Service Assembly Modeler (SAM) User Guide

BEA Service Assembly Modeler (SAM) is an Eclipse plug-in that provides tools for working with SCA composite service assembly models. SAM is bundled with the AquaLogic Enterprise Repository (ALER) and other BEA AquaLogic products.

Use SAM to:

- SCA-enable existing AquaLogic applications and thereby create service assembly models based on those applications.
- View SCA composites and other service assembly model artifacts.
- Submit models to the AquaLogic Enterprise Repository (ALER).

These topics provide an introduction to SAM.

Introduction to Service Assembly Modeler (SAM)
: Provides an overview of SAM

Introduction to Service Component Architecture (SCA)
: Provides an overview of SCA, upon with SAM is based.

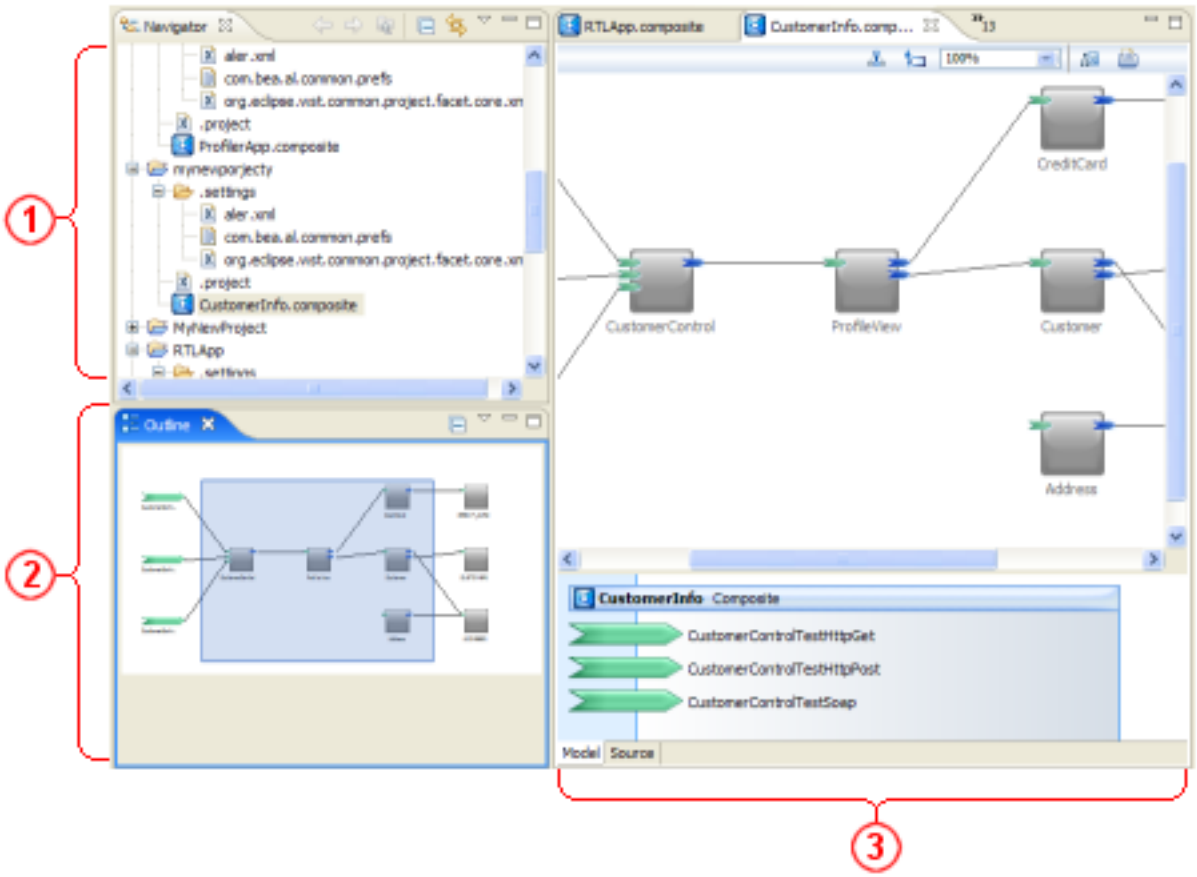
Introduction to Service Assembly Modeler (SAM)

The BEA Service Assembly Modeler (SAM) is an Eclipse plug-in that provides tools for working with SCA composite service assembly models. SAM is bundled with the AquaLogic Enterprise Repository (ALER) and will also be included with other BEA AquaLogic products.

Use SAM to:

- SCA-enable existing AquaLogic applications and thereby create service assembly models based on those applications. See Create SAM Projects
- View SCA composites and other service assembly model artifacts. See Open and View SCA Composites in SAM and View and Edit Service Assembly Assets.
- Submit models to the AquaLogic Enterprise Repository (ALER). See Submit Assets to Enterprise Repository Wizard

SAM includes three main views, as shown in the illustration below. (The organization of the views may differ, depending on how you arrange them in the Eclipse IDE.)



Key:

