AccuRev® Plug-In for Eclipse™

User's Guide

Version 2014.1

Copyright

Table of Contents

Preface

This book describes how to use AccuRev Plug-In for Eclipse (AccuRev Plug-In). AccuRev Plug-In is a standard Eclipse plug-in that provides access to AccuRev version-control facilities from the Eclipse Platform IDE. It assumes you are familiar with AccuRev and its features; a brief overview of AccuRev is provided in *Appendix A The AccuRev Usage Model*.

See the *AcccuRev Plug-In for Eclipse Installation and Release Notes* for information specific to the current release, including notes on platform support and a summary of changes and known issues.

Using This Book

This book assumes you are familiar with AccuRev and Eclipse. It contains the following chapters:

Chapter	Description
Chapter 1 Getting Started with AccuRev Plug-In for Eclipse	Describes how to get up and running with AccuRev Plug-In.
Chapter 2 Working with Eclipse Projects	Describes the different ways to create and share Eclipse projects.
Chapter 3 Working in the Eclipse Navigator View	Describes how AccuRev Plug-In integrates AccuRev functionality in the Eclipse Navigator view.
Chapter 4 Working in the AccuRev Perspective	Describes the features of the AccuRev perspective in the Eclipse IDE.
Appendix A The AccuRev Usage Model	Provides a brief overview of AccuRev SCM concepts and terms.

Typographical Conventions

This book uses the following typographical conventions:

Convention	Description
`blue sans-serif`	Used for sample code or output.
`red monospace`	Used for examples.
bold	Used for command names, icon names, and button names in the Eclipse user interface
light italic	Used for emphasis, book titles, and for first use of important terms
blue italic	Identifies a hyperlink (to a page or Web URL, for example)

Contacting Technical Support

AccuRev offers a variety of options to meet your technical support needs as summarized in the following table.

For	Visit
Information about technical support services	*http://supportline.microfocus.com/*
Information about platforms support	*http://supportline.microfocus.com/prodavail.aspx*
Product downloads and installations	*http://supportline.microfocus.com/websync/ productupdatessearch.aspx*
Product documentation	*http://supportline.microfocus.com/productdoc.aspx*
SupportLine phone numbers, listed by country	*http://www.microfocus.com/about/contact/support/ assistance.aspx*

When you contact AccuRev technical support, please include the following information:

- The version of AccuRev and any other AccuRev products you are using (AccuSync or GitCentric, for example)

- Your operating system

- The version of relevant third-party software (if you are using AccuSync, for example the version of your ITS)

- A brief description of the problem you are experiencing. Be sure to include which AccuRev interface you were using (Web user interface, Java GUI, or CLI), any error messages you received, what you were doing when the error occurred, whether the problem is reproducible, and so on.

- A description of any attempts you have made to resolve the issue

- A simple assessment of how the issue affects your organization

1. Getting Started with AccuRev Plug-In for Eclipse

AccuRev Plug-In for Eclipse (AccuRev Plug-In) provides access to AccuRev version-control facilities from the Eclipse Platform IDE. This chapter describes tasks that get you up and running with AccuRev Plug-In.

For a brief overview of AccuRev, see *Appendix A: The AccuRev Usage Model*.

Logging In to AccuRev

You must be logged in to AccuRev in order to perform AccuRev commands using AccuRev Plug-In. You can log in to AccuRev in one of these ways:

- In AccuRev directly, using one of these interfaces: the AccuRev GUI, the Web User Interface (Web UI), or the command line interface (CLI). If you are logged in to AccuRev when you start Eclipse, AccuRev commands are available for any elements in your Eclipse projects that are under AccuRev control.

- In AccuRev Plug-In, using the **Team** > **Login** menu in the Eclipse Navigator or Package Explorer views. This menu is available only for Eclipse projects and elements that are under AccuRev control. If you have no Eclipse projects associated with AccuRev, you must log in to AccuRev directly.

- In AccuRev Plug-In, using the **Team** > **Share Project** > **AccuRev** menu. When you share a project with AccuRev, you are prompted to log in to AccuRev if you are not logged in already.

In all cases, when you log in you are required to specify the AccuRev Server, and your AccuRev user name and password, as shown in the following illustration of the login dialog box for the AccuRev GUI:

Note: Logging in to a different server directly through AccuRev does not affect your existing connection in Eclipse. For example, if you are currently logged in to the AccuRev Alpha server, you can continue working with Eclipse projects that are under AccuRev control on that server even if you log in to the AccuRev Beta server directly through AccuRev.

Setting AccuRev Preferences in Eclipse

The AccuRev page in the Eclipse Preferences window lets you specify general AccuRev settings such as which Diff and Merge tools you want AccuRev to use, and which label decorations you want to use for files under AccuRev control.

To open the Preferences window:

1. Place the cursor in the AccuRev Console window in Eclipse or select **Window** > **View** > **AccuRev Console**.

2. Select **Window** > **Preferences** > **Team** > **AccuRev**.
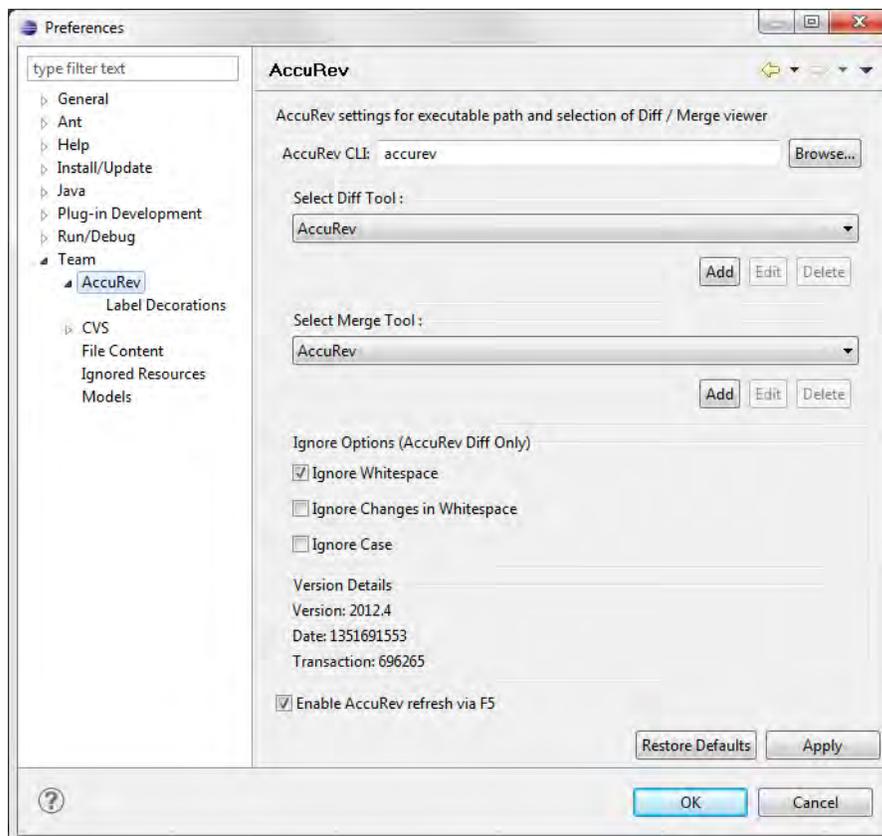
The procedures described in this section describe how to:

- *Change the AccuRev executable path and disable the F5 (Refresh) key*

- *Specify third-party Diff and Merge tools*

- *Specify label decorations*

General AccuRev Settings

Use the procedure described in this section to:

- **Change the path of the AccuRev executable.** By default, AccuRev Plug-In looks for the AccuRev executable (accurev.exe) in the default installation location. If you customized the AccuRev installation, you might need to specify the location of the AccuRev executable here.

- **Use F5 to perform Eclipse refresh.** By default, pressing the F5 key performs the AccuRev refresh operation, which updates the status of elements under AccuRev control displayed in the Navigator view. If you want F5 to perform the Eclipse refresh operation, which refreshes Eclipse project information only, clear the **Enable AccuRev refresh via F5** checkbox. See *Refresh* on page 48 for more information.

See *Using Alternative Diff and Merge Tools* on page 3 to learn how to configure supported and custom third party Diff and Merge tools.

To modify general AccuRev settings:

1. In Eclipse, open the Preferences window (**Window** > **Preferences**).

2. In the Preferences window, navigate to **Team** > **AccuRev**.

3. Change the settings as desired and click **OK**.

Using Alternative Diff and Merge Tools

By default, AccuRev uses its own tools to diff and merge files. If you want, you can configure:

- One of several supported third party Diff and Merge tools:

Vendor	Diff Tool Support	Merge Tool Support
Araxis	Yes	Yes
BeyondCompare	Yes	No
Eclipse	Yes	Yes
Guiffy	Yes	Yes
TkDiff	Yes	Yes

AccuRev provides preformatted command parameters for supported third party Diff and Merge tools. To use one of these tools, all you need to do is specify the location of the tool's executable when you configure it. See *Configuring a Supported Third Party Diff or Merge Tool* on page 4 for more information.

- A custom Diff or Merge tool. In the case of custom Diff and Merge tools, you need to specify the complete command parameters in addition to the executable's location. See *Command Parameters for Diff and Merge Tools* on page 3 for more information.

Command Parameters for Diff and Merge Tools

The following table summarizes the command parameters supported by AccuRev Plug-In.

Tool	Command Parameter	Description
Diff	`{backing stream version}`	Quoted pathname of the first version to be compared
	`{workspace version}`	Quoted pathname of the second version to be compared

Tool	Command Parameter	Description
Merge	{common ancestor version}	Quoted filename of the closest common ancestor version
	{backing stream version}	Quoted filename of the version in the backing stream, or other non-workspace version
	{workspace version}	Quoted filename of the version in the workspace
	{merge results}	Quoted name of the merge-output file – a temporary file for storing the results of the merge

These command parameters are preformatted for supported third party Diff and Merge tools. If you are configuring a custom Diff or Merge tool, you must ensure that the command parameters you specify are correct for that tool. For example, if you were configuring SourceGear DiffMerge, you might configure the Diff tool as follows:

```
/c="AccuRev Diff" /ro2 /t1="First" /t2="Second" {backing stream version}
{workspace version}
```

Similarly, the Merge tool might be configured as follows:

```
/c="AccuRev Diff" -merge -result={merge results} /t1="First" /t2="Second" /t3="Third"
{backing stream version} {workspace version} {common ancestor version}
```

The executable for both Diff and Merge tools might be defined as follows:

```
C:\Program Files\SourceGear\DiffMerge\DiffMerge.exe
```

Specific command parameters vary from tool to tool. See your Diff and Merge tool's documentation for more information.

Configuring a Supported Third Party Diff or Merge Tool

To configure a third party Diff or Merge tool supported by AccuRev:

1. Display the AccuRev page on the Eclipse Preference window by selecting **Window** > **Preferences** > **Team** > **AccuRev** from the Eclipse menu.

2. Select the supported third party tool from the **Select Diff Tool** or **Select Merge Tool** field.

3. Click the **Edit** button.

   The Add/Edit Tool dialog box appears. The dialog box for editing a Diff tool is shown here.
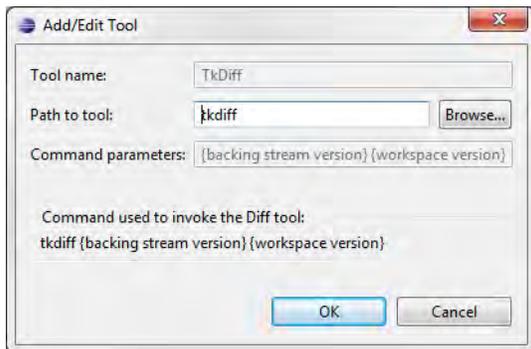
   Values in the **Tool Name** and **Path to Tool** fields are based on the tool you selected in *Step 2*.

4. If the third party tool executable is not on your system's search path, specify the complete path in the **Path to Tool** field.

5. Review the command line that will be used to invoke the third party tool.

   AccuRev displays the command as it will be executed in the **Command used to invoke the Diff tool** (or **Command used to invoke the Diff tool**) field.

6. Click **OK**.

   The Add/Edit Tool dialog box closes.

