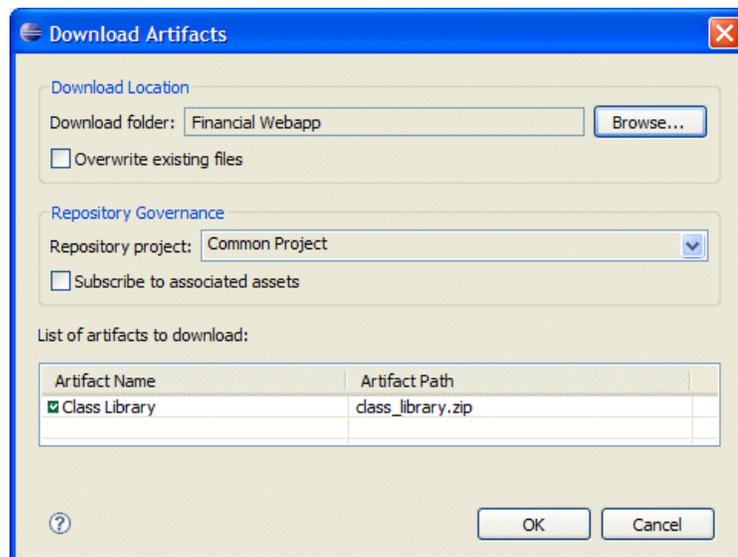


## Downloading Asset Artifacts

You can download an asset’s artifacts (i.e., payload) into an Eclipse project. Typically an asset payload is usually the functionality that a developer needs to use a service (such as a WSDL file) or incorporate into their code base (usually a binary). Within the asset metadata, links to supporting documentation, user guides, test cases, etc., are provided to better enable developers to reuse existing functionality.

1. Query the repository for the desired asset(s), as described in “[Searching for Assets.](#)”
2. Right-click the appropriate asset on the Results pane and if there are available artifacts, select **Download Artifacts** from the shortcut menu to open the Download Artifacts window.

**Figure 3-8** Download Artifacts



3. In the Download Location section:
  - Use the **Download Folder** field to navigate to an Eclipse project and select the destination folder for the download.
  - Select the **Overwrite existing files** check box to overwrite existing versions of the artifacts in the selected project folder.

## Viewing and Querying Assets In the Repository

4. In the Repository Governance section:
  - Select a valid project from the **Repository Project** list.
  - Select the **Subscribe to associated assets** check box to subscribe to all of the selected assets that had files associated with them, plus any associated artifact assets and dependencies for Service Assembly Models.
5. Verify your selection in the list of artifacts to download, and then click **OK**.  
Artifacts associated with the selected asset will be downloaded to the specified location.
6. Click **OK** again on the status confirmation window.
7. Open the selected destination folder to confirm the presence of the selected artifact file(s).

# Submitting Assets to ALER

This section contains information on the following subjects:

- [“Overview”](#)
- [“Submitting Assembly Models to the Repository”](#)
- [“Submitting Eclipse Project Files to ALER”](#)
- [“Submitting Archived Assets”](#)

## Overview

When submitting assets to ALER, the WorkSpace Studio plug-ins for Eclipse determine whether or not the project can be transformed into an assembly model or if the referenced Eclipse project is an existing ALER project, and then renders the appropriate wizard to submit the asset to the enterprise repository.

The Archive Submission wizard maintains support for legacy ALER functionality by allowing you to submit single and/or compound-payload assets to ALER via an archive ZIP file, as explained in [“Submitting Archived Assets.”](#)

## Assembly Models

An Assembly Model is a set of related assets representing the structure of a WorkSpace Studio integration project in terms of the SCA standard. When a project is submitted from WorkSpace Studio, an Assembly Model is automatically created. The model consists of a root asset, of type *AssemblyModel*, that logically contains all the other assets in the model. The model assets represent Components, Services, Interfaces, and Bindings defining a WorkSpace Studio project, as well as Composites that define a hierarchical composition structure. The relationships used to connect the assets in the Assembly Model are ALER system-supplied relationships.

See [“Submitting Assembly Models to the Repository.”](#)

## ALER Projects

Projects are the primary means of gathering metrics in ALER. ALER tracks assets produced by projects, as well as assets consumed by projects. Projects are also a channel for governance practices, with ALER Compliance Templates (usually taking the form of Architecture Blueprints or Project Profiles) being applied to projects.

The ALER plug-in for Eclipse allows projects in ALER to be synchronized with projects in the Eclipse IDE by supporting governance rules. If compliance templates have been applied to a project, the list of assets in the compliance templates will appear in the Eclipse IDE. If SFID/automatic usage detection has been enabled, the assets that are used in the developer’s project will be compared to the assets in ALER, and if a match is found, it will be recorded in ALER as reuse. This process streamlines the development experience, making ALER almost transparent to development teams. It also facilitates compliance reporting, allowing ALER to determine whether project teams have actually used the assets specified in the compliance templates.