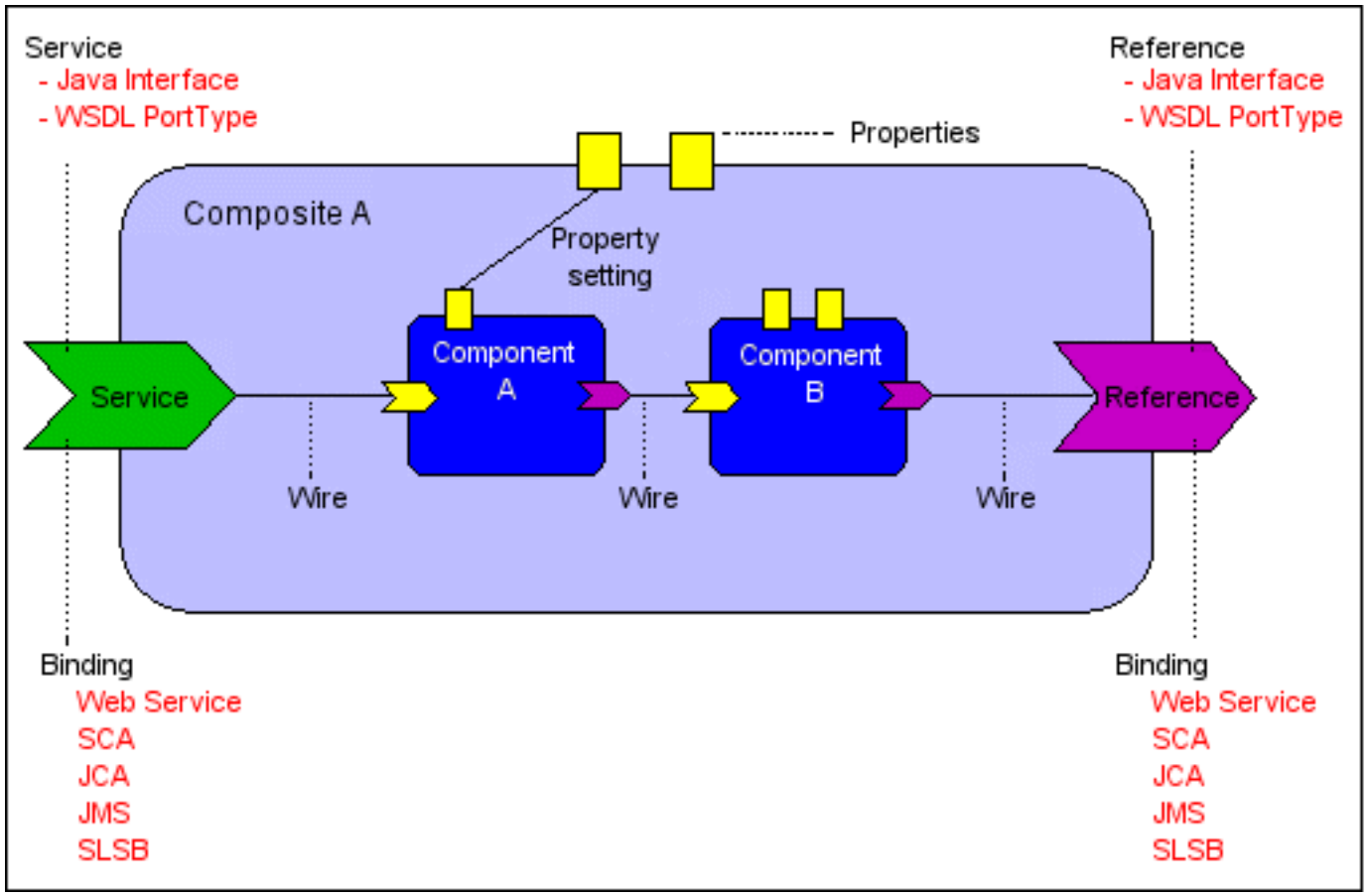
1. Navigator View

2. Outline View
3. SAM Composite View

Follow the above links for more information on each view.

Introduction to Service Component Architecture (SCA)

SAM is based on Service Component Architecture (SCA). SCA provides a programming model for building applications and solutions based on a Service Oriented Architecture. SCA is based on the idea that business functionality can be composed from a series of services, which are assembled together to create solutions that serve a particular business need. These solutions are called composite applications, and they are visually represented by *assembly models*. Composite applications can contain both new services created specifically for the application and existing services from existing systems, applications, and business function, reused as part of the composition.The following figure provides an example of an assembly model for a composite application.



The SCA programming model is a standard from OMG. SCA standardizes the information exchanged between different tools, ensuring that services and components that comprise an application are consistently modeled, represented, and described. Consistency means that services built in different tools, using different programming paradigms can be easily reused and quickly deployed, helping organizations realize the promised efficiencies of SOA.

BEA is leveraging the SCA standard to facilitate information exchange among BEA tooling, including ALSB, ALDSP, ALBPM, WLS, WLP, WLI, and Tuxedo – SALT. BEA will also leverage SCA to facilitate information exchange with third party tooling.

This means that services from data service platforms, a service bus, business process modeling tools, etc. can be captured and represented in a common format, and then easily reused to compose other business applications.

BEA's Service Assembly Modeler (SAM) 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 reusing services that already exist, or by creating new services. SAM allows developers to view the assembly model for composite applications, as well as the XML definition of the composite.

The AquaLogic Enterprise Repository (ALER) allows organizations to manage composite applications and reusable services. Development teams can submit their composite applications to ALER directly from SAM. ALER provides visibility into the existing applications and services, traceability between composite applications and their dependencies, and analytics that measure reuse and facilitate impact analysis.

Service Assembly Modeler (SAM) Tasks

These topics explain how to accomplish tasks in Service Assembly Modeler (SAM).

Open SAM
: Describes how to open SAM in Eclipse.

Create SAM Projects
: Describes how to create a SAM Project.

View Service Assembly Assets in SAM
: Describes how to view configuration files and other SCA assets.

Open and View SCA Composites in SAM
: Describes how to open SCA composites and view them in the SAM Composite View.

Use Graph Tracing in the SAM Composite View
: Describes how to use the graph tracing feature to highlight dependencies in an SCA composite.

Submit Assembly Models to AquaLogic Enterprise Repository
: Describes how to submit projects to ALER.

Create SAM Projects

Use the SAM Project Wizard to create a new SAM project. This topic describes the first steps in creating any SAM project.

1. Select **File > New > Other...**
2. From the **Select a Wizard** dialog, select **AquaLogic > Service Assembly Modeler Project** to launch the SAM Project Wizard.
3. Follow the instructions in the wizard to create the project. Different pages are displayed, based on the options you select:
   - SAM Project Wizard - New SAM Project - This is the first page. You always start here.
   - SAM Project Wizard - ALDSP 2.5 - For creating SAM projects based on AquaLogic Data Service Platform 2.5.
   - SAM Project Wizard - ALSB 2.6 - For creating SAM projects based on AquaLogic Service Bus 2.6.
   - SAM Project Wizard - Enterprise Repository Project Properties - For connecting to an enterprise repository where you can submit models.

Note: To view ALDSP 3.0 projects in the **SAM Composite View**, right-click the ALDSP 3.0 project and select **Show in Service Assembly Modeler**.

Open and View SCA Composites in SAM

Method 1

1. If it is not already open, open the **Navigator** view by choosing **Window** > **Show View** > **Navigator**.
2. Find the composite file in the **Navigator** view. A composite file is identified by the SCA composite  icon and the `.composite` extension.
3. Double-click the file name or right-click the name and select **Open** to open the composite in the SAM Composite View. See SAM Composite View.

Method 2

1. If it is not already open, open the Navigator view by choosing **Window** > **Show View** > **Navigator**.
2. If the project is SCA-enabled, right-click the project in the **Navigator**view and select **Launch Service Assembly Model (SAM) editor**.

Open SAM

Do either of the following:

- Start Eclipse in any workspace and create a SAM project. See Create SAM Projects.
- Start Eclipse, and when prompted to select a workspace in the **Workspace Launcher** dialog, select a project containing SAM projects.

Submit Assembly Models to AquaLogic Enterprise Repository

You can submit SAM and ALER projects to the AquaLogic Enterprise Repository.

1. If it is not already open, open the Navigator View by choosing **Window** > **Show View** > **Navigator**.
2. In the Navigator, right-click a project and select **Submit to AquaLogic Enterprise Repository** from the menu. See Submit Assets to Enterprise Repository Wizard.

Use Graph Tracing in the SAM Composite View

Graph tracing is a feature in the SAM Composite View that makes it easy to see dependencies in an SCA composite.

1. Open the composite in SAM. See Open and View SCA Composites in SAM.
2. Make sure the composite is displayed in the **Model View** tab. See Sam Composite View
3. If graph tracing is not currently on, click  **Enable Graph Tracing** in the toolbar. This button is a toggle, which turns graph tracing on or off. When on and no items are selected, all the items displayed in the Overview pane are dimmed.
4. In the Overview pane, click the service, reference, or component whose dependencies you want to see. You can **Ctrl**-click to select multiple items.

   All items associated with the selected item are highlighted while all other items are dimmed. (That is, selected items are displayed normally against the dimmed background.) For example, if you select a component, the component and all its associated services and references are highlighted. If you select a service, the service and all its associated components are highlighted.

View Service Assembly Assets in SAM

1. If it is not already open, open the Navigator View by choosing **Window** > **Show View** > **Navigator**.
2. Find the file in the **Navigator** view.
3. Open the file using any of the following:
    a. Double-click the file name to open the file in the default editor for that file type.
    b. Right-click the file name and select **Open** to open the file in the default editor for that file type.
    c. Right-click the file name and select **Open With > *Name* Editor** (where **Name** is the name of an editor) to open the file in the specified editor. When you open a file that can be displayed as text or in a structure (such as an XML file), you can toggle between the two views by clicking the **Design** and **Source** tabs at the bottom of the editor.

Service Assembly Modeler (SAM) User Interface Reference

These topics describe the views, dialogs, editors, and other Service Assembly Modeler (SAM) user interface components.

AquaLogic Enterprise Repository Preferences Dialog
: Describes the dialog used for manually setting ALER connection credentials for an Eclipse workspace.

