BEA AquaLogic® Enterprise Repository

ALER Integration with Microsoft Visual Studio .NET

Version 3.0.1
Revised: February, 2008

Table of Contents

Installing the ALER Plug-in for VS. NET

Overview

ALER integration with Visual Studio .NET provides users with the ability to easily search for and use assets from the repository without leaving the VS .NET IDE environment. Assets and any associated artifacts are downloaded directly to your VS .NET solution. Repository Access within the VS .NET solution also provides a view into ALER that enables you to download artifacts and assets from the repository, query the repository, and view the contents of the repository.

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. For more information, see **Enabling Advanced Usage Detection Options**.

Prerequisites

- Microsoft Visual Studio 2005.
- Microsoft Visual J# 2005 runtime. (If J# is not installed on your machine, the installer will prompt you download the correct version from Microsoft.)
- The VS .NET **Always show solution** option should be selected (Tools -> Options -> Projects and Solutions -> General).
- Users must be assigned to at least one ALER project. A **Project Administrator** can assign users to projects using the ALER **Projects** page.
- If your ALER is or will be secured by Siteminder, you will need to configure the policy server to ignore (or unprotect) the following URL to allow the OpenAPI integration to function properly:

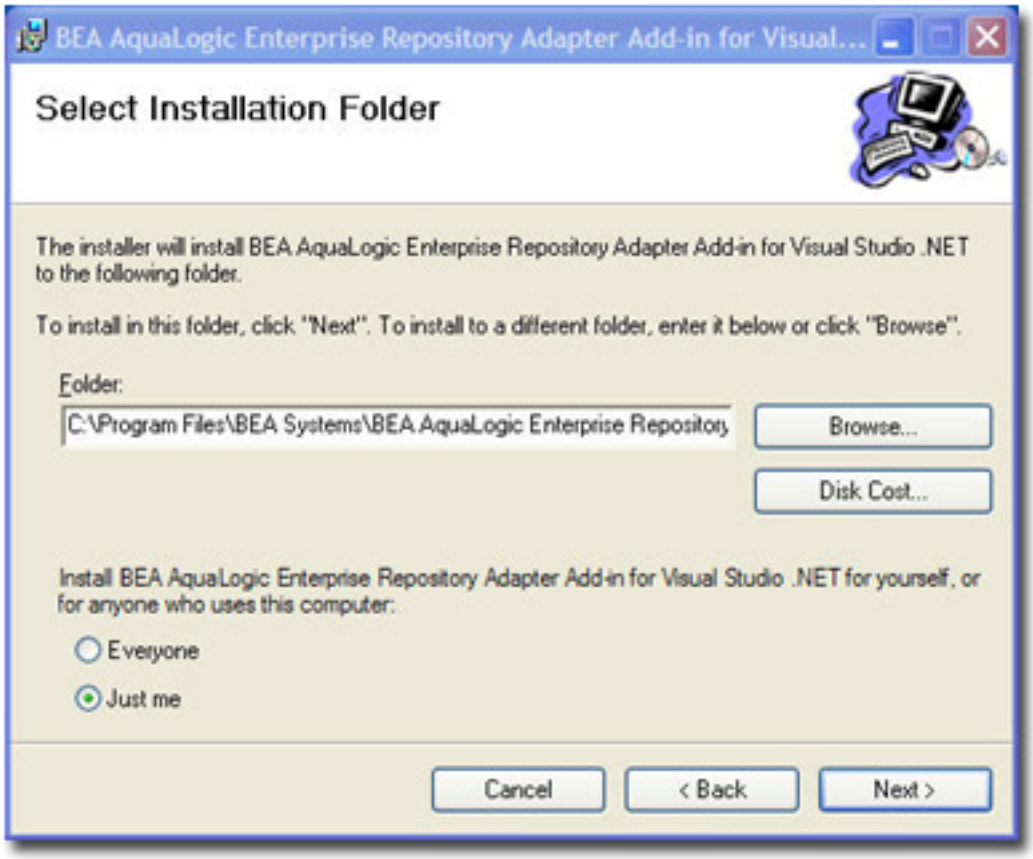
  ```
  http://appserver.example.com/ALER/services/
  ```

Installation

1. Download the VS .NET plug-in Zip file from your ALER instance at the following URL:

   ```
   http://appserver.example.com/ALER-web/integration/dotnet/
   ALERForVisualStudio_3_0.zip
   ```

2. Unzip the `ALERForVisualStudio_3_0.zip` file.
3. Locate and run the `setup.exe` program.
4. Follow the prompts to select installation parameters.

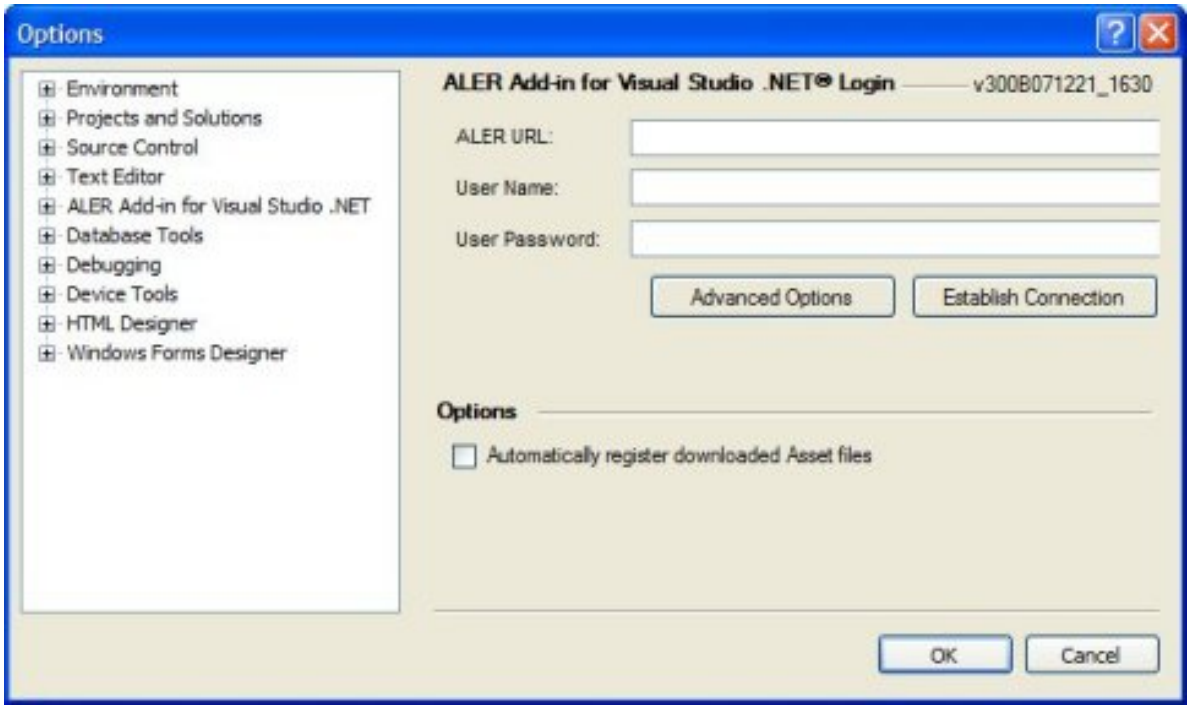


5. Click **Finish** to complete the installation.
6. Follow the instructions in **Configuring the Connection to ALER from VS .NET** to configure and establish a connection to an ALER instance from VS. NET.