See [“Submitting Eclipse Project Files to ALER.”](#)

## Submitting Assembly Models to the Repository

When you right-click an Eclipse project and select the **Submit to AquaLogic Enterprise Repository** from the shortcut menu, the Workspace Studio plug-ins first determine if connectivity has been established to ALER. If a connection is not yet established, the Connect to Enterprise Repository wizard is launched to establish connectivity.

Once connectivity is established, the Workspace Studio plug-ins determine whether or not the project can be transformed into an assembly model or an ALER project (see “[Submitting Eclipse Project Files to ALER.](#)”). If it can be transformed into an assembly model, the **Submit Assets to Enterprise Repository** wizard enables you to submit the project’s associated assets to the enterprise repository.

For example, if the referenced project is an integrated external endpoint, such as AquaLogic Data Services Platform or AquaLogic Service Bus, the wizard rendered represents the information required to submit the assets associated with the AquaLogic project.

---

**Tip:** In order to submit assembly models to the repository, you must be assigned to an Advanced Submitter role. If you are unsure of your role status, check with your ALER system administrator.

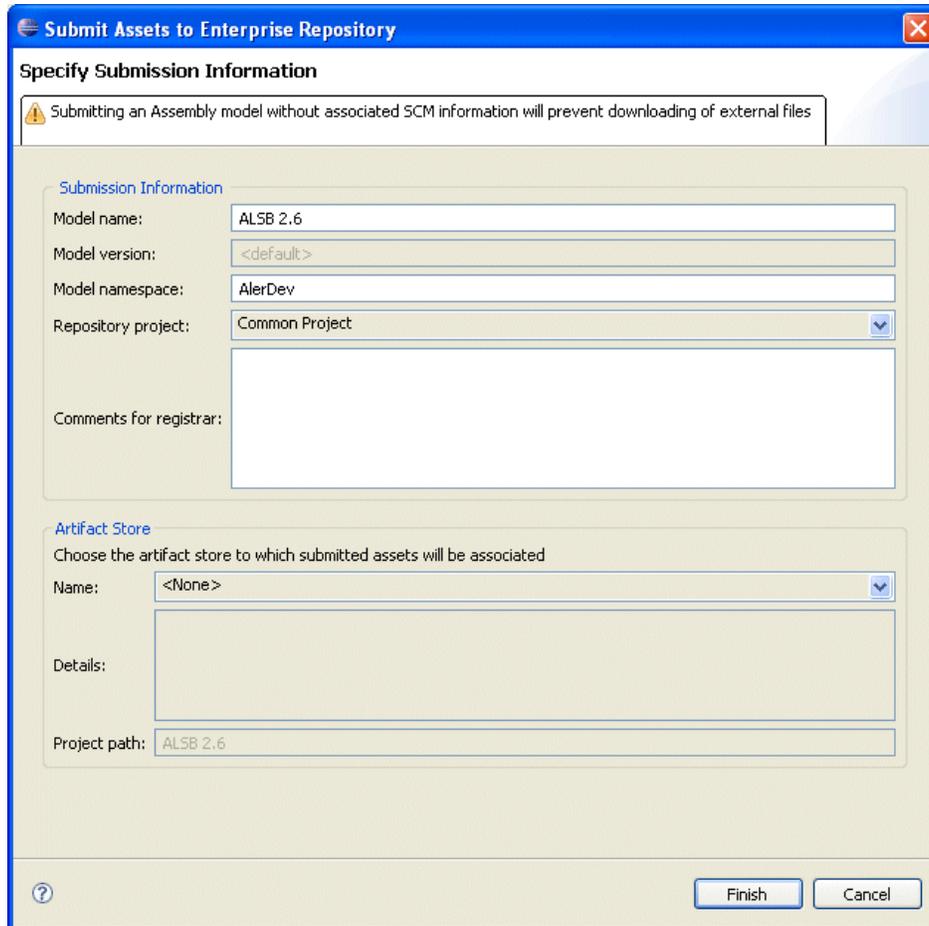
---

## Specifying Model Submission Information

When submitting assembly models, WorkSpace Studio plug-ins capture metadata associated with the submitted asset, such as model and associated Artifact Store information.