Build File Archive for ALER Assets Wizard
: Describes the wizard used for submitting legacy ALER assets to AquaLogic Enterprise Repository

Download Artifacts Wizard
: Describes the wizard used for downloading an asset's artifacts (i.e., payload) into an Eclipse project.

Enterprise Repository Access View
: Describes the view used for viewing and querying assets associated with a project.

Enterprise Repository Asset Relationships View
: Describes the view used for reviewing relationships for a selected asset.

Establish Enterprise Repository Connection Dialog
: Describes the dialog used for connecting to an Enterprise Repository.

Sam Composite View
: Describes the view used for viewing SAM composites, as graphics and as source.

Navigator View in SAM
: Describes the view that provides a hierarchical display of SAM projects and their contents.

Outline View in SAM
: Describes the view that provides an overview of a composite displayed in the SAM Composite View or provides a hierarchical view of any structured document displayed as source.

SAM Project Wizard - New SAM project
: Describes the first page in the wizard used for creating SAM projects.

SAM Project Wizard - AquaLogic Data Service Platform 2.5
: Describes the page in the SAM Project Wizard used for creating SAM projects based on ALDSP 2.5 resources.

SAM Project Wizard - Enterprise Repository Project Properties
: Describes the page in the SAM Project Wizard used for setting enterprise repository properties for this project and to establish a connection to the repository.

SAM Project Wizard - AquaLogic Service Bus 2.6
: Describes the page in the SAM Project Wizard used for creating SAM projects based on ALSB

2.6 resources.

Submit Assembly Models to Enterprise Repository Wizard

Describes the wizard used for submitting assembly models to an enterprise repository.

Submit Project Files to ALER Wizard

Describes the wizard used for selecting and updating an existing archive with assets to submit to ALER.

Workspace Automatic Usage Detection Preferences Dialog

Describes the dialog used for setting asset usage tracking preferences.

Service Consumption Dialog

Consume services from an AquaLogic Enterprise Repository, a UDDI registry, a URI, and other resource types.

AquaLogic Enterprise Repository Preferences Dialog

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

1. In the **Credentials** area, enter the server location and login credentials:
    - **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.
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 specify your workspace preferences:
    - Enter a **Model Namespace** 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 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.
4. 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.
5. Click **OK** when finished.

For detailed instructions on setting ALER workspace preferences, see the *Repository Integration with Eclipse Using WorkSpace Studio*.

Submit Assets to Enterprise Repository Wizard

When you right-click an Eclipse project and select **Submit to AquaLogic Enterprise Repository** from the menu, the repository access plug-in determines whether or not the project can be transformed into an assembly model. If it can be transformed into an assembly model, the **Submit Assets to Enterprise Repository** wizard enables you to submit the associated assets. The first dialog box captures metadata associated with the submitted asset, such as model and Artifact Store information. If necessary, a second dialog box provides a means to resolve external service references that may exist for an assembly model before submitting the assets.

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.

For detailed instructions on submitting assets to the Enterprise Repository, see the *Repository Integration with Eclipse Using WorkSpace Studio*.

Specify Submission Information

1. Right-click an Eclipse project and select **Submit to AquaLogic Enterprise Repository** from the menu.
2. Complete the fields in the Submission Information box, as necessary. Once submitted, this information will appear in the asset query 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 string and can only be edited under certain conditions.
    - **Model Namespace** – a grouping mechanism for assembly models, with the Namespace pre-pended 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, in 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**, **Namespace**, and **Version** fields reflect the values used during the initial submission and cannot be edited when resubmitting a project -- unless a model lock or name conflict is encountered. See "Guidelines for Submitting Models to ALER" in the *Repository Integration with Eclipse Using WorkSpace Studio*.

3. Review and complete the fields 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**.

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 **Resolving External Service References** dialog box is automatically invoked, which provides potential matches for the unresolved 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. 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.

Workspace Automatic Usage Detection Preferences Dialog

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.

1. 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
2. Enter the appropriate information in the **File Pattern** text boxes:
   - Include File Pattern – Includes indicated file pattern
   - Exclude File Pattern – Excludes the indicated file pattern
3. 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.
4. Click **OK** when finished.

For more information on setting ALER workspace preferences, see the *Repository Integration with Eclipse Using WorkSpace Studio*.

Build File Archive for ALER Assets Wizard

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

For more information on creating asset archive files, see the *Repository Integration with Eclipse Using WorkSpace Studio*.

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.
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**.
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 ALER'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.

Download Artifacts Wizard

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.
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.
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).

For detailed instructions on downloading asset artifacts, see the *Repository Integration with Eclipse Using WorkSpace Studio*.

Enterprise Repository Access View

The Enterprise Repository Access View view provides access to assets and artifacts in the Enterprise Repository. This view supports assets that conform to the SCA standard, in addition to assets that already exist in your ALER instance. 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.

The Enterprise Repository Access view is displayed as a tabbed pane containing two tabs. The Search tab enables querying of assets and displays results based upon specified criteria. The Project Team Assets tab allows you to view all assets associated with a specified ALER project.

For detailed instructions on using the Enterprise Repository Access View, see the *Repository Integration with Eclipse Using WorkSpace Studio*.

Search Tab

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.

Query Pane

The Query pane allow 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 2.6 or an AquaLogic DSP 2.5 project, and common ALER assets already defined in your ALER instance.

- You can filter queries by keyword, asset type, registration status, or by any combination of these filters. Three important asset types to note are:
  - **Assembly Model Assets** – assets that are submitted from an integrated external endpoint, such as AquaLogic DSP, version 2.6 and AquaLogic Service Bus, version 2.5.
  - **Common Assets** – assets from an existing ALER project.
  - **Consumable Assets**
- You can also filter your queries using the asset Categorizations checkboxes. Note that if more than one categorization type is used in the criteria, the selected types are AND'ed together, and that there is no OR option.
- 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.
- You can refresh the query results by using the **Refresh enterprise repository information** toolbar button, which queries the enterprise repository for its metadata, and repopulates the appropriate fields based upon the results of the query.

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, as follows:

- **Asset** – The simple name of the asset.
- **Model** – The name for Assembly Model assets imported into ALER.
- **Model Namespace** – The Namespace is 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 **Download artifacts** 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.

There are also toolbar buttons that allow you to toggle the asset views:

- **Toggle the display of the asset detail view** – Displays a selected asset's details in a web browser view, or to not launch the web-based view when an asset is selected.
- **Display the asset properties and relationships view** – Allows you to view the details of a selected asset in the Results pane.

For more information, see Enterprise Repository Asset Relationships View.

You can also 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 the Eclipse Integration Guide.
- **Subscribe/Unsubscribe** – subscribes to the selected asset if currently unsubscribed, and visa versa.
- **Show in asset details view** – opens the Asset Details view.

Project Team Assets Tab

The Project Team Assets view allows you to view a list of assets appearing in all of the Compliance

Templates assigned to your project. The viewer indicates 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.

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.

Enterprise Repository Asset Relationships View

The Enterprise Repository Asset Relationships view displays the relationships for a selected asset in a graphical format.

The Asset Relationship tab's content is based upon an asset selected 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.

Note: There are specific *system* relationships that are present for Assembly Models only.

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.

For detailed instructions on using the Enterprise Repository Asset Relationships View, see the *Repository Integration with Eclipse Using WorkSpace Studio*.

Establish Enterprise Repository Connection Dialog

When you invoke an action on an enterprise repository, such as querying or submitting 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 gesture, such as attempting to view asset details.

1. In the **Credentials** area, enter the server location and login credentials:
    - **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.
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 **Next** to select your workspace preferences (skip to Step 4).
    - Click **Finish** to exit.
4. 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 and is 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.
    - 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.
5. Click **Finish** to exit.