Configuring the Connection to ALER from VS .NET

Follow these steps to configure and establish a connection to an ALER instance from VS. NET.

1. Launch Visual Studio .NET.
2. Open the **Tools** menu and click **Options**.
3. On the list of options, click the **ALER Add-in for Visual Studio .NET** option and provide the required login information.



   - **ALER URL**
     - The URL of the ALER instance. For example: `http://appserver.example.com/ALER`

       Note: Do not include the `index.jsp` used in the default home page as part of the URL.

4. **User Name**
   - The user name to connect as.
5. **User Password**
   - The password to connect with. Passwords are case-sensitive.
6. **Establish Connection**
   - Click to verify a valid connection.
7. **Automatically register downloaded Asset files**
   - If selected, downloaded asset files are registered with the Windows Registry, as appropriate. This may be overridden on an case-by-case basis for each asset download.

8. Click the **Establish Connection** button to connect to the ALER instance you specified.

9. Optionally, click the **Advanced** button to enable additional ALER options:
    - Usage detection for VS .NET Solution Projects
    - Automated usage detection of referenced DLLs, WSDLs, and allow local caching of SFIDs (if SFID is enabled at your installation)
    - File name patterns to include and exclude

    For more information, see **Enabling Advanced Usage Detection Options**.

10. Click **OK** when finished.

Enabling Advanced Usage Detection Options

Follow these steps to enable advanced configuration options, such as enabling automatic usage detection of DLLs, WSDLs, local caching of SFIDs, and file pattern detection.

Overview of SFID

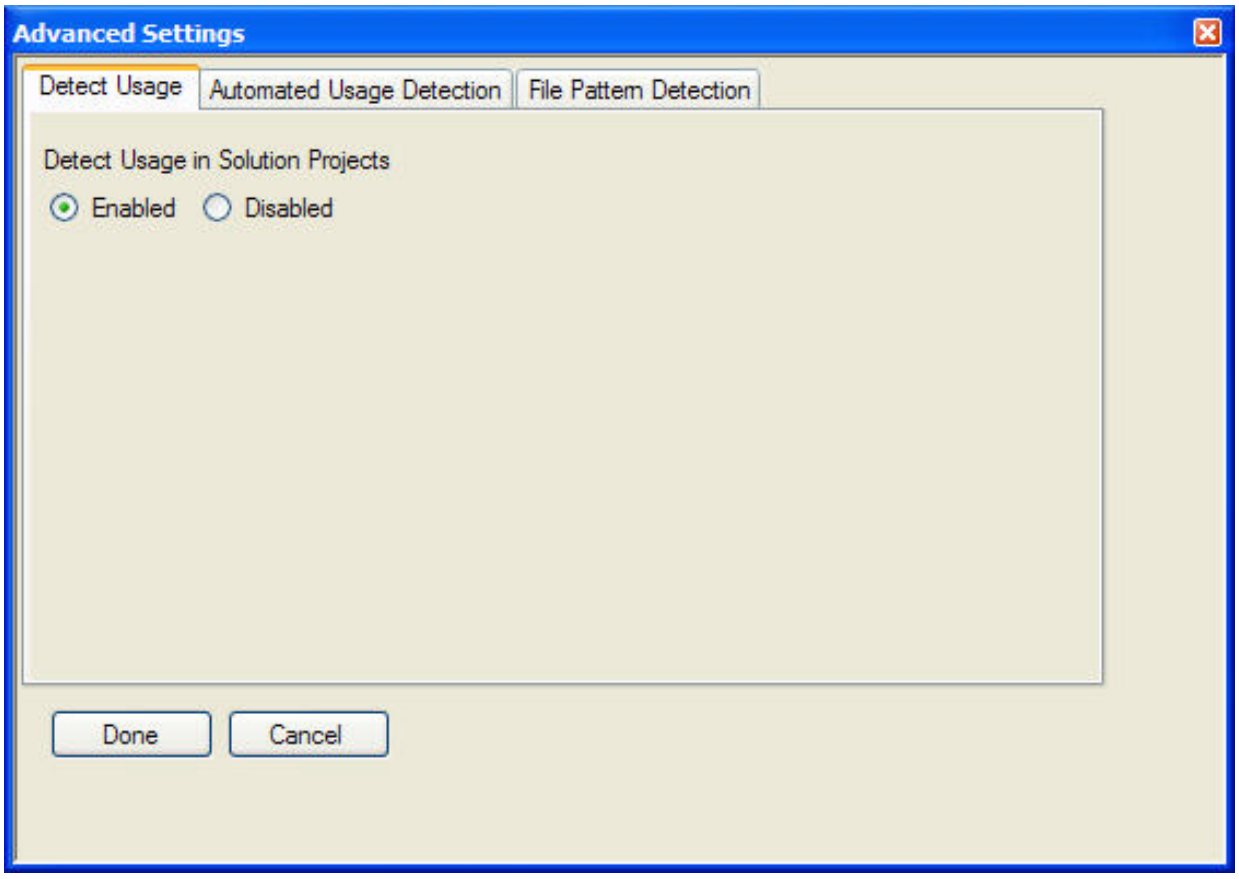
If SFID is enabled at your installation, ALER can automatically detect asset reuse within the development environment. This allows development teams to ensure that they get asset reuse credit, regardless of whether the assets have been downloaded through ALER. Automated Usage Detection relies on a fingerprinting 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.

SFID is available only in the Advanced Edition of ALER. See the *ALER Software File Identification Guide* for more information.