1. Right-click an Eclipse project and select **Submit to AquaLogic Enterprise Repository** from the shortcut menu.

Figure 4-1 Submit Assets to ALER - Specify Submission Information



2. Complete the fields in the Submission Information box, as necessary. Once submitted, the Model and Repository Project information will appear in the asset query Results pane, as described in [“The Results Pane.”](#)
  - **Model Name** – the model name by which the project will be referred in ALER Every asset that is part of the model will have that model name. The name defaults to the name of the current Eclipse project, but you can modify this name for an initial submission.

- **Model Version** – a label that is common to all assets in a model. During the initial submission and subsequent resubmission, this field displays the non-editable <Default> string and can only be edited under certain conditions, as described in [“Resubmitting an Existing Model.”](#)
- **Model Namespace** – a grouping mechanism for assembly models, with the Namespace prepended to the names of all the assets in the model in the repository. If the project you are submitting has an associated namespace, this will be displayed by default. Otherwise, the workspace preference default will be displayed, if one exists. You can modify the default Namespace name for an initial submission. The new Namespace will only be saved for the current project and will not affect the workspace name.
- **Repository Project** – the ALER project associated with the current Eclipse project. Asset usage is tracked in the repository and attributed to repository projects, which typically represent software development programs, business initiatives, etc. If the project you are submitting has an associated repository project, this will be displayed by default. Otherwise, the workspace default will be displayed, if one exists. You can modify the default Repository Project name for an initial submission. The new name will only be saved for the current project and will not affect the workspace name.
- **Comments for Registrar** – optionally provide information about the current project to the ALER Registrar. These comments will be available to the Registrar via the Asset Editor, in the Audit Log element on the Administration tab. The log entry is named “Submission Comment Added.”

**Note:** The **Model Name**, **Version**, and **Namespace** fields reflect the values used during the initial submission and cannot be edited when resubmitting a project. For more information, see [“Resubmitting an Existing Model.”](#)

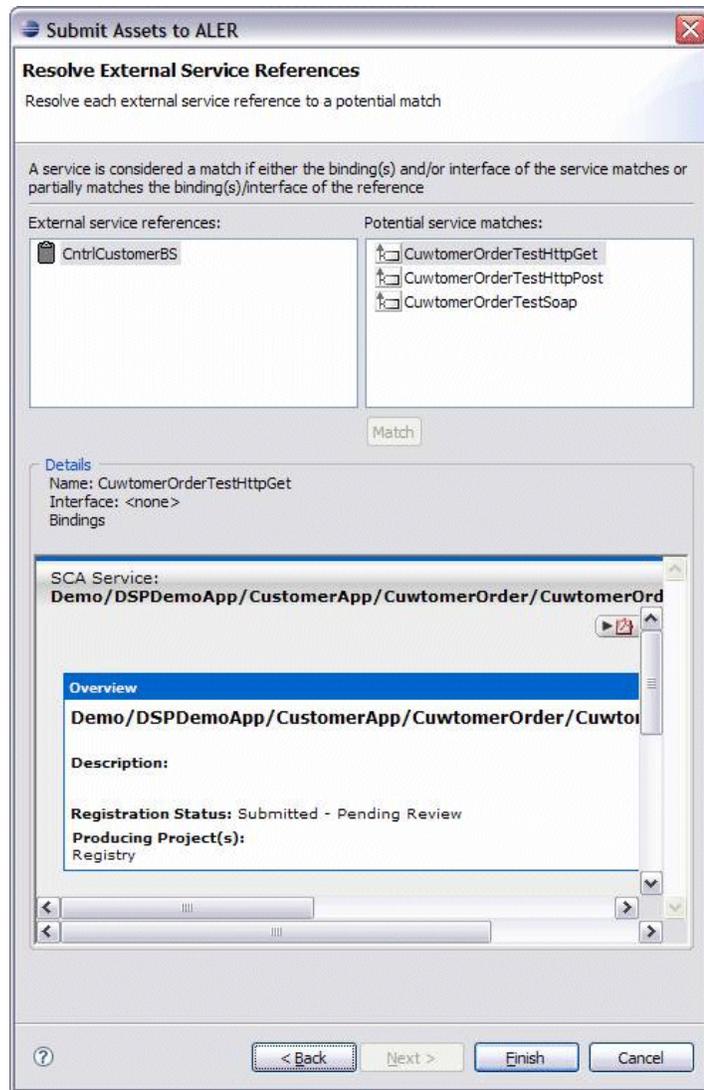
3. In the **Artifact Store** area:
  - The **Artifact Store** area displays the name of a preconfigured Artifact Store that the submitted assets will be associated with. Artifact Stores contain the files relevant to assets in ALER and are configured in the ALER console by an administrator. The **Details** box may also display some additional information about the Artifact Store.
  - In the **Project Path** box, enter the name of the Project Path that the submitted assets will be associated with. The Project Path is defined at the Eclipse project level and is relative to the **Path** attribute on the Artifact Store.
4. Click **Finish** to complete the submission.