7. Click **Apply** if you want to change other AccuRev preferences. Otherwise, click **OK**.

Configuring a Custom Diff or Merge Tool

To configure a custom Diff or Merge tool:

1. Display the AccuRev page on the Eclipse Preference window by selecting **Window** > **Preferences** > **Team** > **AccuRev** from the Eclipse menu.

2. Click the **Add** button associated with the custom Diff or Merge tool you want to configure.

   The Add/Edit Tool dialog box appears. The dialog box for adding a Diff tool is shown here.
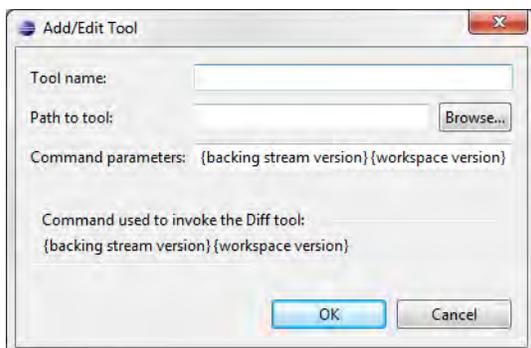
3. Enter values in the **Tool Name**, **Path to Tool**, and **Command Parameters** fields appropriate for the custom Diff or Merge tool you are configuring. See *Command Parameters for Diff and Merge Tools* on page 3 if you need help with this step.

AccuRev displays the command as it will be executed in the **Command used to invoke the Diff tool** field (or **Command used to invoke the Diff tool** field).

4. Review the command line that will be used to invoke the custom third party tool.

5. Click **OK**.

   The Add/Edit Tool dialog box closes.

6. Click **Apply** if you want to change other AccuRev preferences. Otherwise, click **OK**.

Setting Label Decorations

In AccuRev, each element has an AccuRev status, which is expressed using one or more status indicators-- or example, **(kept)(member)**. In AccuRev Plug-In, an element's AccuRev status is represented using *label decorations*. Label decorations can be optionally displayed in the Navigator and Package Explorer views; they are always displayed in the Synchronize view.

Types of Label Decorations

You can specify one of two types of label decorations:

• **Icon decoration** – a colored symbol displayed in the lower-right corner of an object's file-type icon. Icon decorations are displayed by default, but you can turn them off if you want.

• **Name decoration** – one or more characters appended (or prefixed) to an object's name in the Navigator display. Name decorations are not displayed by default; you can turn them on and, optionally, change the characters used to represent the status.

The following table shows the AccuRev status indicators and the label decorations that can be used to represent them.

Status	Icon Decoration	Default Name Decoration
(backed)	green computer	^
(member)	blue dot	>
(modified)	white wrench	*
(missing)	red circle	->
(overlap)	white exclamation mark	<>
(stale)	white arrow	<

AccuRev status indicators are not mutually exclusive--for example, an element's status can include both the **(modified)** and **(member)** indicators. But Eclipse uses only one decoration on each icon and/or each filename. The precedence order of the AccuRev statuses determines which decoration is displayed; statuses are listed in precedence order from highest to lowest in the table.

Folder and Project Decorations

The following table shows the AccuRev folder indicators.

Decoration	Description	Indicates
	Folder marked with an asterisk icon	One or more of the elements in that folder has one of the following statuses: ▪ **(external)** ▪ **(modified)** ▪ **(overlap)** ▪ **(member)**
	Project marked with a yellow database icon	The project is shared and you are logged into AccuRev.
	Project marked with a grey database icon	The project is shared and you are not logged into AccuRev.
	Project not marked with an AccuRev decoration	The project is not shared with AccuRev.

Turning Off Label Decorations

Label decorations for AccuRev elements are displayed by default. Use this procedure to turn them off.

To turn off label decorations:

1. In Eclipse, open the Preferences window (**Window** > **Preferences**).

2. In the Preferences window, navigate to **General** > **Appearance** > **Label Decorations**.
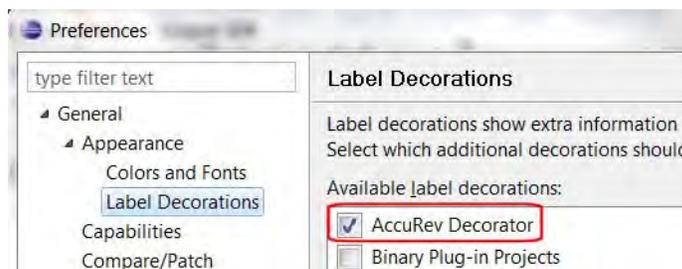
3. Clear the **AccuRev Decorator** check box to turn off label decorations for AccuRev elements in the Navigator view.

4. Click **OK**.

Displaying Name Decorations

By default, only icon decorations are displayed in Eclipse. Use the following procedure to display name decorations and, optionally, change the characters used for the decoration.

Note: You can perform this procedure only on an existing Eclipse project.

1. In Eclipse, open the Preferences window (**Window** > **Preferences**).

2. In the Preferences window, navigate to **Team** > **AccuRev** > **Label Decorations**.

3. Select the **Enable text labels** check box.

4. By default, name decorations are prefixed to element names. Select **Append** if you want name decorations added to the end of element names.

5. Optionally, change the characters used to represent AccuRev statuses.

    For more on label decorations, see *Setting Label Decorations* on page 6.

6. Click **OK**.

Executing AccuRev Commands

There are several ways to access AccuRev functionality within Eclipse:

*   **Integrated commands** – Some AccuRev functionality is integrated with existing Eclipse commands on the context menu. See *Integrated Commands* on page 35 for more information.

*   **Team menu** – Many AccuRev commands are available in the **Team** menu. See *Team Menu Commands* on page 38 for more information.

*   **Web UI** – Some AccuRev functionality is accessed from the AccuRev Web Interface (Web UI), which is also started from the context menu. See *AccuRev Web UI* on page 49 for more information.

*   **Toolbar** – A commonly-used subset AccuRev features, for both the AccuRev GUI and Web UI, is available on the Eclipse toolbar. See *Toolbar Access to AccuRev Commands* on page 51 for more information.

Note that while Eclipse allows you to select objects from multiple projects, AccuRev commands are available only if all the selected objects belong to the same AccuRev workspace.

Output from AccuRev Commands

AccuRev uses custom Console and Search/Status views to display output from AccuRev commands. These views, along with Eclipse's own Synchronize and History views, constitute the AccuRev perspective. For a description of these views, see *Working in the AccuRev Perspective* on page 57.