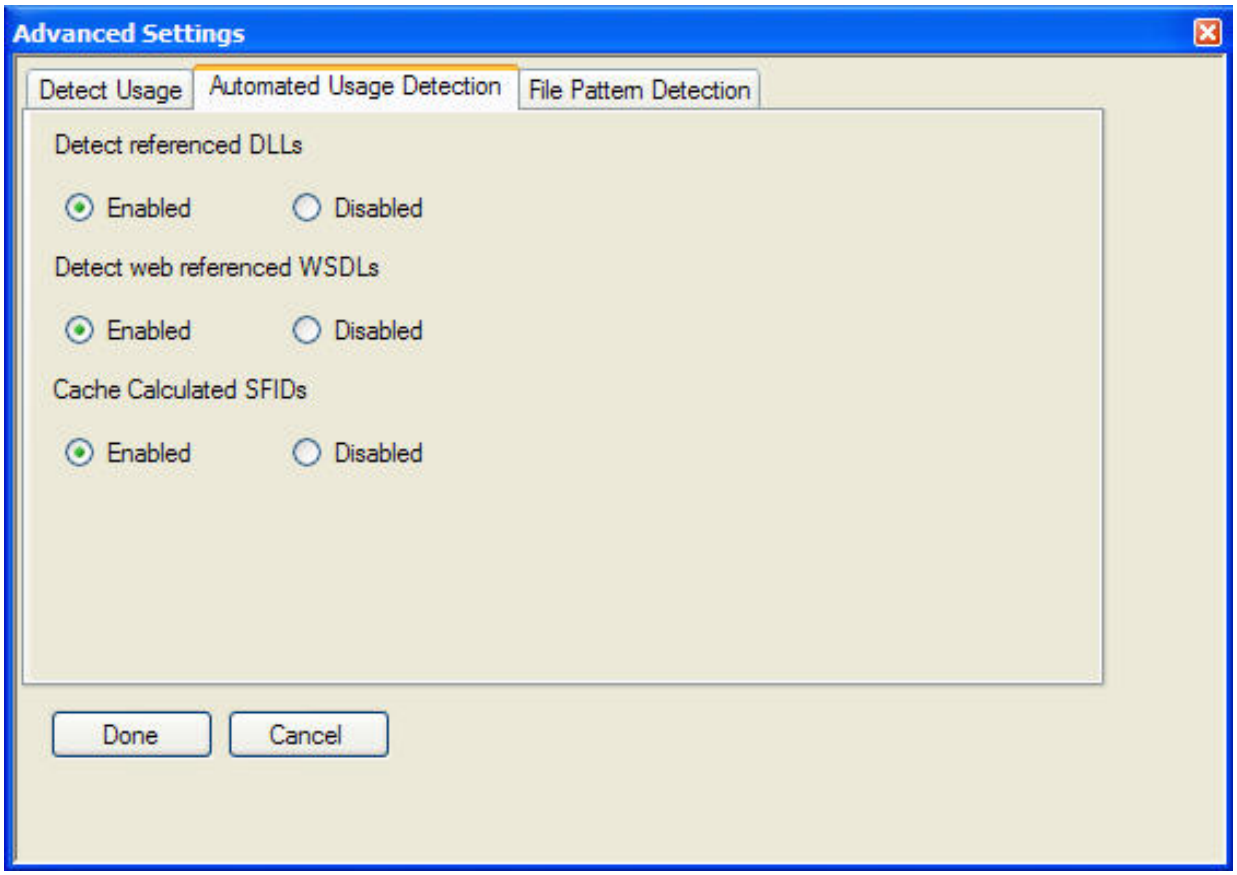
Configuring Automatic Usage Detection

1. Launch Visual Studio .NET.
2. Open the **Tools** menu and click **Options**.
3. On the list of options, click **ALER Add-in for Visual Studio .NET** to reopen the Login window.
4. Click the **Advanced Options** button to open the Advanced Settings window.

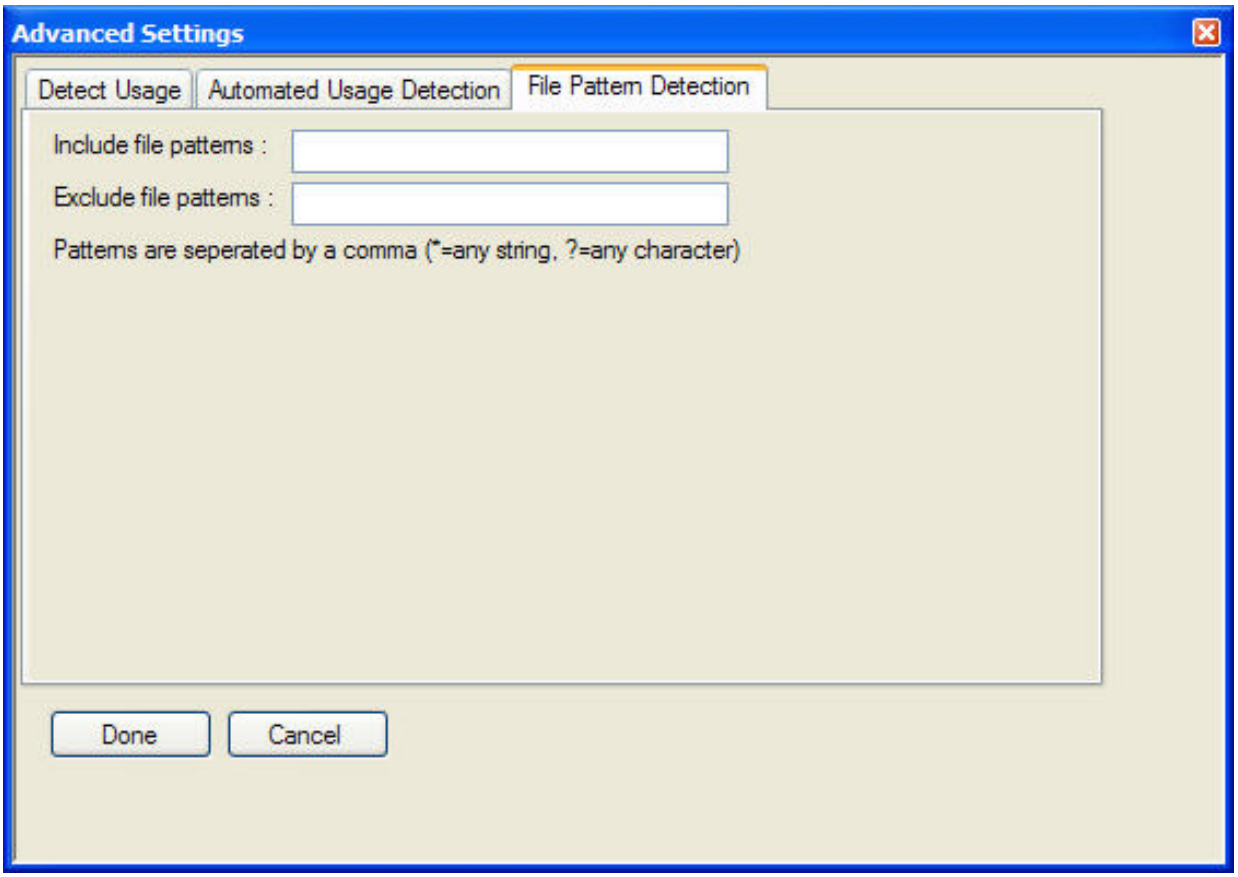
   Use the **Detect Usage** tab to enable usage detection for VS .NET Solution Projects.



5. Click the **Automated Usage Detection** tab to enable usage detection of referenced DLLs, WSDLs, and allow local caching of SFIDs.