## Resolving External Service References

If an AquaLogic project contains unresolved external service references, you may want to resolve any unresolved references before submitting the project to the Enterprise Repository; however, this is not required.

1. When submitting an AquaLogic project with unresolved external service references, the dialog box shown in [Figure 4-2](#) is automatically invoked, which provides potential matches for the unresolved references.

Figure 4-2 Submit Assets to ALER - Resolve External Service References



2. Match the unresolved external reference(s) with the potential service matches supplied by the dialog box, as follows:

- **External Service References** – Contains a list of any unresolved external service references associated with the AquaLogic project being submitted to ALER. When a service is selected, the list of Potential Service Matches is updated. This box also displays the name of the currently resolved match (if there is one) between braces { }.
- **Potential Service Matches** – Contains all the potential matching services for the selected external service reference and their percentage matching value as you hover your pointer over each match. Obviously, the higher the percentage, the better the chance of achieving a potential match. When a potential match is selected, details of the service are provided in the Details section.
- **Match** – When the Match button is selected it associates the selected potential match with the selected External Service Reference. It also updates the label of the selected external service reference by displaying the name between braces { }

3. Click **Finish** to resolve the service references.

## Guidelines for Submitting Models to ALER

The following guidelines may apply when submitting assembly models to ALER.

### Asset Naming Algorithm

In the event an asset name/version collision occurs against an existing ALER non-deleted asset during an assembly model submission, the name will be augmented with a dash and number as follows: `assetname-1`, `assetname-2`, `assetname-3`, etc. In the event name trimming needs to occur, such as where the name uses all the available characters for an asset name, the name will be trimmed at the end to make room for the name modification.

### Producing Projects

Projects submitted as assembly models will include a reference to the producing project. This project is applied to all assets in the import bundle and are additive in nature. For any assets in the bundle that already exist in ALER and have producing projects applied, the existing producing projects will remain intact and the new producing project will be added to the asset.

### No Support Imported WSDLs With Duplicate Assets

A WSDL imported using the ALER Plug-in for Eclipse cannot subsequently be imported directly using the ALER import/export tool. Any attempt to import such a WSDL using the import/export tool will fail, throwing the following “Unsupported WSDL” error message:

Error [10,001]: Unsupported WSDL - Specified WSDL can be imported only using ALSync tool as previous version of this WSDL was imported via ALSync.?

## Resubmitting an Existing Model

When initially submitting a model, the Model fields reflect the values used during the initial submission and cannot be edited when resubmitting a project. This is because they uniquely identify the assets of an assembly model, and changing any of these fields would result in a new model being created instead of an existing model getting updated.

The only time any Model fields can be modified is when an error occurs when resubmitting (e.g. a naming conflict) or when a model is *locked*, which will require entering a new model name, version, and namespace. A model can become *locked* under two conditions:

- A user changes the registration status of one or more assets in the model using ALER's Web-based console.
- If a different user resubmits a model that someone else originally submitted.

If either of these conditions is detected by ALER when a model is being resubmitted, a message will be displayed and the user can change any or all of the following fields: Model Namespace, Model Name, or Model Version. The user can then continue submit the model, but that model will now be a new model, and will not overwrite the original model (since that model is locked).

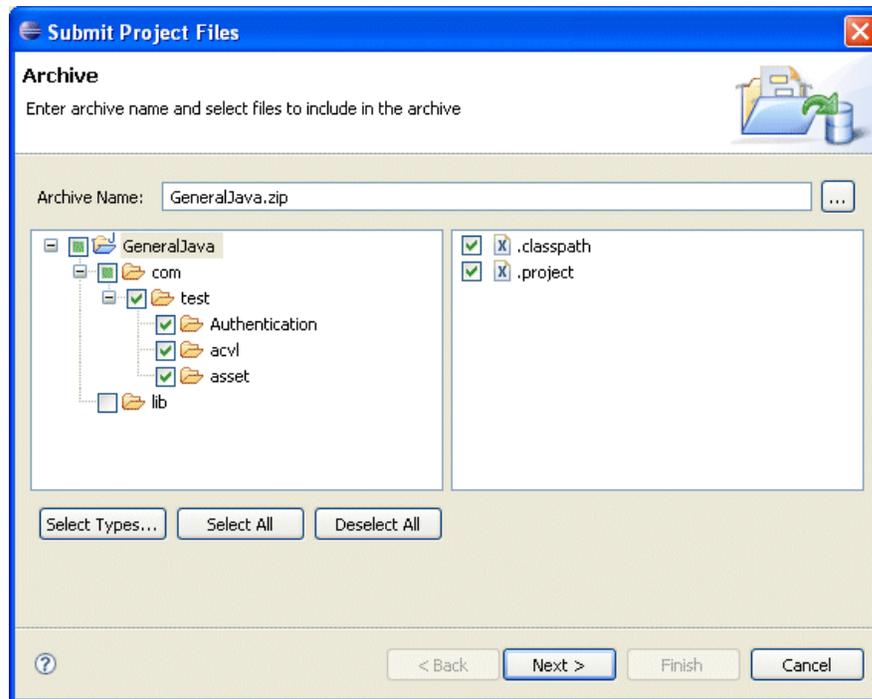
## Submitting Eclipse Project Files to ALER