What to Do Next

Once you have configured your preferences, you are ready to start working with Eclipse projects using AccuRev Plug-In. See *Chapter 2 Working with Eclipse Projects* for more information.

2. Working with Eclipse Projects

AccuRev commands allow data to be managed within the central source-code repository (called a *depot* in AccuRev) and your personal work area (called a *workspace* in AccuRev, and a *project* in Eclipse).

Techniques for Associating Eclipse Projects with AccuRev

There are several techniques for associating Eclipse projects with AccuRev. The following table summarizes these techniques; they are described in detail within this chapter.

Eclipse/AccuRev Association Technique	Description	For more information
Create an Eclipse project in an existing AccuRev workspace	Creates a new Eclipse project and places it in the workspace you choose. Two options are available: • A wizard can be used to create a new project and a new .project file is created by Eclipse. • If the workspace already contains a .project file, that file is used to create the Eclipse project.	See *Creating an Eclipse Project in an Existing AccuRev Workspace* on page 10.
Create a single Eclipse project in a new AccuRev workspace	Creates a new Eclipse project and a new AccuRev workspace and places the project in the workspace. All files from in the AccuRev stream you specify are placed into the new workspace. Two options are available: • A wizard can be used to create a new project and a new .project file is created by Eclipse. • If a .project file is already within the stream, that file is used to create the Eclipse project.	See *Creating a Single Eclipse Project in a New AccuRev Workspace* on page 14.
Create multiple Eclipse projects in new AccuRev workspaces	Creates a new Eclipse project for each folder you select and places the files from each of them in new AccuRev workspaces. A new .project file is automatically created for each Eclipse project.	See *Creating Multiple Eclipse Projects in a New AccuRev Workspace* on page 20.
Share an Eclipse project with AccuRev	Creates an internal Eclipse association (mapping) of an Eclipse project to AccuRev.	See *Sharing an Eclipse Project with AccuRev* on page 24.

Note: Associating an Eclipse project with AccuRev does not copy any data into your Eclipse project folder. The data is located in the AccuRev workspace, and simply becomes accessible through the Eclipse project.

Creating an Eclipse Project in an Existing AccuRev Workspace

Use this procedure to create an Eclipse project in an existing AccuRev workspace.

1. Click the **New** button ( ) on the Eclipse toolbar.

   Alternative: Select **File** > **New** > **Project** from the Eclipse main menu or select **File** > **Import** from the Eclipse main menu.

   The New Project dialog box appears.
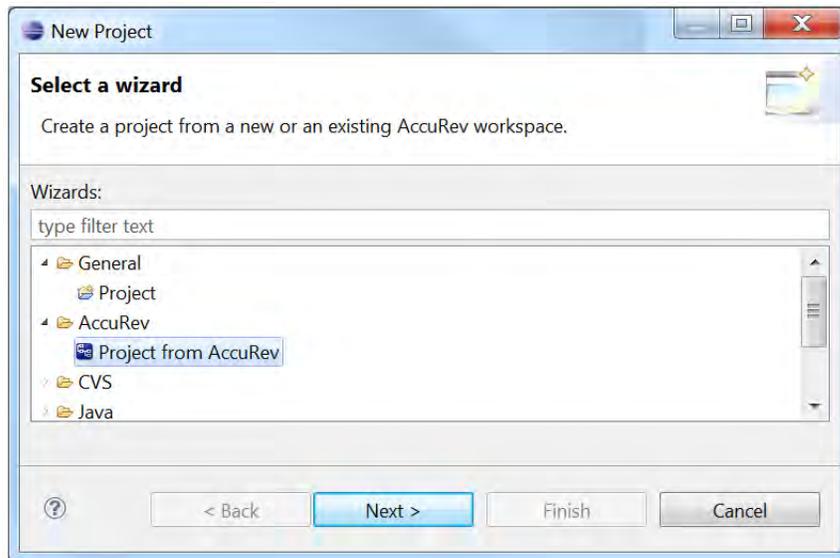
2. In the **Wizards** field, navigate to **AccuRev** > **Project from AccuRev**. (If you have many wizards, simply type **A** in the **Wizards** field to go directly to AccuRev in the list.)

3. Click **Next**.

   Note: If you are not already logged in to AccuRev, you are prompted to do so. See *Logging In to AccuRev* on page 1.

   The Checkout from AccuRev page appears.
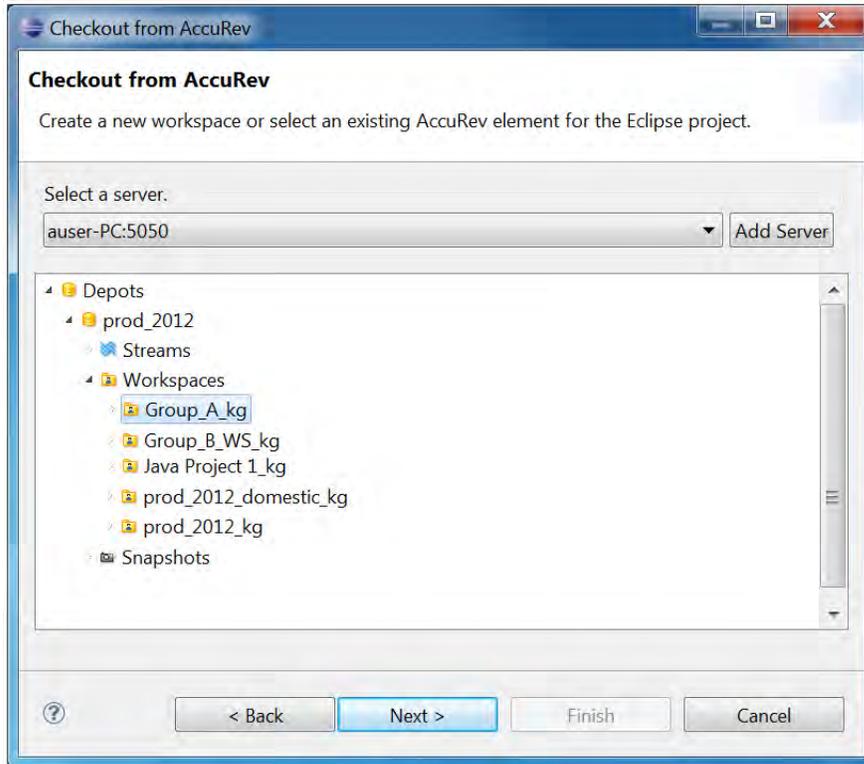
4. Select a server where your AccuRev files are stored if the correct server is not already displayed in the **Select a server** field. If you need to add a server, see *Adding an AccuRev Server* on page 28.