For detailed instructions on establishing ALER connectivity, see the *Repository Integration with Eclipse Using WorkSpace Studio*.

SAM Project Wizard - New SAM project

The SAM Project Wizard displays a series of pages with options for creating a SAM project. The New SAM Project page is the first page in the wizard. Complete this page as follows:

1. In the **Project Name** field, enter a name for the project.
2. Under **Project Contents**, set the location for storing project items such as configuration files, composite files, etc., as follows:
   - Select **Use Defaults** to use the default folder for the current Eclipse workspace.
   - Deselect **Use Defaults** and enter the location or click **Browse** to search for one.
3. To create a SAM project based on an existing Aqualogic application,
   a. From the drop-down list under **Project Source**, select an AquaLogic product type from which to create the SAM project. This list displays product types supported by the current version of SAM. More types will be added in future releases of SAM.
   b. Click **Next**. The next page of the wizard is displayed. Depending on the type you chose, see the following to continue:
      - SAM Project Wizard - AquaLogic Service Bus 2.6
      - SAM Project Wizard - AquaLogic Data Service Platform 2.5

SAM Project Wizard - AquaLogic Data Service Platform

Use the AquaLogic Data Service Platform page of the SAM Project Wizard to select the AquaLogic Data Service Platform (ALDSP) application to SCA-enable. This page appears when you select **AquaLogic Data Service Platform 2.5** under **Project Source** on the SAM Project Wizard - New SAM Project page.

1. For the **ALDSP Project Root Directory** field, click **Browse** to select a directory containing one or more ALDSP projects.
2. Under **Select the Project to SCA-enable**, click a project to select it.
3. Do either of the following:
    a. Click **Next** to set enterprise repository project properties and to establish a connection to the repository. See SAM Project Wizard - Enterprise Repository Project Properties.
    b. Click **Finish** to create the project without setting repository project properties. The new project is displayed in the Navigator View.

Note: To view ALDSP 3.0 projects in the **SAM Composite View**, right-click the ALDSP 3.0 project and select **Show in Service Assembly Modeler**.

SAM Project Wizard - Enterprise Repository Project Properties

Use this page of the SAM Project Wizard to set enterprise repository properties for this projectand to establish a connection to the repository. You can skip this step by clicking **Finish** at the bottom of the page before trying to connect to the repository.

1. If you have not already established a connection with an enterprise repository, click **Connect to Repository** to establish the connection. When a connection is made, the button is disabled until you enter valid credentials. Once a connection is established, the button is disabled until you change credentials. Under **Enterprise Repository Property Settings**, accept or override the default repository project settings, as follows:
    - Select **Use workspace defaults** to accept the defaults for the model namespace and the repository project.
    - Deselect **Use workspace defaults** to override the defaults. Then:
        - In the **Model Namespace** field, enter a name to be used as the namespace for the model, if it does not already have one defined.
        - From the **Repository project** list, select an ALER project for the model you are submitting. Usage of assets is tracked in the repository and attributed to repository projects. Repository projects typically represent software development programs, business initiatives, etc."
2. When a connection is made, the Enterprise Repository Connection dialog is displayed. Enter your credentials, then click **Next**.
3. Under **Artifact Store**, do the following:
    - **Name** - Select the name of a preconfigured Artifact Store with which the submitted assets will be associated. This is where the files relevant to assets in ALER are stored.
    - **Details** - Optionally provide additional information about the Artifact Store.
    - **Project Path** - Enter the project path with which the submitted assets will be associated.
4. Click **Finish**. The new project is displayed in the Navigator View.

SAM Project Wizard - AquaLogic Service Bus

Use the AquaLogic Service Bus page of the SAM Project Wizard to select the AquaLogic Service Bus (ALSB) application to SCA-enable. This page appears when you select **AquaLogic Service Bus 2.6** under **Project Source** on the SAM Project Wizard - New SAM Project page.

1. In the **Server Host** field, enter the address of the server hosting the application, for example, example.host.com.
2. In the **Server Port** field, enter the port used for the host.
3. In the **Server User** field, enter a valid user ID.
4. In the **Server Password** field, enter the password for the user.
5. Click the **Get Application Composite** button to retrieve the ALSB applications available at the given location.
6. Under **Select the application composite to SCA-enable**, select an application to enable.
7. Do either of the following:
    a. Click **Next** to set enterprise repository project properties and to establish a connection to the repository. See SAM Project Wizard - Enterprise Repository Project Properties.
    b. Click **Finish** to create the project without setting repository project properties. The new project is displayed in the Navigator View.

Patch Profile Dialog

This dialog is displayed when a you open the SAM or ALER plug-ins and new patch profile is detected. Select the appropriate profile from the list and click **OK**. The profile is then prepended to the relevant classpath.

Submit Project Files to Enterprise Repository Wizard

When you right-click an Eclipse project and select **Submit to AquaLogic Enterprise Repository** from the menu, if connectivity is established, WorkSpace Studio's ALER plug-in determines whether the project is an Assembly Model 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.

For more information on submitting ALER project files, see the *Repository Integration with Eclipse Using WorkSpace Studio*.

1. 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.
2. When selecting an existing archive, click **OK** to confirm that it is okay to overwrite the selected project archive.
3. Use the resulting project folder structure to select at least one file from the project to submit to ALER.
4. 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.
5. 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.
6. 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.
7. Click **Finish**.
8. 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.

Navigator View in SAM

The Navigator View provides a hierarchical view of the service assembly models in the current workspace. Models are represented in Eclipse as SAM projects containing SCA composites and other assets that configure the model.

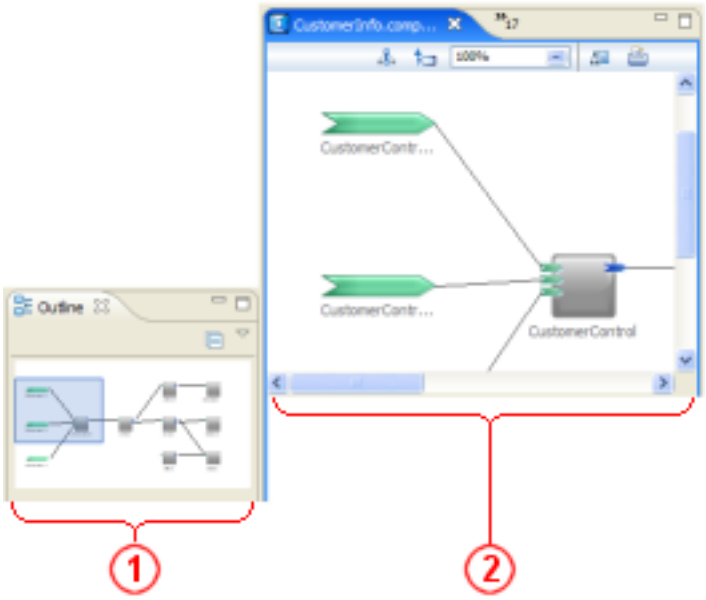
To open a file in its default view or editor, double-click the file name, or right-click and select **Open** from the menu. To open in a different view or editor, right-click and select **Open With** from the menu. Then select the view or editor to use.

To submit an assembly model to AquaLogic Enterprise Repository, right click the name of the SAM project and select **Submit to AquaLogic Enterprise Repository**. See See Submit Assets to Enterprise Repository Wizard.

Outline View in SAM

When an SCA composite is open in the **Model View** tab of the SAM Composite View, the Outline View provides an overview of the entire composite. If the composite is so big that it does not fit into the SAM Composite View, the Outline View superimposes a shaded box over the part that is visible. Drag the box in the Outline View to display any hidden parts of the composite in the Composite View.