6. Click the **File Pattern Detection** tab to specify include and exclude file name patterns.

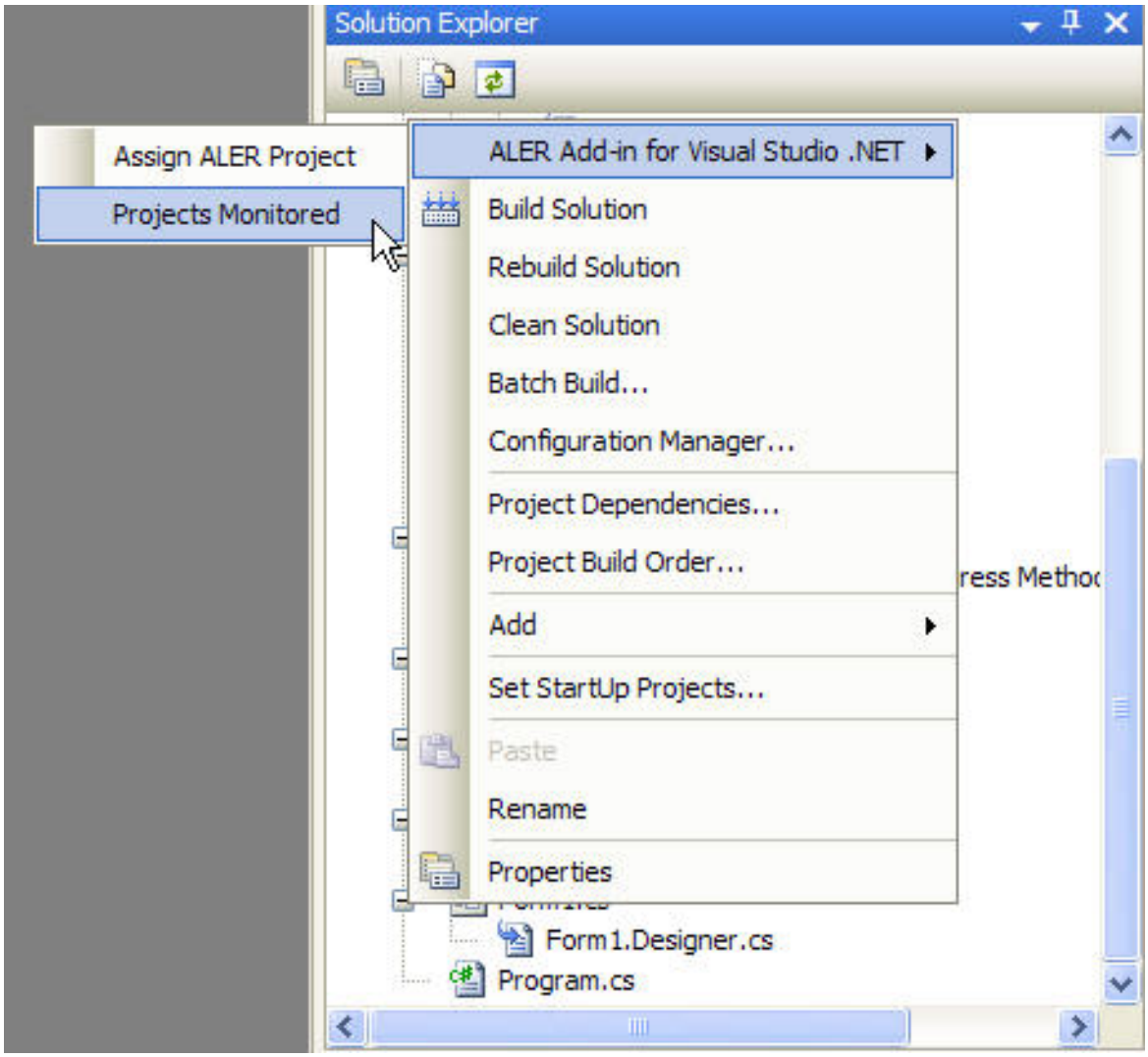


7. Click **Done** to save your settings.

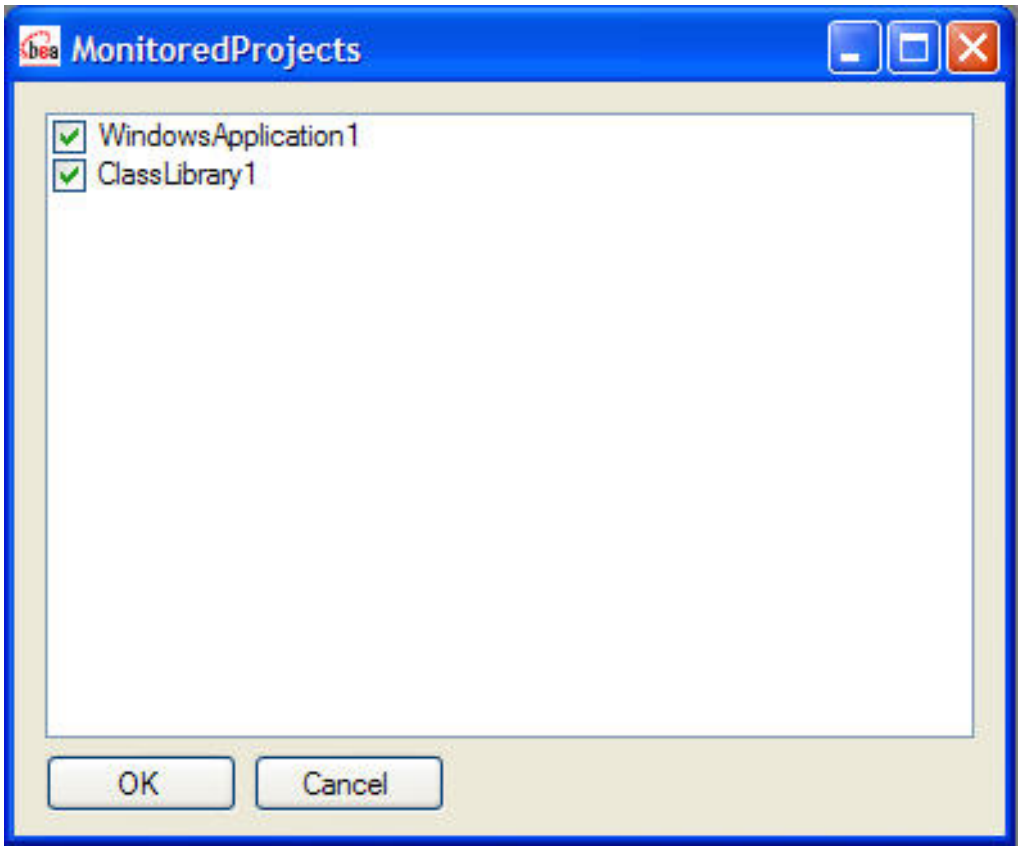
Configuring .NET Projects for Automatic Usage Detection

In order to automatically detect usage, a .NET project must be monitored.

1. Open the .NET **Solution Explorer**.
2. Right-click the solution in the file tree and select the **ALER Add-in for Visual Studio .NET** option from the context menu.
3. Click **Projects Monitored** from the submenu.



4. In the **MonitoredProjects** window, select the .NET projects that you want monitored for automated usage detection.



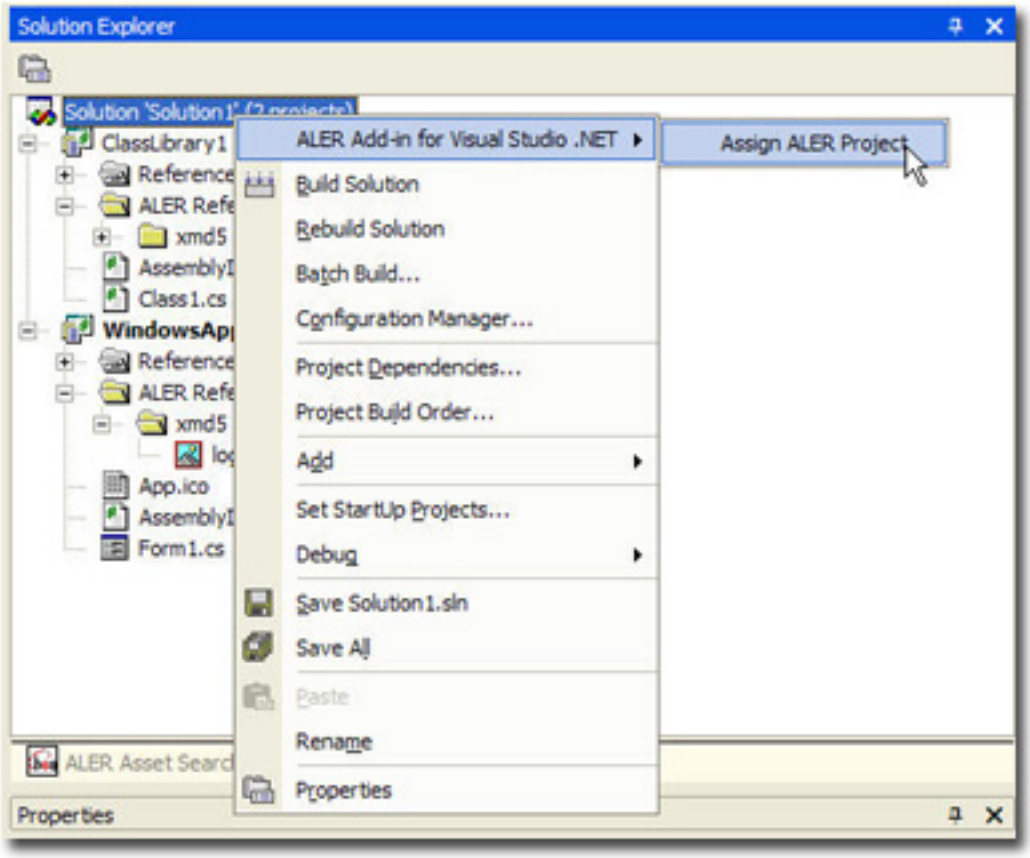
5. Click **OK** when finished.