When you right-click an Eclipse project and select the Submit to AquaLogic Enterprise Repository from the shortcut menu, WorkSpace Studio plug-ins first determine if connectivity has been established to ALER. If a connection is not yet established, the Connect to Enterprise Repository wizard is launched to establish connectivity.

Once connectivity is established, the ALER plug-in determines whether the associated project is an Assembly Model (see [“Submitting Assembly Models to the Repository.”](#)) or an existing ALER API project. When it is an ALER project, the **Submit Project Files** wizard allows you to select and update an existing archive with assets to submit to ALER.

1. Right-click an Eclipse project and select **Submit to AquaLogic Enterprise Repository** from the shortcut menu.
2. In the Archive Name field, enter the path to an existing project archive field or browse to an archive using the **Ellipses** button. You can also create a new archive file. When an existing file is selected, its fully qualified path is placed in the Archive Name text field. A valid project archive must have a `.zip` file extension.
3. When selecting an existing archive, click **OK** to confirm that it is okay to overwrite the selected project archive.

Figure 4-3 Submit Project Files - Select Archive Name



4. Use the resulting project folder structure to select at least one file from the project to submit to ALER.
5. If necessary, you can click the **Select Types** button to open a dialog box where can select certain file types to include in the archive.
6. After selecting the files you want to include in the archive, click **Next**. All artifacts selected from the project will be zipped into the archive file.
7. After the archive and its contents have been specified, you can enter asset submission data, such as a version number, the type of asset to be submitted, a description, and associated comments.

Figure 4-4 Submit Project Files - Enter Submit Data

The screenshot shows a Windows-style dialog box titled "Submit Project Files" with a close button in the top right corner. The main area is titled "Submit Data" and contains the instruction "Enter asset submission data". There is a folder and pencil icon in the top right of the form area. The form fields are: "Asset Name" with the text "Deposit Service"; "Asset Version" with the text "2.1"; "Asset Type" with a dropdown menu showing "Application"; "Description" with the text "Online Web Service"; and "Comments" with an empty text area. At the bottom, there is a help icon (question mark) on the left and four buttons: "< Back", "Next >", "Finish", and "Cancel".

8. Click **Finish**.

9. Click OK on the confirmation window to complete the submission process.

The project will appear as an asset in the **Submitted - Pending Review** folder in the ALER Asset Editor file tree. You can also refresh the data in the Enterprise Repository View's Results pane to view the asset in Pending Review status. You can also click the asset to view more details. For more information, see ["The Results Pane."](#)