For example, in the following illustration, the blue box in the Outline View (1), shows what portion of the SCA composite is visible in the SAM Composite View (2). Dragging the blue box on the left changes what portion of the composite is shown on the right.



Key:

1. Outline View
2. SAM Composite View, **Model View** tab

SAM Composite View

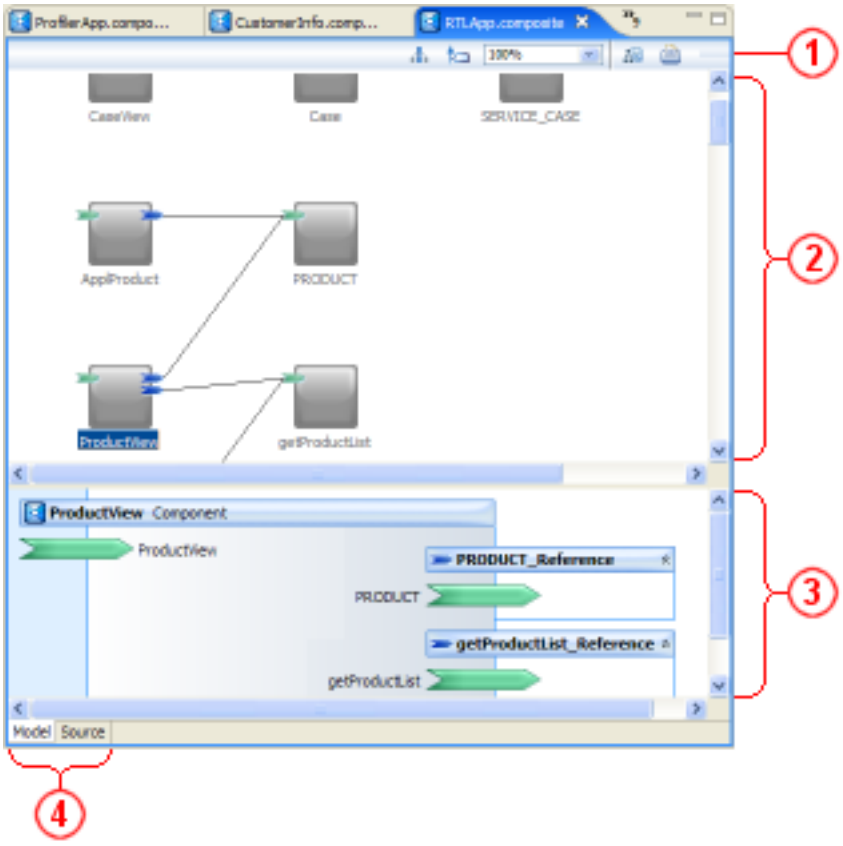
The SAM Composite View contains two tabs (whose labels are displayed at the bottom of the SAM Composite View page):

- The **Model View** tab displays a graphical representation of a SCA composite. See Model View, below.
- The **Source View** tab displays the composite's descriptor.

Both tabs are read-only. If you edit the XML outside of SAM, the composite cannot be displayed graphically in the SAM Composite View.

Model View

The **Model View** tab of the SAM Composite View provides a graphic view of the selected composite. It contains a toolbar and two panes:



Key to the callouts in the above graphic:

1. Toolbar
2. Overview Pane

3. Detail View Pane

4. **Model View** and **Source View** tabs

Toolbar

The toolbar is always available, regardless of which view has focus. It contains the following buttons.

Toolbar Item	Description
	**Graph Tracing** - Toggles the graph tracing effect. When enabled, only those components and wires associated with the currently selected component are enabled. All other composite elements are “dimmed. See the Graph Tracing section for more details.
	**Show Link Labels** - Displays or hides link (wire) labels.
	**Zoom** - Sets zoom percentage for displaying the composite overview in the window.
	**Export to Image** - Save the composite overview as an image.
	**Print** - Print the composite overview as a graphic.

Overview Pane

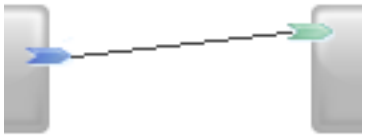
The Overview pane displays the relationships among services, components, and references for the selected composite. The services are displayed on the left side of the overview pane, and the external references are displayed on the right.

You can roll the mouse pointer over icons in the Overview pane to display information about the items, and you can right-click anywhere in Overview pane to display a context menu. To do either of these actions, the Overview pane must have focus. Click anywhere in background of the Overview pane to give it focus.

Overview Pane Icons and Behaviors

The following table identifies and describes the icons that represent aspects of the composite.

Icon	Name / Description / Behavior

	**Component** • Represents a component in the composite. • Roll the mouse pointer over the component to display its name. • Click a component in the Overview pane to display its services and external references in the Detail View pane. • Double-click a component to navigate to its implementation (if it exists in the same project).
	**Component service tag** • Represents a component service associated with the component. • Roll the mouse pointer over the tag to display the name of the service.
	**Component reference tag** • Represents a component reference associated with the component. • Roll the mouse pointer over the tag to display the name of the reference.
	**Component property tag** • Represents a component property associated with the component.
	**Link (or wire)** • A line that connects a component to a composite service (on the left) or a composite external reference (on the right), to indicate a dependency. • Click  **Show Link Labels** on the toolbar or right-click in the background and select **Show Link Labels** to display or hide link (wire) labels, which show what service or reference is connected to the component by the wire.
	**Component service** • Represents a service in the composite. • Roll the mouse over this icon to display its name.

	**Component external reference** • Represents a reference to an SCA service in an SCA composite that is external to this composite. • Roll the mouse over this icon to display its name.
	**Unresolved external reference** • Represents an unreolved reference to an SCA service in an SCA composite that is external to this composite.

Overview Pane Context Menu

To display the context menu for the composite overview,

1. Click anywhere in the background to give focus to the Overview pane. Make sure that no composite or wire on the canvas is selected.
2. Right-click to display the menu. It contains the following commands:

Command	Description
Enable Graph Tracing (toggle)	Turns graph tracing on or off. When on, the dependencies on the selected item(s) are enabled. All other composite elements are dimmed. See Use Graph Tracing for more details.
Show link labels (toggle)	Displays or hides link (wire) labels.
Zoom	Sets zoom percentage for displaying the composite overview in the window.
Export to image	Save the composite overview as an image.
Print	Print the composite overview as a graphic.
Submit to AquaLogic Enterprise Repository	Submit the composite and its dependencies to an AquaLogic Enterprise Repository. See Submit Composites to AquaLogic Enterprise Repository.

Detail View Pane

The Detail View pane provides a detailed view of the services and references for a component selected in the Overview pane. The component's services are displayed on the left side of the Detail View pane, and the external references are displayed on the right.