Assigning an ALER Project to a .NET Solution

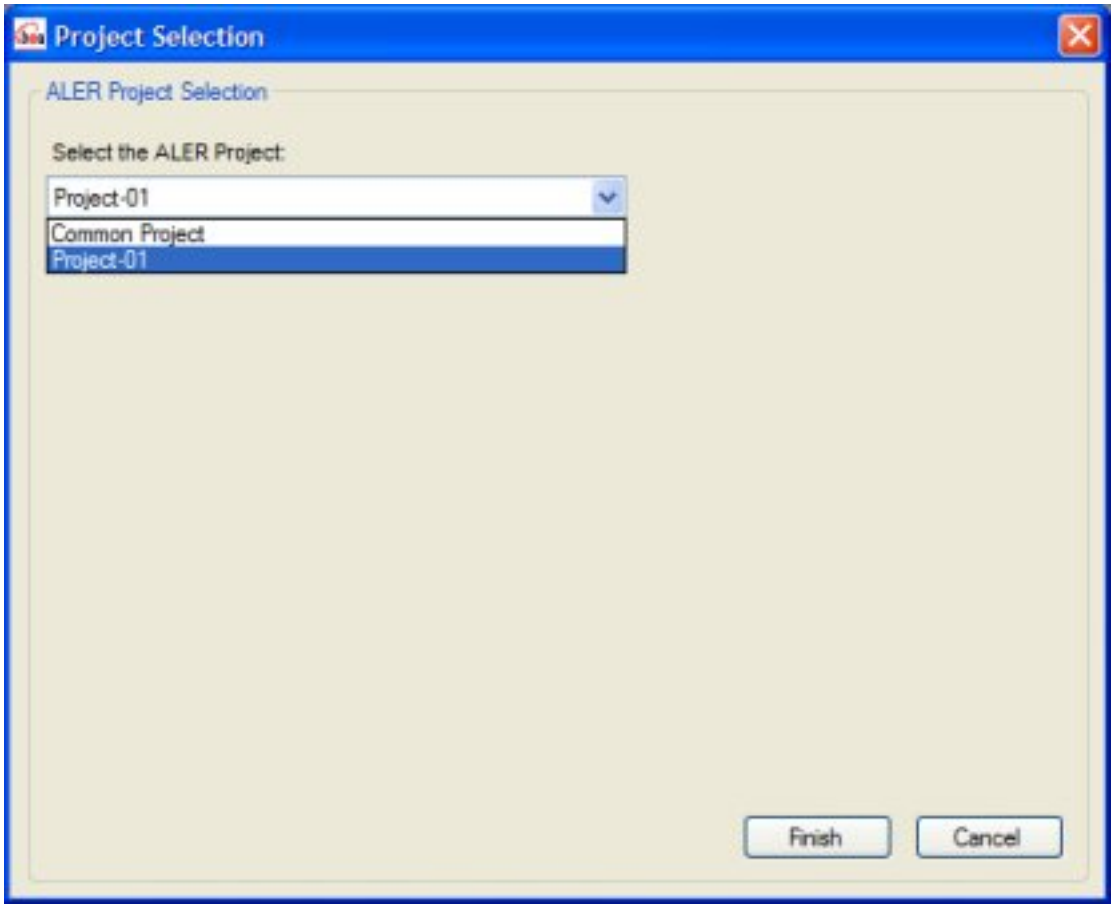
In order to track the usage of downloaded assets, an ALER project must be assigned to a .NET solution.

Note: Before using this feature, you must be assigned to at least one ALER Project by a Project Administrator.

1. Open the .NET **Solution Explorer**.
2. Right-click a solution in the file tree and select the **the ALER Add-in for Visual Studio .NET** option from the context menu.
3. Click **Assign ALER Project** from the submenu.



4. In the **Project Selection** window, use the **Select the ALER Project** drop-down list to view the ALER projects that you are assigned to.



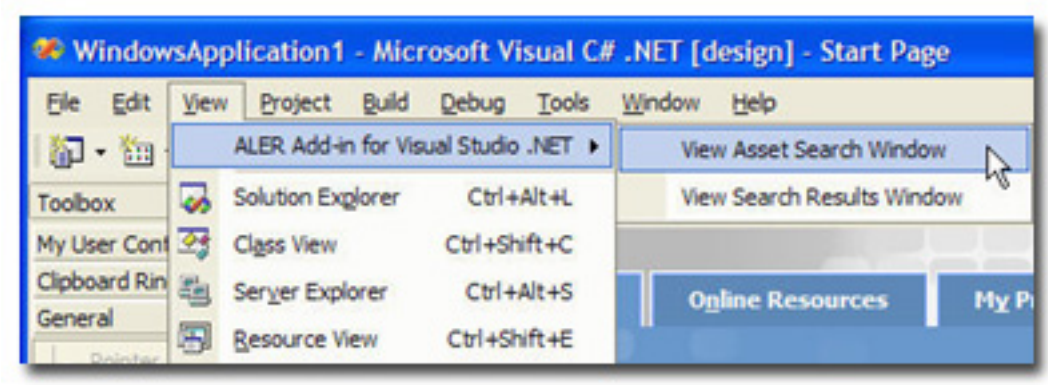
Note: If the list is empty, you have not been assigned to any projects and the procedure must be canceled.