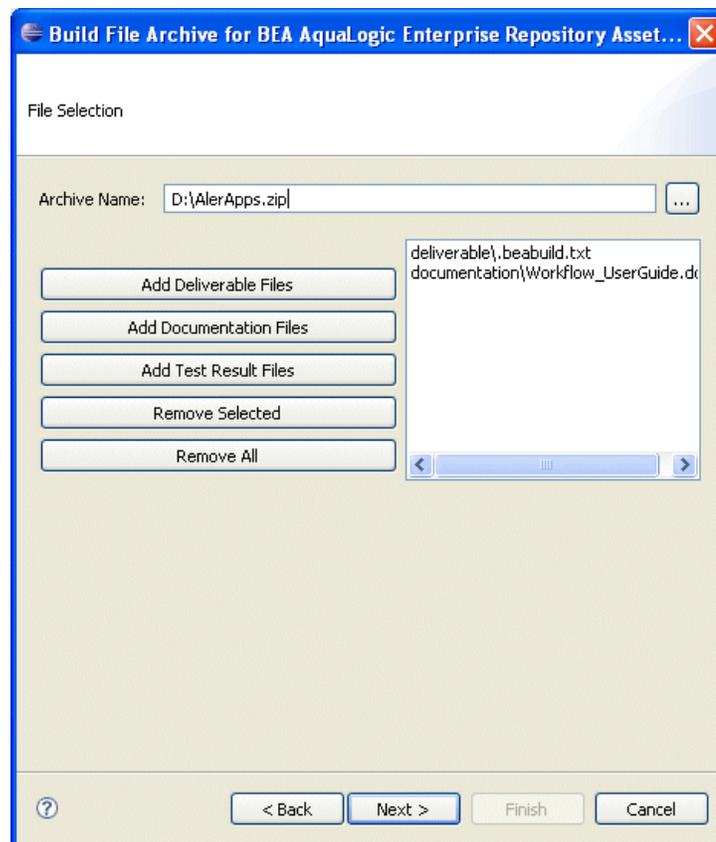
## Submitting Archived Assets

The Archive Submission wizard maintains support for legacy functionality for ALER 2.x or earlier releases. This allows you to submit single and/or compound-payload assets to AquaLogic Enterprise Repository by creating an asset archive ZIP file.

To build an archive to submit to ALER:

1. Select the project that contains assets that you want to create an archive for.
2. Open the Export dialog box, by either:
  - Right-click the desired project and select **Export** on the context menu.
  - Select **Export** from the main File menu.
3. In the Export dialog box, open the **AquaLogic Enterprise Repository** folder and select the **Build Archive Submission** option.

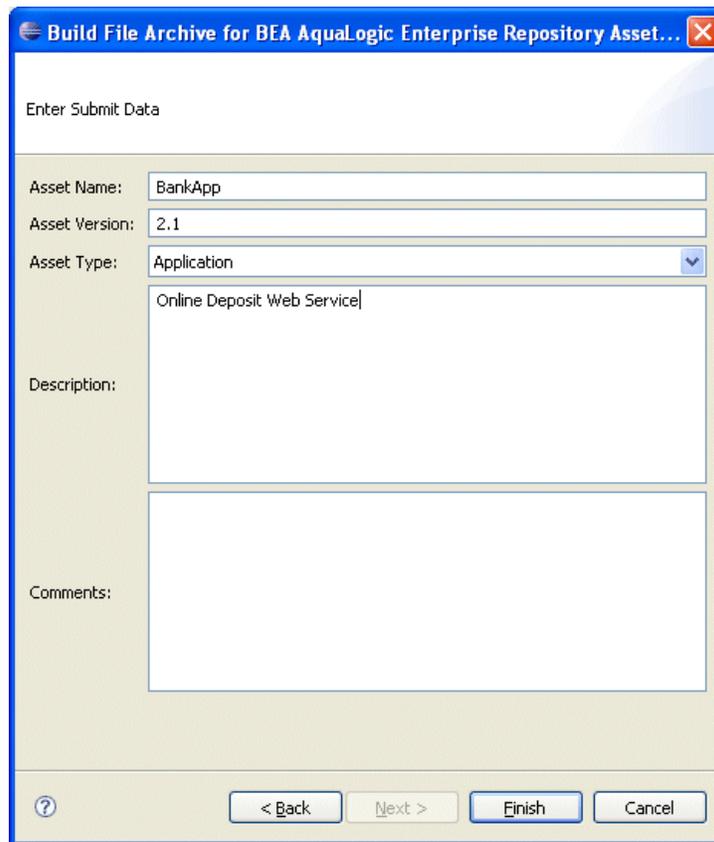
**Figure 4-5 Build File Archive - File Selection**



## Submitting Assets to ALER

4. Complete the Build File Archive - File Selection dialog box as follows:
  - a. Enter an archive name by browsing to select an existing archive on your local file store, or create a new archive. In either case the archive must have a .zip extension.
  - b. Use the file category buttons to select and assign files to the archive file.
  - c. Click **Next**.

**Figure 4-6 Build File Archive - Enter Submit Data**



The screenshot shows a dialog box titled "Build File Archive for BEA Aqualogic Enterprise Repository Asset...". The dialog box has a blue title bar and a white background. The main area is titled "Enter Submit Data". It contains several input fields and a dropdown menu:

- Asset Name:** A text box containing "BankApp".
- Asset Version:** A text box containing "2.1".
- Asset Type:** A dropdown menu with "Application" selected.
- Description:** A text area containing "Online Deposit Web Service".
- Comments:** An empty text area.

At the bottom of the dialog box, there are four buttons: a help button (question mark icon), "< Back", "Next >", "Finish", and "Cancel".

5. On the Enter Submit Data dialog box, fill in the appropriate information describing the asset.
6. Click **Finish**.

7. Click **OK** to confirm the submission to ALER.

The asset will upload to the installed registry, and will appear in the Submitted - Pending Review folder in the file tree in AquaLogic Enterprise Repository's Asset Editor. You can also refresh the data in the Enterprise Repository View's Results pane to view the asset in Pending Review status. You can also click the asset to view more details. For more information, see "[The Results Pane.](#)"

Submitting Assets to ALER

# Setting Eclipse Preferences for ALER

This section contains information on the following subjects:

- [“Overview”](#)
- [“Configuring the ALER Connection”](#)
- [“Configuring Automated Usage Detection”](#)

## Overview

A number of ALER workspace preferences and project properties are available for workspace connection configuration and automatic usage detection. These are accessible using the standard Eclipse workspace preferences, using the **Window > Preferences** menu.