5. Expand the Workspaces folder.

6. Select the AccuRev workspace where you want the new Eclipse project to be placed.

7. Click **Next**.

The Check Out As dialog box appears and displays two checkout options:

- **Check out as a project configured using the New Project Wizard** -- This option allows you to create a new Eclipse project file using the New Project Wizard. Eclipse creates a new .project file.

- **Check out in the workspace as projects** -- This option allows you to use an existing .project file to create the new project. This option is selected by default if a .project file is detected in the workspace you selected in *Step 6* on page 11.

  Note: The .project file must have an AccuRev status of (**kept**) in order to be detected.
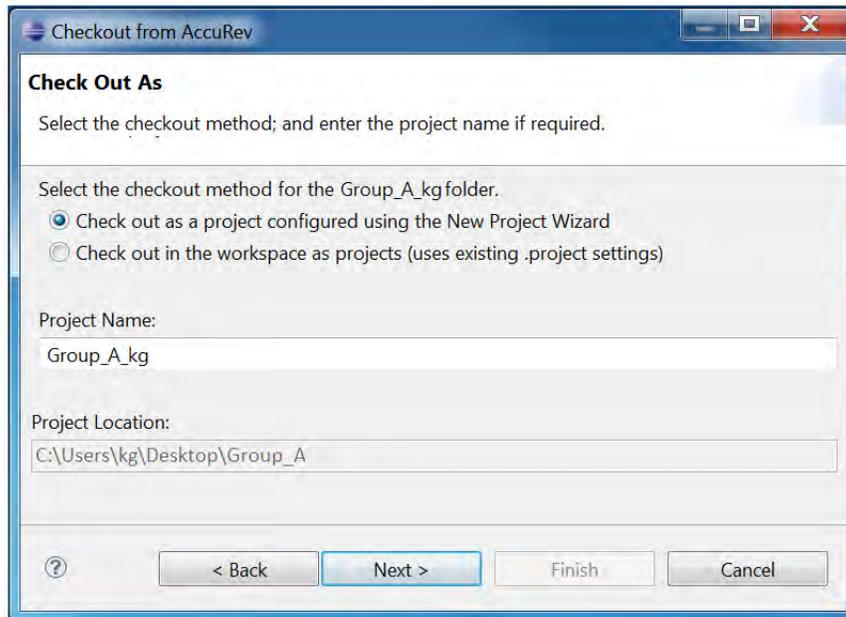
8. Select the project checkout option you want to use.

9. Complete the **Project Name** field or leave it as displayed:

- If you selected the **Check out as a project configured using the New Project Wizard** option in *Step 8*, the AccuRev workspace name is used as the default Project Name. You can change this name or leave the default.

- If you selected the **Check out in the workspace as projects** option in *Step 8*, the Project Name is taken from the existing .project file. This name cannot be changed and the **Project Name** field is disabled. (See *How New Eclipse Project Names are Taken from Existing .project Files* on page 14 for more information.)

  Note: The **Project Location** field displays the workspace path you selected in *Step 6* on page 11.

10. Click **Finish** or **Next** as appropriate:

- If you selected the **Check out in the workspace as projects** option in *Step 8*, the **Finish** button is active. Click **Finish**. A "Creating Eclipse Projects" message appears and you are done. A new Eclipse project is created.

- If you selected the **Check out as a project configured using the New Project Wizard** option in *Step 8*, the **Next** button is active. Click **Next**. The New Eclipse Project Type Selection page appears.
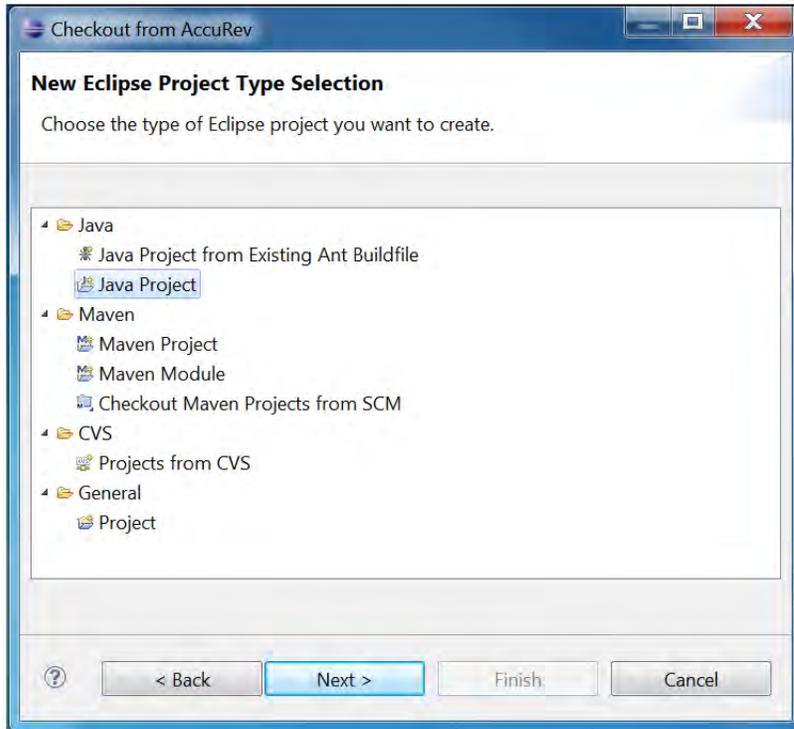
11. Expand the project category you want to create, and then select a specific project type.

   Note: The list of project types varies based on the Eclipse plug-ins you have installed.

12. Click **Next**.

   The first page of the configuration wizard for the project type you selected appears.

   Note: The wizard pages displayed from this point forward vary based on the project type you selected in *Step 11*.

13. Complete the remaining wizard pages for the project type and click **Finish**.

   A new Eclipse project is created.