5. Select an ALER project from the list.
6. Click **Finish** to save your changes.

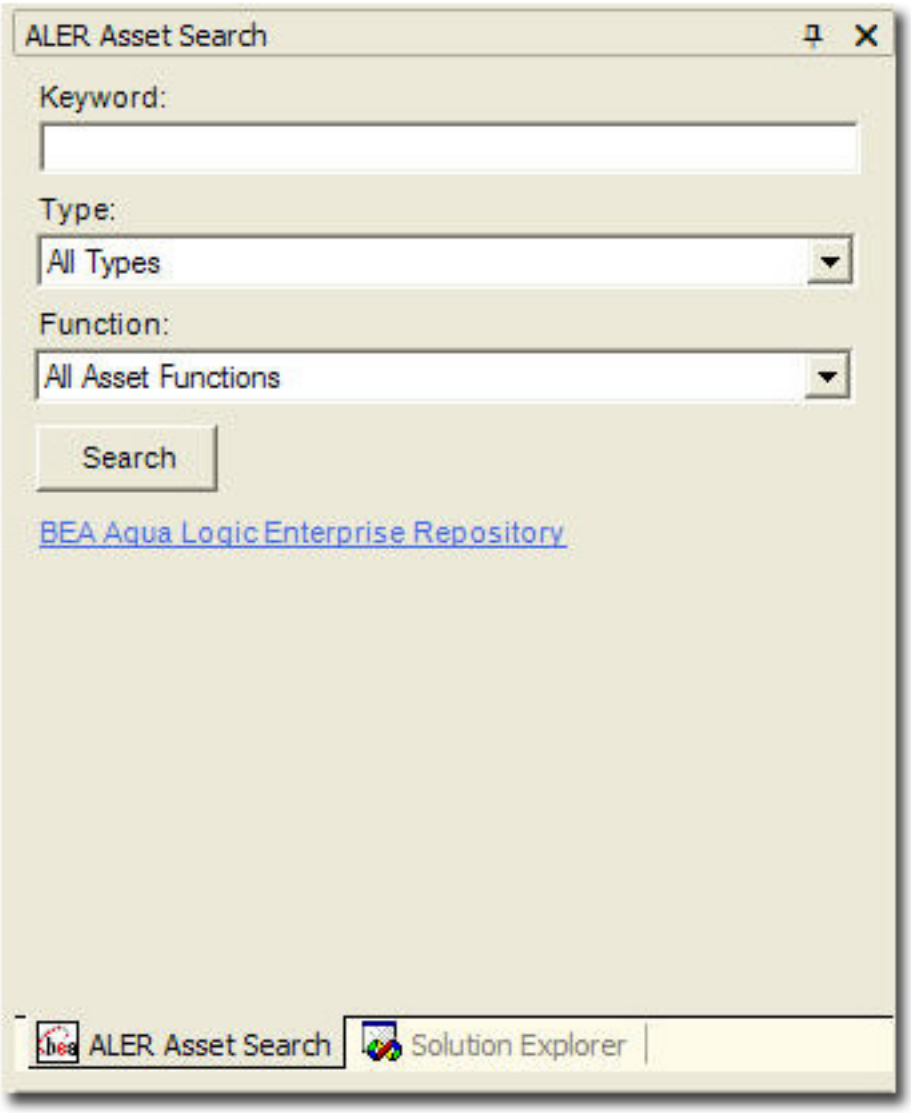
Searching for Assets

Perform a simple keyword search in VS .NET to locate an asset in ALER in order to view the asset's metadata.

1. Open the **View** menu.
2. Click the **ALER Add-in for Visual Studio .NET** option and select **View Asset Search Window** from the submenu.



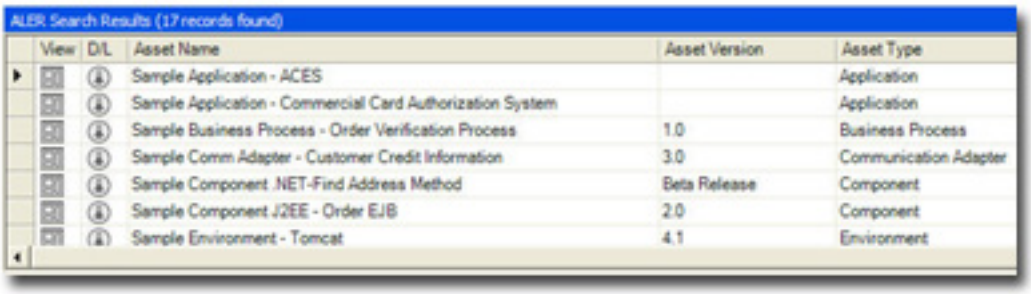
3. The **ALER Asset Search** window appears.



4. Type a keyword or phrase into the **Keyword** box.

5. Use the **Type** and **Function** drop-down lists to refine the search.

6. Click the **Search** button. The **ALER Search Results** window displays a list of all assets matching the search criteria.



7. Click the **Display Details** icon for any listed asset (or double-click the row) to view the asset's

detail display.

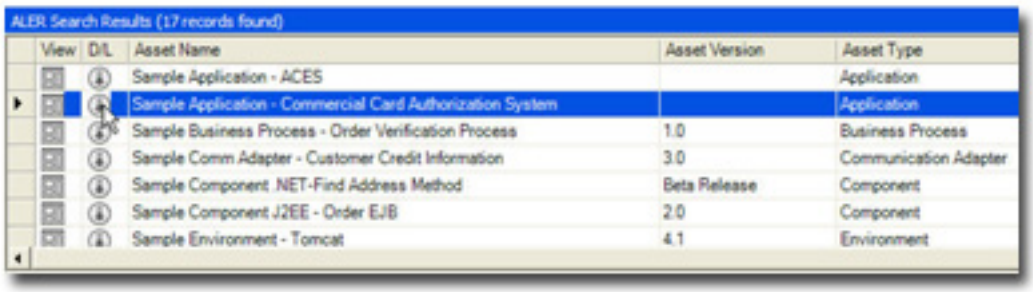
8. Click the **Download**  icon for any listed asset to download the asset.

Downloading Asset Artifacts

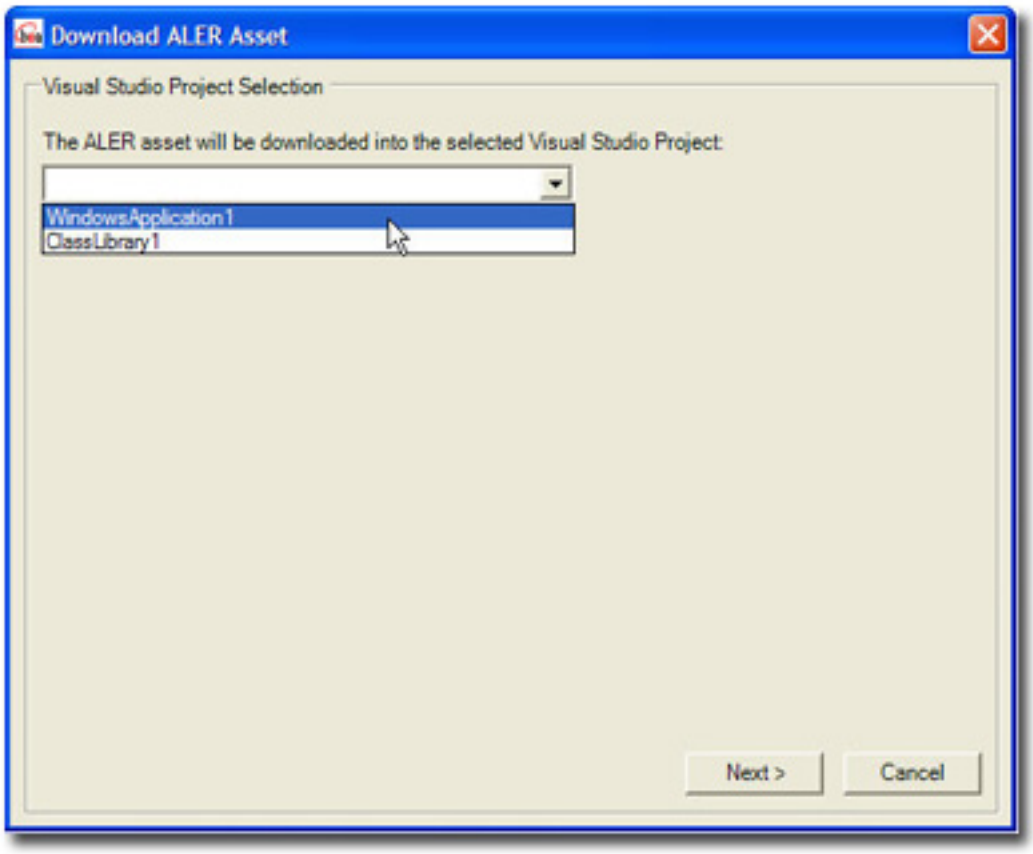
You can download an asset's artifacts (i.e., payload) into a VS .NET project. Typically an asset payload is usually the functionality that a developer needs to use a service (such as a WSDL file) or incorporate into their code base (usually a binary). Within the asset metadata, links to supporting documentation,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