Icon	Name / Description / Behavior

	**Component Service** • Represents a service in the composite. • Roll the mouse over this icon to display a tooltip containing an XML fragment of the component.
	**Component Reference** • Represents a reference in the composite. • Roll the mouse over this icon to display a tooltip containing an XML fragment of the reference.
	**Expand/collapse** - Expand or collapse the representation of a composite external reference

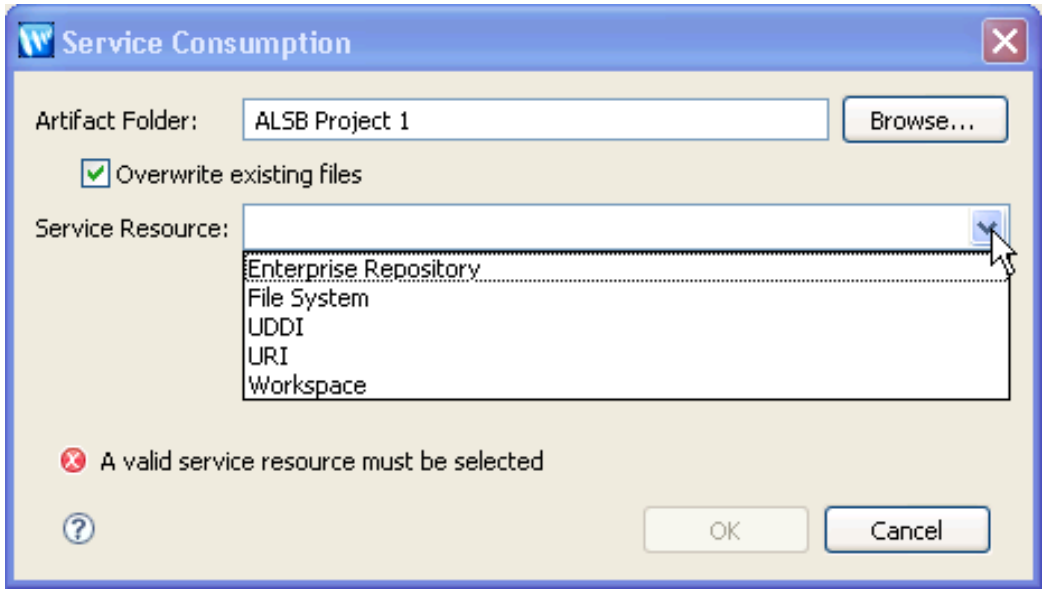
If a component has configured properties, those properties are displayed under **Properties**, at the bottom of the pane.

Service Consumption Dialog

The **Service Consumption** dialog allows you to consume services from the following resource types:

- **AquaLogic Enterprise Repository**: consume a service through an AquaLogic Enterprise Repository using the Enterprise Repository Access View.
- **file system**: consume a WSDL file in your local file system.
- **UDDI**: consume a service via a UDDI registry.
- **URI**: consume a remote WSDL file.
- **workspace**: consume a service residing in the current workspace

Select the resource type using the **Service Resource** dropdown menu, as shown below.

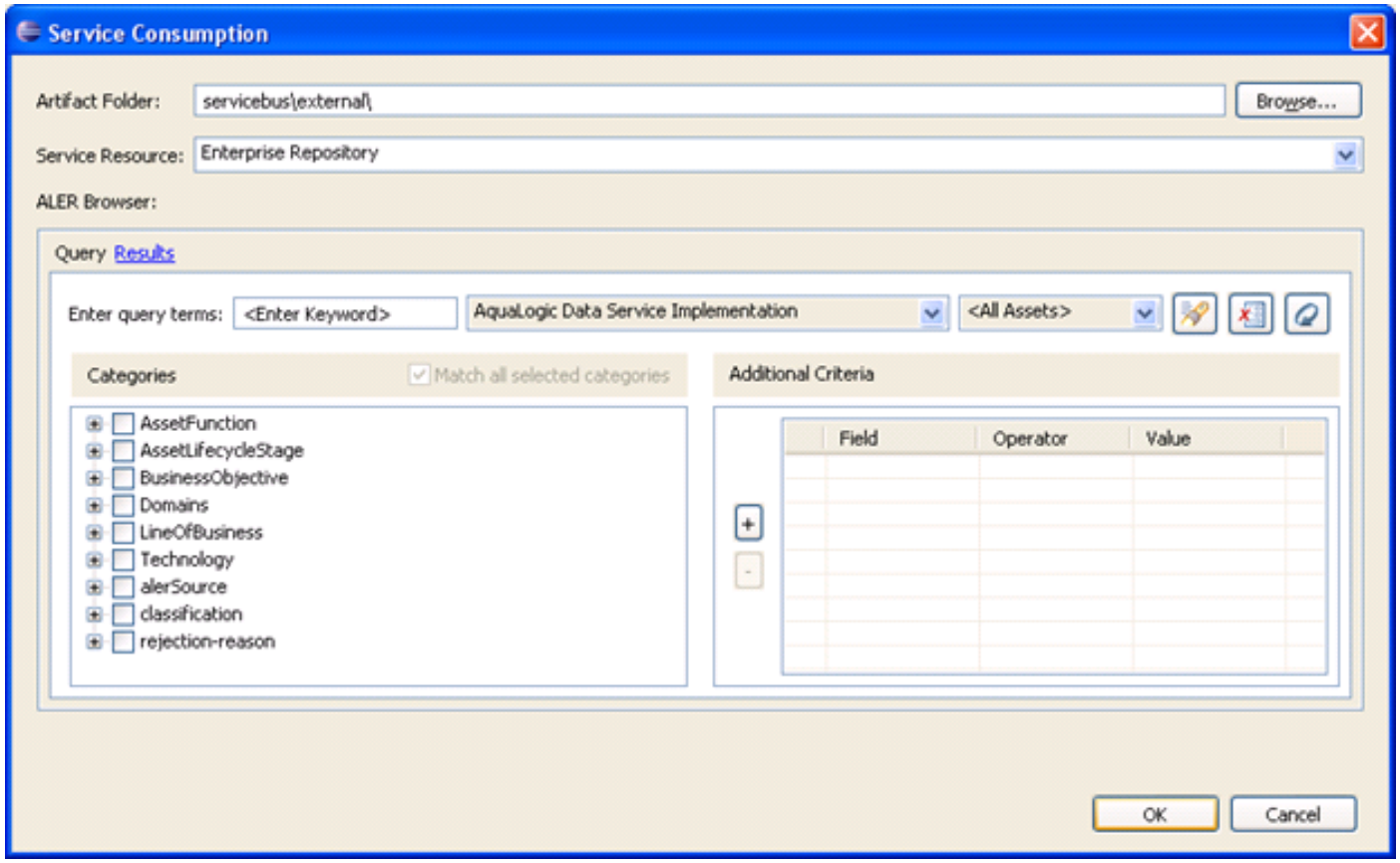


> **Note** that the **Artifact Folder** field allows you to select the download folder for files associated with the service, such as WSDL files. The default location is the project root folder.

The wizard provides additional options depending on the resource type selected:

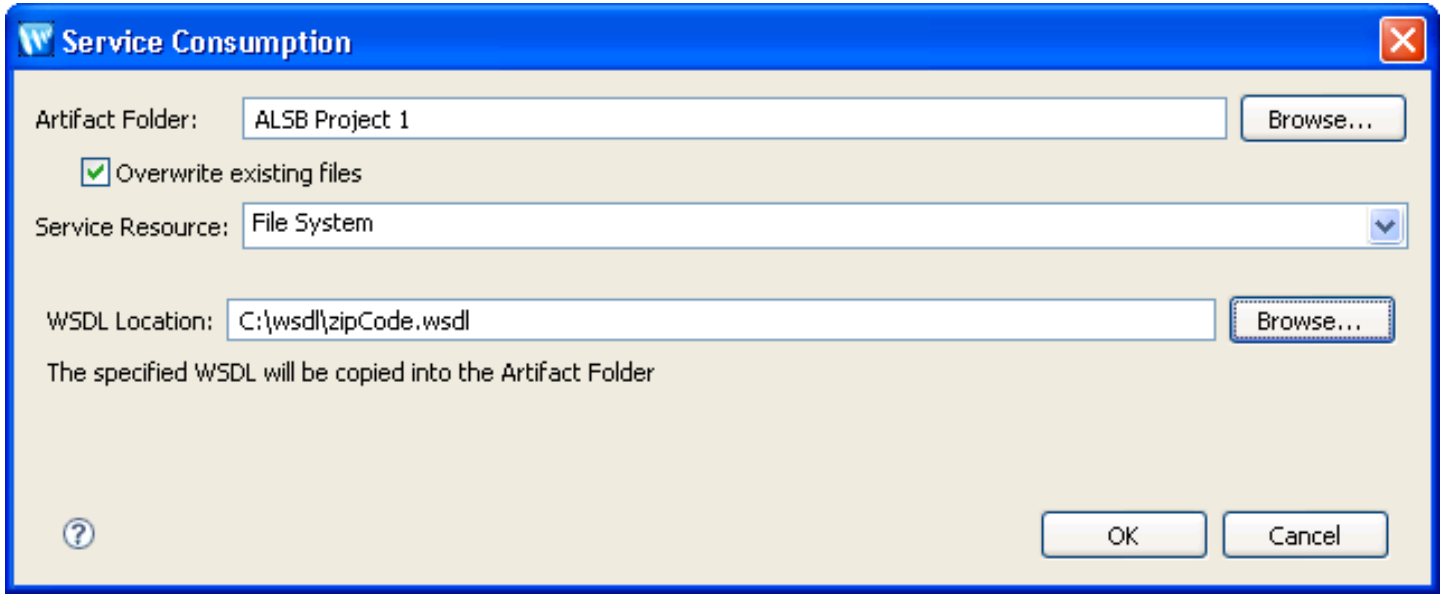
AquaLogic Enterprise Repository

For details on connecting to an enterprise repository, see the Enterprise Repository Access View



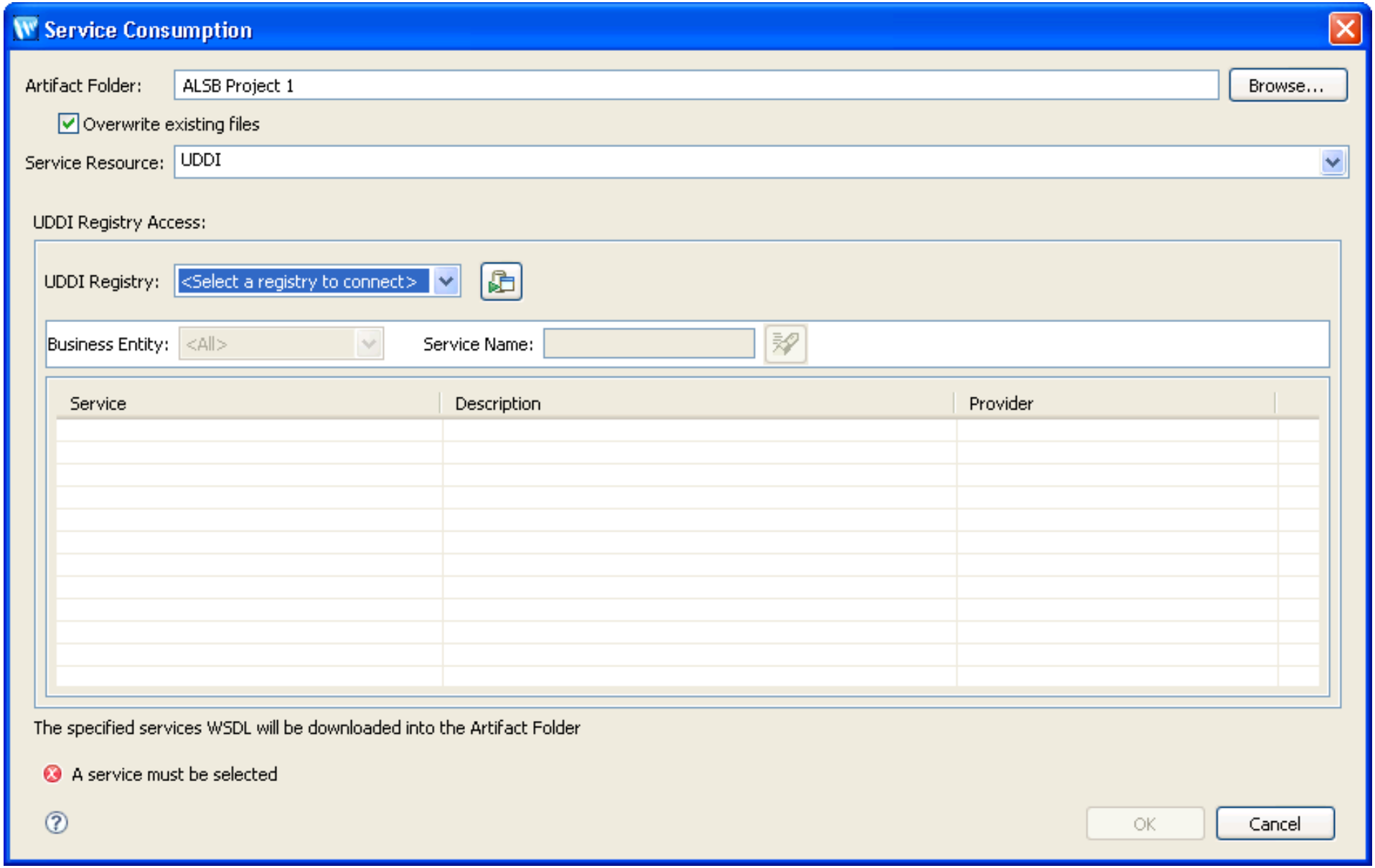
File System

Browse to a WSDL file in your local file system. Note that the local WSDL will be copied to the Artifact Folder.



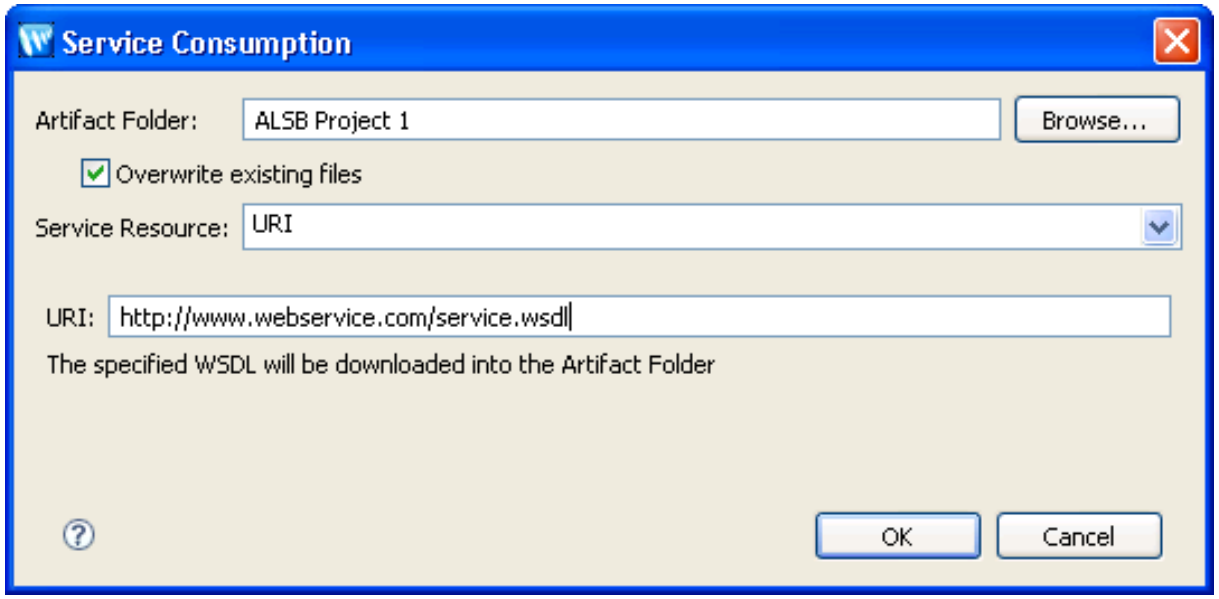
UDDI Registry

When **UDDI** is selected, the wizard provides additional options for connecting to a UDDI registry.