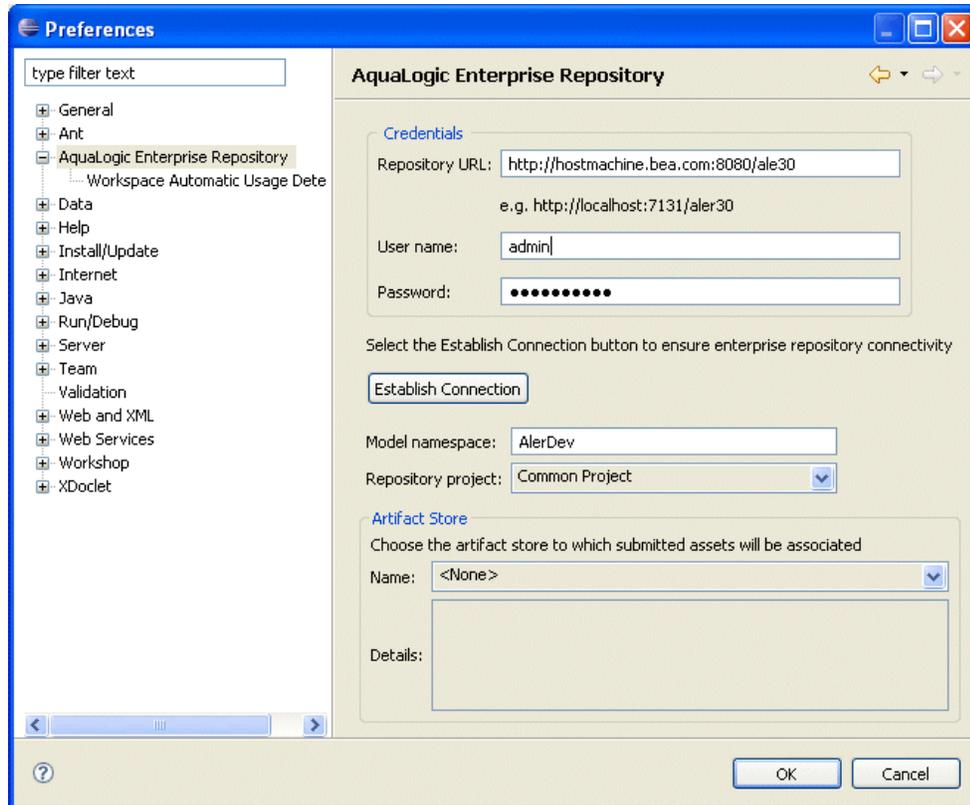
For an Eclipse project, you can right-click the project and select **Properties** from the menu.

## Configuring the ALER Connection

If you haven't already entered connection credentials using the Connect to Enterprise Repository wizard, which is automatically invoke when querying or submitting assets before repository connectivity has been established, then you can manually set ALER connection credentials for an Eclipse workspace as follows:

1. On the Window menu, click **Preferences**.
2. Select **AquaLogic Enterprise Repository**.

Figure 5-1 Preferences - Connection Credentials



3. In the Credentials area, enter the server location and login credentials, as follows:
  - **Repository URL** – the URL of the repository server. The URL must include the host, port, and ALER server name. For example, `http://localhost:7001/aler30`.
  - **User Name** – user name to gain access to the repository.
  - **Password** – password to gain access to the repository.
4. Click the **Establish Connection** button to ensure enterprise repository connectivity.  
If a connection cannot be established, an appropriate error message will be displayed.
5. Once connectivity is established, you can specify your workspace preferences:

## Setting Eclipse Preferences for ALER

- Enter a **Model Namespace** to use as a default for your all of your projects. The Namespace provides a means to organize your models, with the Namespace pre-pended to the names of all the assets in the model in the repository. However, you can change the Namespace on a project-by-project basis (such as when submitting assets), and the new Namespace will only be saved for that project, but will not affect the Workspace Preference name.
  - Select a **Repository project** in ALER that the submitted model will be associated with. Asset usage is tracked in the repository and attributed to repository projects, which typically represent software development programs, business initiatives, etc.
6. The **Artifact Store** area displays the name of a preconfigured Artifact Store that the submitted assets will be associated with. Artifact Stores contain the files relevant to assets in ALER and are configured in the ALER console by an administrator. The **Details** box may also display some additional information about the Artifact Store.
  7. Click **OK** to finish.