1. Perform a keyword search to locate an asset in ALER, as described in **Searching for Assets**

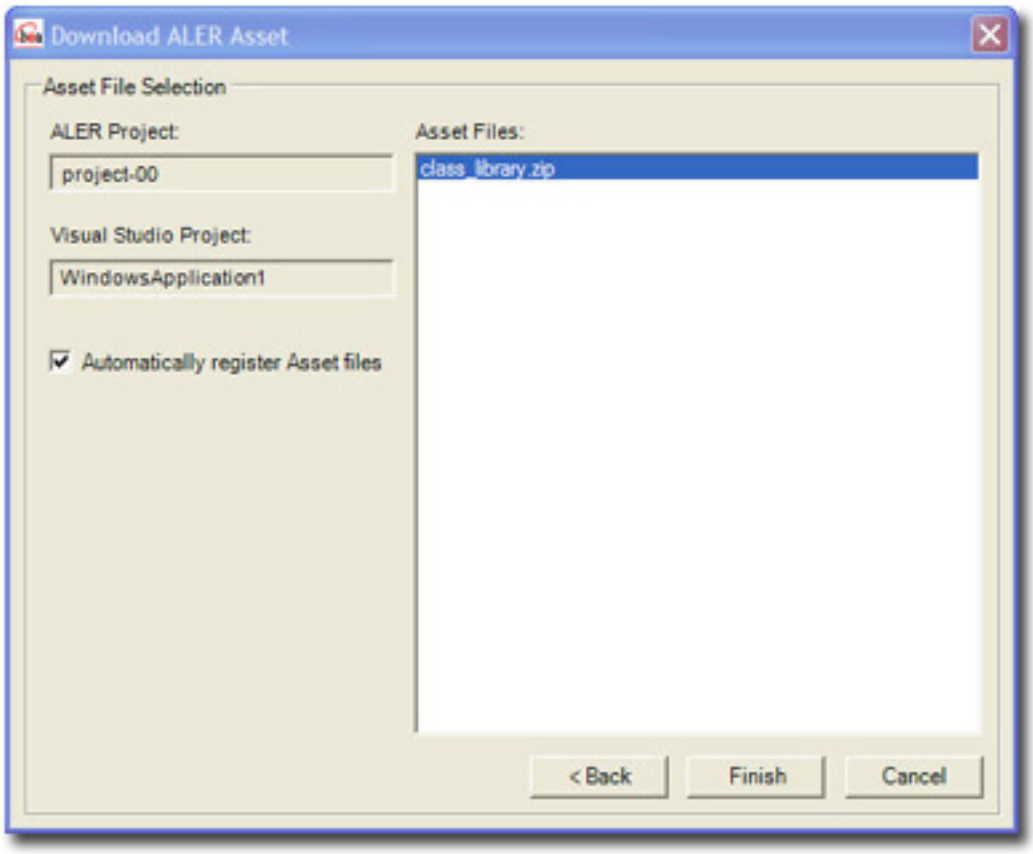
2. In the **ALER Search Results** window, click the **Download** icon.



3. Select the VS project that you want to download the asset's files into, and then click **Next**.

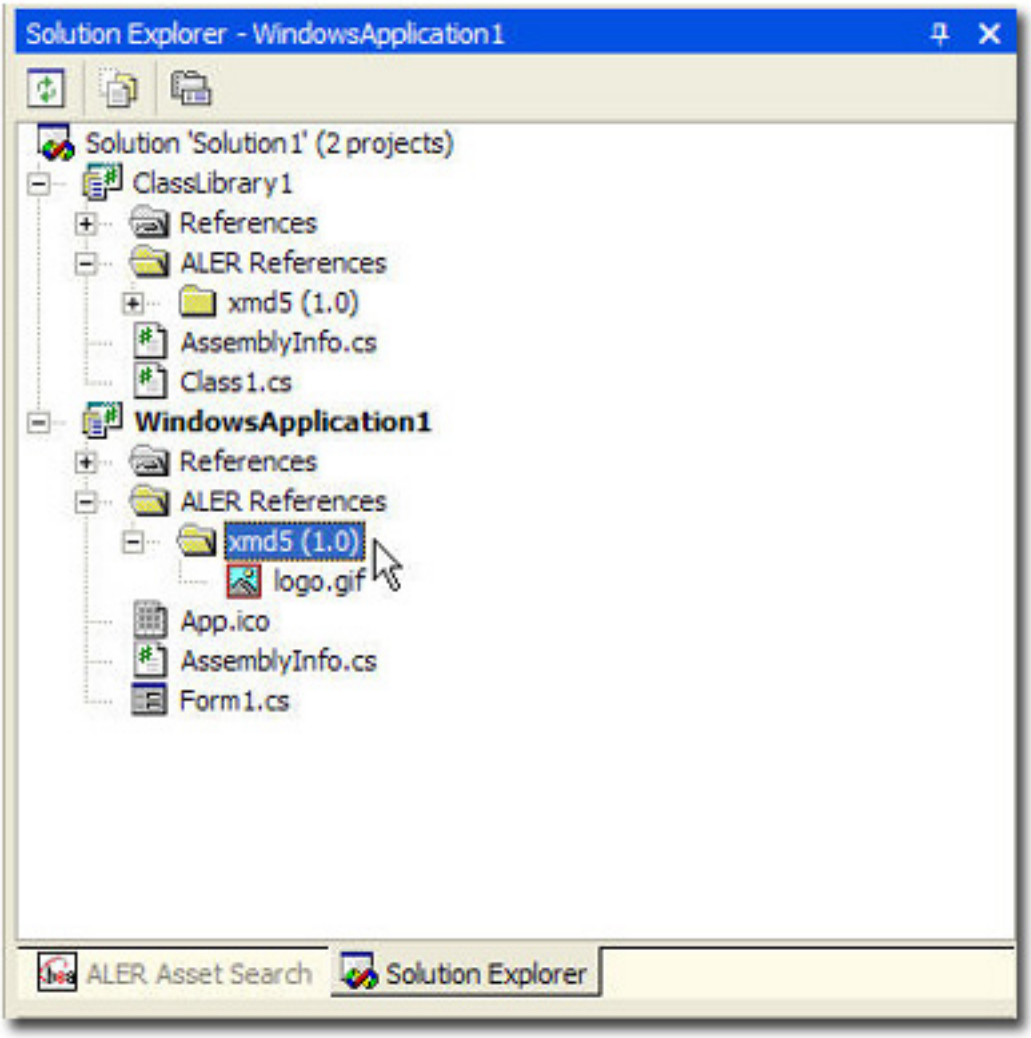


4. Select the asset files to download, and then click **Finish**.



5. If necessary, launch the **Solution Explorer**.

6. A new folder labeled **ALER References** will appear in the project's file tree. This folder contains a subfolder bearing the name of the downloaded asset (for example, "xmd5 (1.0)" in the illustration below). This folder contains the asset's artifacts.