How New Eclipse Project Names are Taken from Existing .project Files

Existing .project files contain project names in the <**name**> tag within the file. If you use an existing .project file to create a new project, the name of the new Eclipse project is taken from the <**name**> line in the .project file as shown in the following illustration.
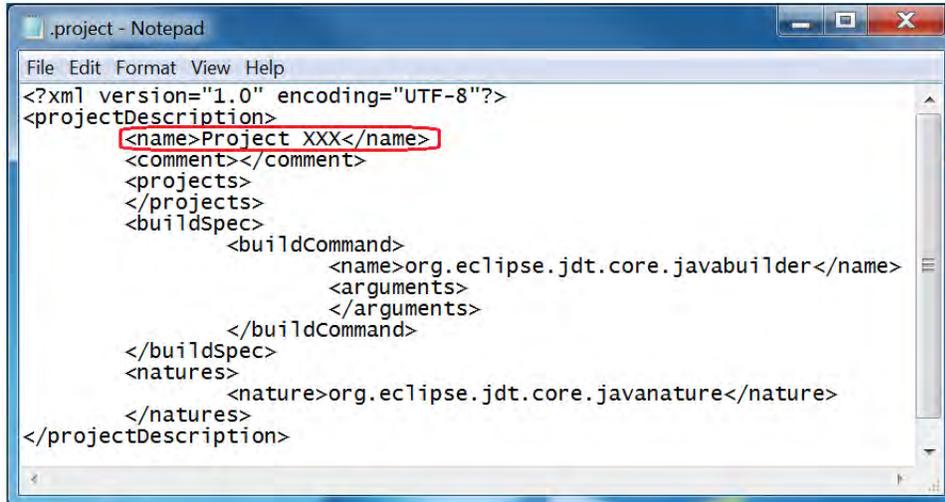
Creating Eclipse Projects in a New AccuRev Workspace

Use the procedures in this section to create either a single new Eclipse project or multiple new Eclipse projects and place them in a new AccuRev workspace. You can:

- **Create a Single Eclipse Project in a New AccuRev Workspace** -- This procedure allows you to create a new Eclipse project and place all files from a specified stream into a new AccuRev workspace.

- **Create Multiple Eclipse Projects in a New AccuRev Workspace** -- This procedure allows you to create a new Eclipse project for each folder that you select within a stream. All the files within the selected folders are placed into one new AccuRev workspace.

Creating a Single Eclipse Project in a New AccuRev Workspace

Use this procedure to create a new project and place all the files from an AccuRev stream into a new AccuRev workspace.

1. Click the **New** button ( ) on the Eclipse toolbar.

   Alternative: Select **File** > **New** > **Project** from the Eclipse main menu or select **File** > **Import** from the Eclipse main menu.

   The New Project dialog box appears.

2. In the **Wizards** field, navigate to **AccuRev** > **Project from AccuRev**. (If you have many wizards, simply type **A** to go directly to AccuRev in the list.)

3. Click **Next**.

    Note: If you are not already logged in to AccuRev, you are prompted to do so. See *Logging In to AccuRev* on page 1.

    The Checkout from AccuRev page appears.

4. Select a server where your AccuRev files are stored if the correct server is not already selected. If you need to add a server, see *Adding an AccuRev Server* on page 28.

Note: Press F5 to refresh the view.

5. Expand the Streams folder.

6. Navigate to the stream off of which you want the new AccuRev workspace to be located.

7. Select a stream.

8. Click **Next**.

   The Create New Workspace page appears.
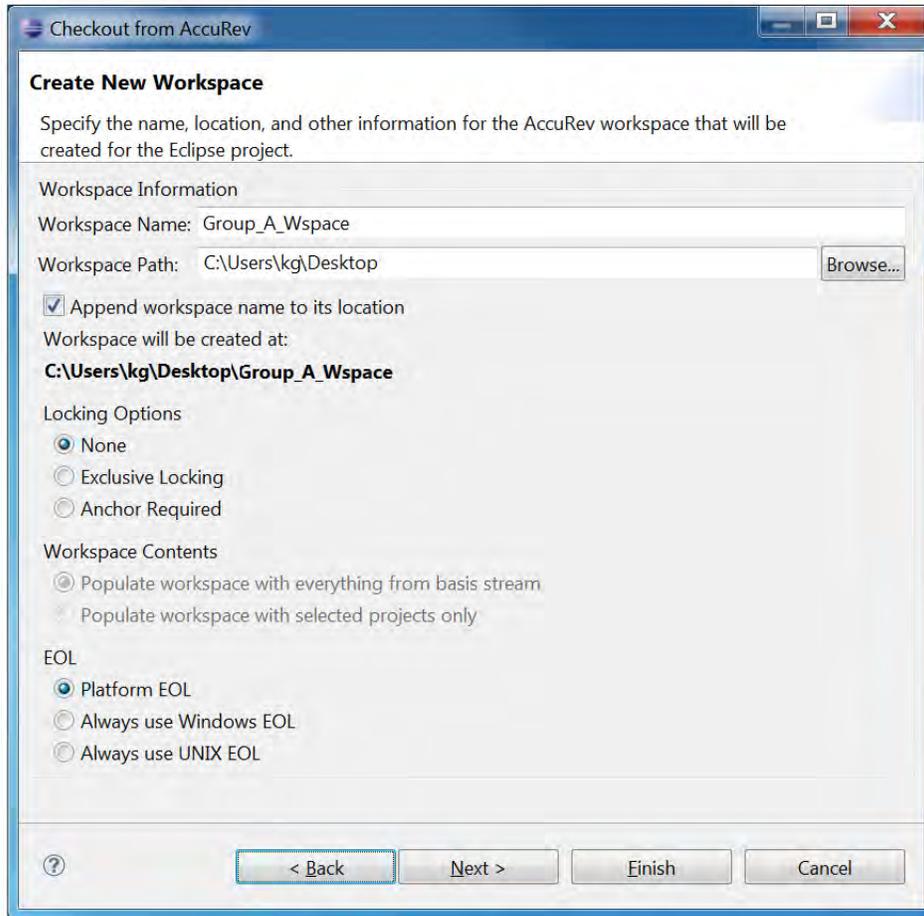
9. In the **Workspace Name** field, enter a name for the new AccuRev workspace.