## Configuring Automated Usage Detection

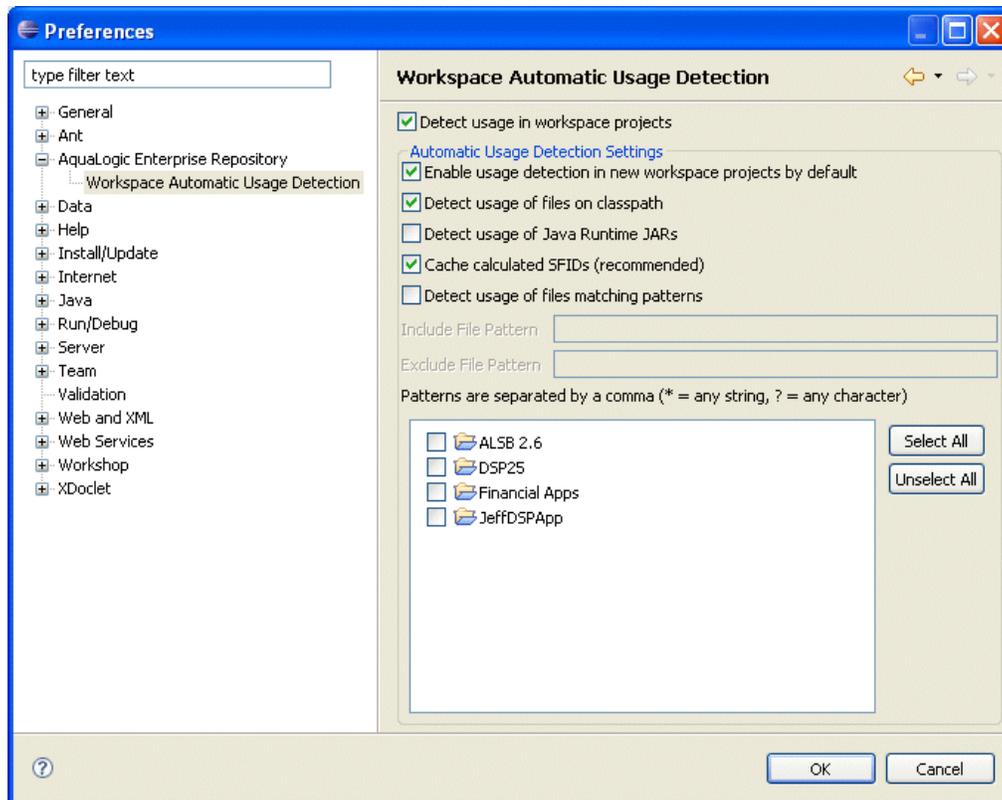
ALER can automatically detect asset reuse within the development environment. This allows development teams to ensure that they get asset reuse credit, regardless of whether the assets have been downloaded through ALER. Automated Usage Detection relies on a fingerprinting process, called *Software File Identification* (SFID), which tags selected files within an asset with a unique ID. This SFID is then used to detect when and where an asset is used, even if the asset was acquired through means other than the ALER Use - Download process. An instance of usage is recorded by ALER when tagged files within the asset are brought into the developer's IDE, and a new build or build clean occurs.

See the *ALER Software File Identification Guide* for more information.

**Note:** Automated Usage Detection requires the installation of the ALER Plug-in for WorkSpace Studio, and is currently compatible only with Eclipse and Eclipse-based IDEs.

1. On the Window menu, click **Preferences**.
2. Select **AquaLogic Enterprise Repository**.
3. Select **Workspace Automatic Usage Detection**.

Figure 5-2 Preferences - Workspace Automatic Usage Detection



4. Click the **Detect usage in workspace projects** check box, and then activate the desired usage detection features, as appropriate:
  - Enable usage detection in new workspace projects by default – monitors new projects
  - Detect usage of files on classpath – monitors files on classpath.
  - Detect usage of Java Runtime JARs – monitors Java Runtime JARs
  - Cache calculated SFIDs (recommended) – caches calculated SFIDs (enhances performance)
  - Detect usage of files matching pattern – monitors files matching specified patterns
5. Enter the appropriate information in the **File Pattern** text boxes:

## Configuring Automated Usage Detection

- Include File Pattern – Includes indicated file pattern
  - Exclude File Pattern – Excludes the indicated file pattern
6. Specify which project directories will be targets for automatic usage detection by using the individual check boxes or by using the **Select All** and/or **Unselect All** buttons.
  7. Click **OK** when finished.

## Setting Eclipse Preferences for ALER