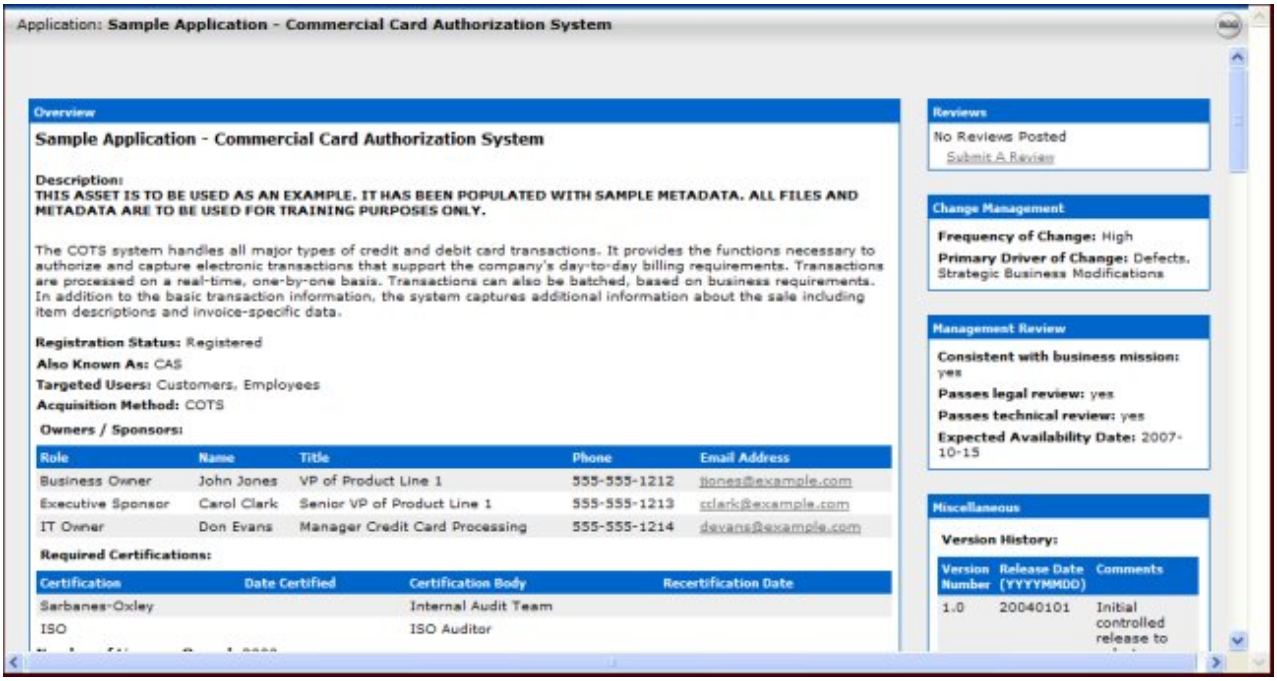


Viewing Asset Details

The Asset Details view provides asset details for any listed asset in an embedded Web-based browser view, which calls out the enterprise repository application for details associated with the selected asset.

1. Perform a keyword search to locate an asset in ALER, as described in **Searching for Assets**
2. In the **ALER Search Results** window, click the **Display Details** icon.

   ALER opens to display information on the selected asset, as shown here:



Accessing Repository Assets in a Project

The Repository 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.

About ALER Projects

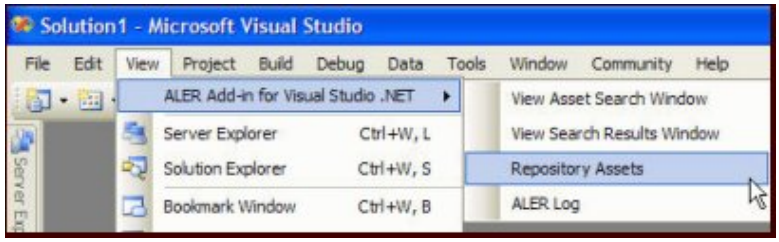
Through the Enterprise Repository, analysts, architects, technical leads, and others that are involved in the design stages of a project, can create a list of assets that might fulfill a project's requirements. The lists of assets are captured in compliance templates in the repository, and the compliance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

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

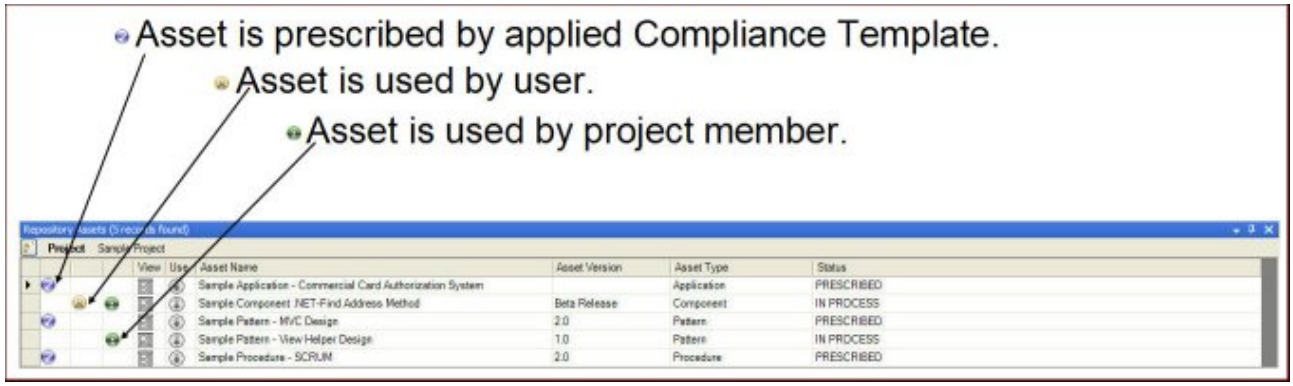
From the Repository Assets view, you can view a list of assets appearing in all of the Compliance Templates assigned to your project. The viewer will indicate which of the assets have been used by you and/or other project members. The viewer will also display other assets that are already in use in the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

Accessing the Repository Assets View

1. Open the **View** menu.
2. Click the **ALER Add-in for Visual Studio .NET** option and select **Repository Assets** from the submenu.



3. The **Repository Assets** window displays.



4. Click the **Display Details** icon for any listed asset (or double-click the row) to view the asset's detail display.
5. Click the **Download** icon for any listed asset to download the asset.

Accessing the ALER Log

1. Open the **View** menu.
2. Click the **ALER Add-in for Visual Studio .NET** option and select **ALER Log** from the submenu.
3. The **ALER Log** window displays.

4. Click **Clear** to remove the log entries.