   Note: When the workspace is created in AccuRev, the suffix *_username* is appended to the workspace name that you enter in this field. For example, if your AccuRev username is *jsmith* and you enter *W1* as the workspace name, AccuRev creates the workspace name as *W1_jsmith*.

10. In the **Workspace Path** field, enter or browse to the path (containing no other workspaces) where the new AccuRev workspace will be created.
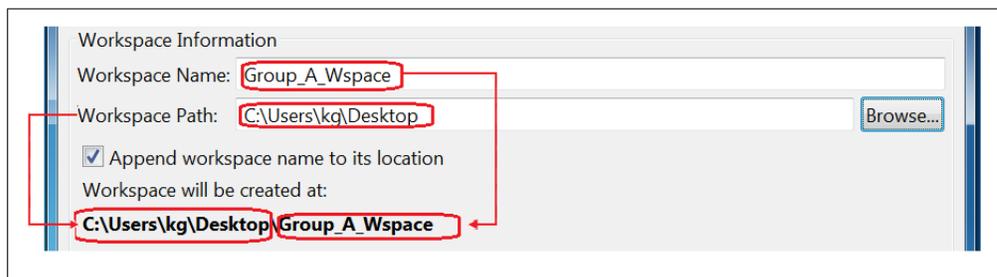
   Note: The **Workspace Path** must be a unique location that contains no other workspaces, and AccuRev displays a message if it is not unique. There are several ways to try to make the Workspace Path unique:

- In the **Workspace Path** field, enter a unique name of a directory to be created. This new directory will be the location of the new workspace.

- Click the **Append workspace name to its location** box. In this way, a new directory will be created based on the workspace name. For example, if the **Workspace Path** you enter is *C:\Workspaces* and the **Workspace Name** you enter is *NEW_WS*, then a new directory will be created named *C:\Workspaces\NEW_WS*. This new directory will be the location of a workspace named *NEW_WS*. (Make sure there is not already a workspace of that name already within the selected stream.)

- Create a unique directory to contain the workspace and browse to this in the **Workspace Path** field.

11. Optionally change the default values for settings for the new AccuRev workspace: locking options, workspace contents, and end-of-line (EOL) convention.

    Note: If you choose either Exclusive Locking or Anchor Required locking options, you will need to perform an **Anchor** command on a file before editing it. See *Anchor* on page 40 for more information.

12. Click **Next**.

    The Check Out As page appears and displays two checkout options:

    - **Check out as a project configured using the New Project Wizard** -- This option allows you to create a new .project file using the New Project Wizard.

    - **Check out in the workspace as projects** -- This option allows you to use an existing .project file to create the new project. This option is selected by default if a .project file is detected in the selected stream.

        Note: The .project file must have an AccuRev status of (**kept**) in order to be detected.
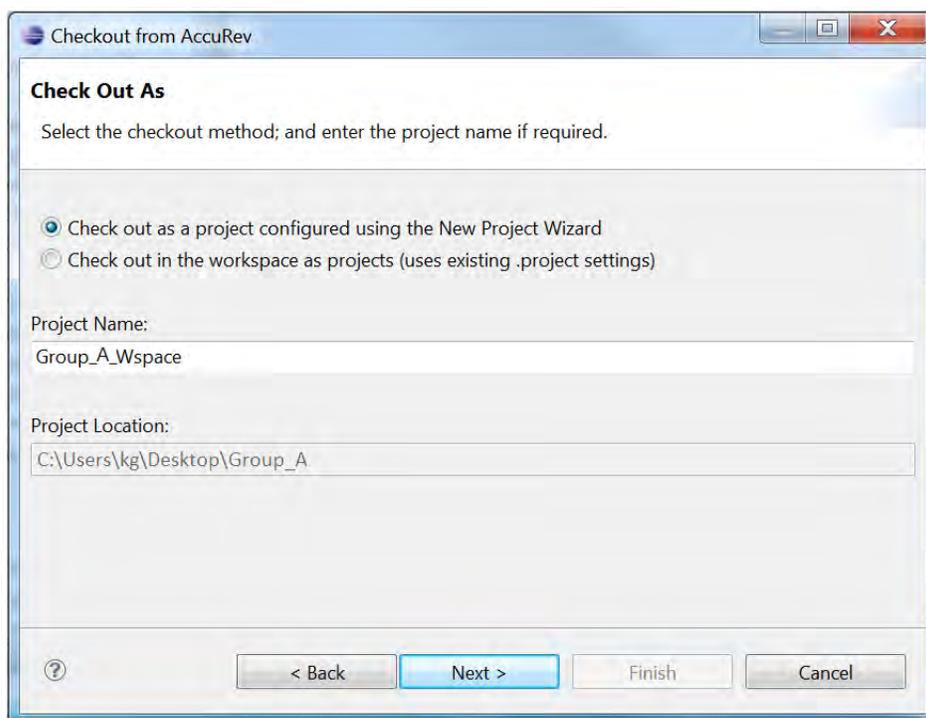
13. Select the project checkout option you want to use.

14. Complete the **Project Name** field or leave it as displayed:

    - If you selected the **Check out as a project configured using the New Project Wizard** option in *Step 12*, the AccuRev workspace name displays as the default Project Name. You can change this name or leave the default.

    - If you selected the **Check out in the workspace as projects** option in *Step 12*, the Project Name is taken from the existing .project file. This name cannot be changed and the **Project Name** field is disabled. (See *How New Eclipse Project Names are Taken from Existing .project Files* on page 14 for more information.)

    Note: The **Project Location** field displays the workspace path you selected in *Step 10* on page 17.

15. Click **Finish** or **Next** as appropriate:

    - If you selected the **Check out in the workspace as projects** option in *Step 12*, the **Finish** button is active. Click **Finish**. A "Creating Eclipse Projects" message appears and you are done. A new Eclipse project is created.