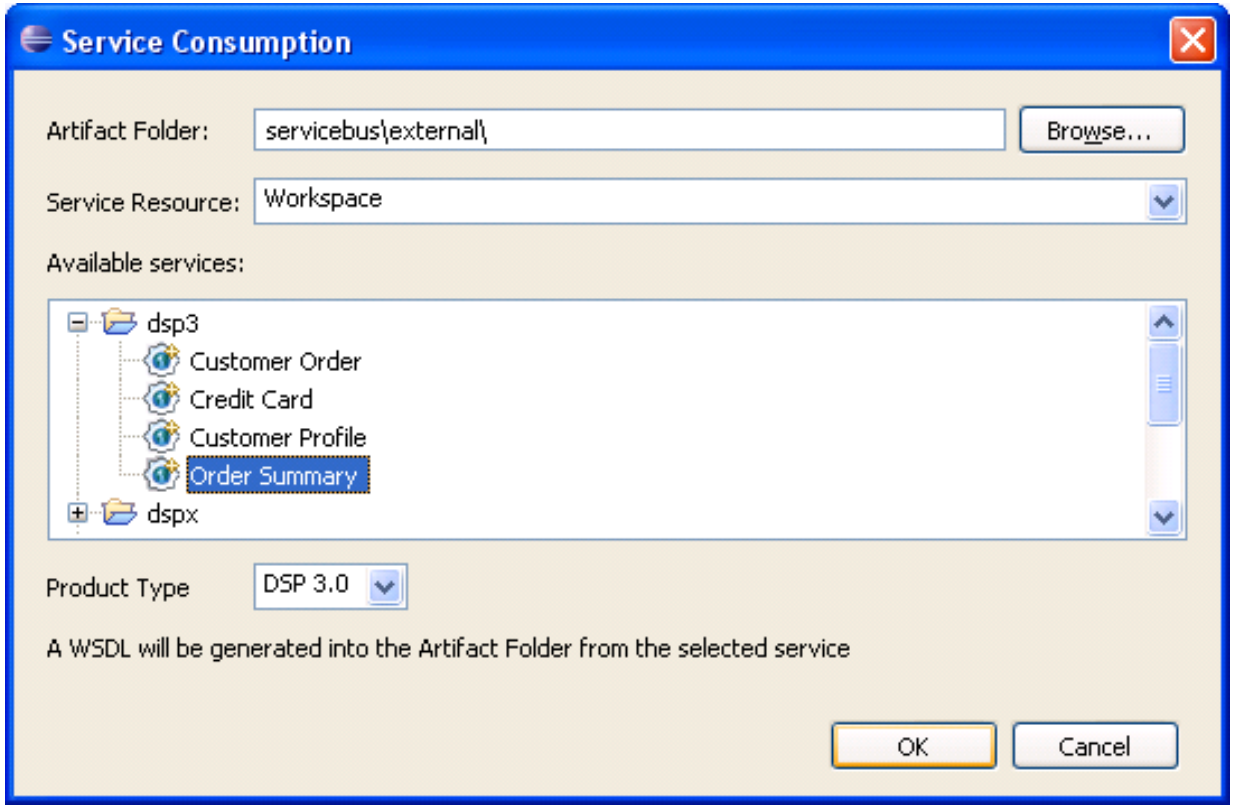
URI

The WSDL file you specify in the **URI** field will be downloaded.



Workspace

When **Workspace** is selected, consumable services are displayed for each project in the current workspace.


Service Consumption
Artifact Folder:
servicebus\external\
Browse...
Service Resource:
Workspace
Available services:
dsp3
Customer Order
Credit Card
Customer Profile
Order Summary
dspx
Product Type
DSP 3.0
A WSDL will be generated into the Artifact Folder from the selected service
OK
Cancel

Keyboard Shortcuts

You can perform many actions in SAM using the mouse or the keyboard. The following table describes what you can do using the keyboard.

Pressing this...	When this has focus ...	Does this...
Tab and **Shift-Tab** or **Ctrl-Tab** and **Shift-Ctrl-Tab**	Anywhere in Eclipse	Moves forward and backward between groups of controls.
Right-arrow and **Left-arrow**	Any toolbar	On a toolbar, cycle through the buttons in a group of buttons.
Up-arrow and **Down-arrow**	Composite Overview pane in the **Model** tab of the SAM Composite view (see SAM Composite View)	• When a component is selected, moves focus from component to component vertically. • When graph tracing is on and a link is selected, cycles focus among any other outbound links from the same component.
Right-arrow and **Left-arrow**	Composite Overview pane in the **Model** tab of the SAM Composite view (see SAM Composite View)	• When a node is selected, moves focus from node to node horizontally. • When graph tracing is on and a link is selected, cycles focus between a component's inbound link(s) and outbound link(s).
Shift-F10	Any item that has a context-menu associated with it.	Displays the context menu (equivalent to right-clicking an item).
Enter	Any item that responds to a double-click (for example to open a file)	Equivalent to double-clicking.
F1	Anywhere in Eclipse.	Displays help. Usually, when an item such as a view or a dialog has focus, a help topic about that item is displayed.

Eclipse provides many other built-in keyboard shortcuts, and you can modify many of them, as follows:

1. Select **Windows>Preferences.**

2. In the Preferences dialog, select **General > Keys**.
3. Edit the settings, according to Eclipse guidelines.

Service Component Architecture (SCA) Glossary

assembly model
See SCA assembly model.

component
See service component.

composite
See SCA composite.

model
See SCA assembly model.

project
In Eclipse, a collection of related folders and files. A SAM project in Eclipse is the Eclipse representation of an SCA assembly model.

reference
A dependency that an implementation has on a service that is supplied by some other implementation, where the service to be used is specified through configuration. In other words, a reference is a service that an implementation may call during the execution of its business function.

SAM
See Service Assembly Modeler.

SCA
See Service Component Architecture.

SCA assembly model
A series of artifacts which define the configuration of an SCA Domain in terms of composites which contain assemblies of service components and the connections and related artifacts which describe how they are linked together.

SCA composite
An application assembled as a series of services to serve a business need. An SCA composite assembles SCA elements in logical groupings. It is the basic unit of composition within an SCA Domain. An SCA composite contains a set of components, services, references and the links that interconnect them, plus a set of properties which can be used to configure components

SCA domain
A representation of related of SCA-enabled services, for example service that provide a set or related business functionality.

SCA-enable
To create an SCA composite based on an existing application. When you SCA-enable an application, you create SCA components and other artifacts based on the corresponding features in the source application.

service
: A business function, provided by a component, that is available for use by other components. Components provide and consume services.

Service Assembly Modeler (SAM)
: An Eclipse Plug-in for assembling and viewing SCA assembly models. SAM is based on SCA.

service component
: A configured instance of an implementation, where an implementation is code that provides business functions. The business functions are available for use by other components as *services*. Implementations can have settable properties, which influence the operation of the business function. A component configures its implementation by providing properly values and by associating references to services provided by other components.

Service Component Architecture (SCA)
: A programming model for building applications and services based on a Service Oriented Architecture (SOA). SAM is based on the SCA Service Component Architecture Assembly Model Specification specification. See http://www.osoa.org/display/Main/Service+Component+Architecture+Specifications.