    - If you selected the **Check out as a project configured using the New Project Wizard** option in *Step 12* on page 18, the **Next** button is active. Click **Next**. The New Eclipse Project Type Selection page appears.
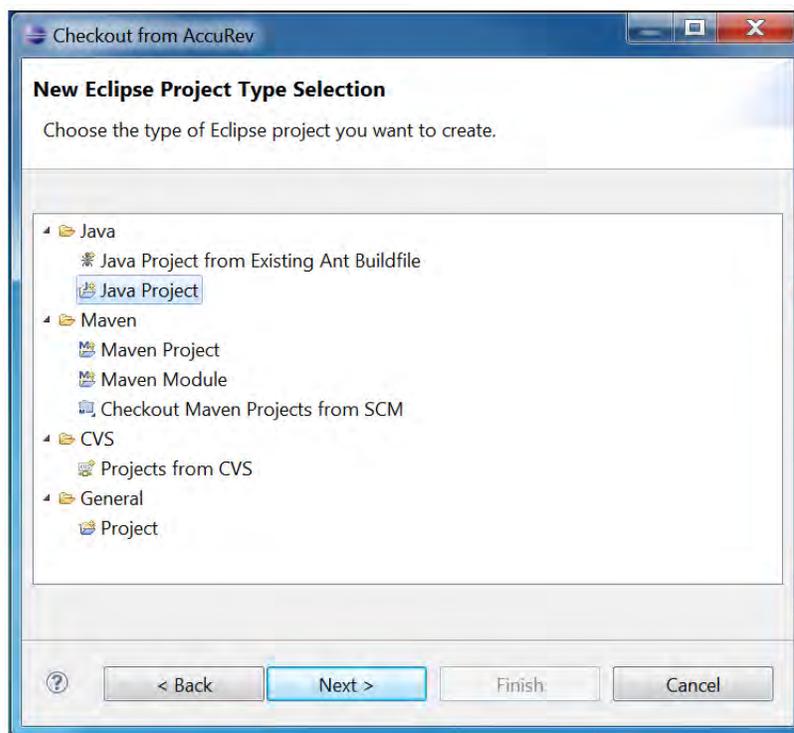
    Note: The list of project types varies based on the Eclipse plug-ins you have installed.

16. Expand the project category you want to create, and then select a specific project type.

17. Click **Next**.

    The first page of the configuration wizard for the project type you selected appears.

    Note: The wizard pages displayed from this point forward vary based on the project type you selected in *Step 16*.

18. Complete the remaining wizard pages for the project type and click **Finish**.

A new Eclipse project is created.

Creating Multiple Eclipse Projects in a New AccuRev Workspace

Use this procedure to create a separate Eclipse project for each AccuRev folder that you select, and place all the files from the folders into a new AccuRev workspace. This procedure is useful if you have an AccuRev stream that has a large number of folders and you only need to place the files from a few specific folders in the new AccuRev workspace.

1. Click the **New** button ( ) on the Eclipse toolbar.

   Alternative: Select **File** > **New** > **Project** from the Eclipse main menu or select **File** > **Import** from the Eclipse main menu.

   The New Project dialog box appears.

2. In the **Wizards** field, navigate to **AccuRev** > **Project from AccuRev**. (If you have many wizards, simply type **A** to go directly to AccuRev in the list.)

3. Click **Next**.

   Note: If you are not already logged in to AccuRev, you are prompted to do so. See *Logging In to AccuRev* on page 1.

   The Checkout from AccuRev page appears.

4. Select a server where your AccuRev files are stored if the correct server is not already selected. If you need to add a server, see *Adding an AccuRev Server* on page 28.

5. Expand the Streams folder.

AccuRev® Plug-In for Eclipse User's Guide

Tip: Place the focus on the tree view and then press F5 if you want to refresh the view.

6. Expand the stream where you want the new AccuRev workspaces to be created.

7. Select one or more folders within the stream. (To select multiple folders, hold down the Ctrl key or Shift key and then click the relevant folders.)

8. Click **Next**.

The Create New Workspace page appears.
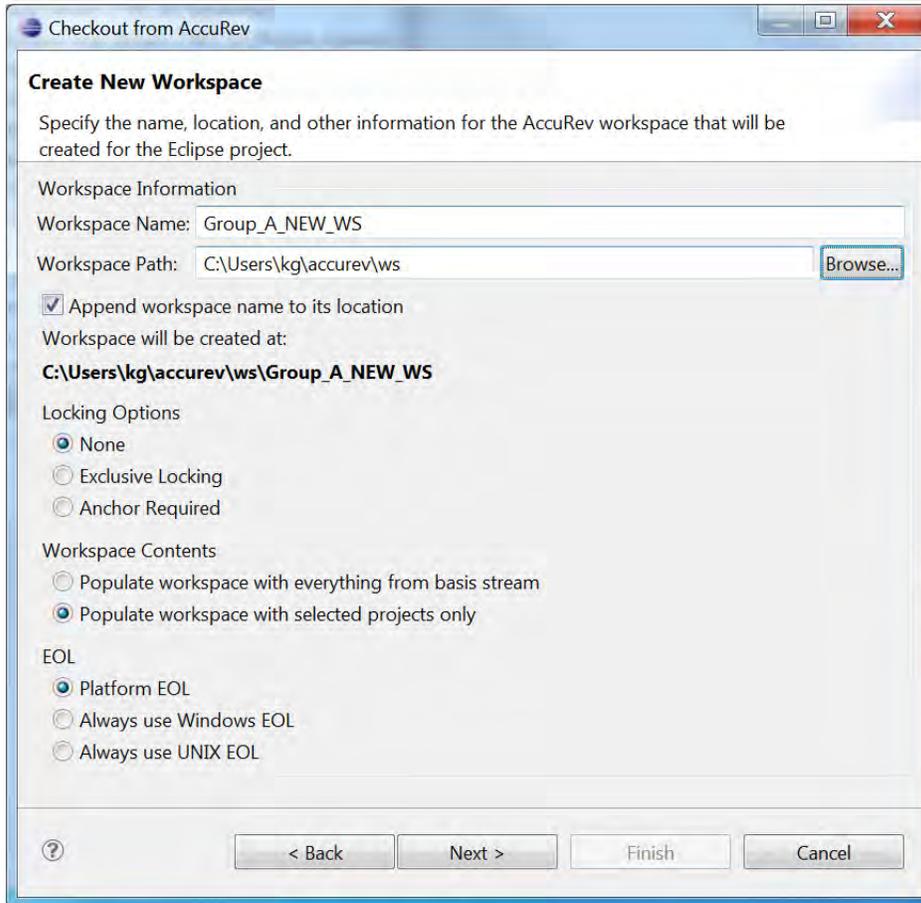
9. In the **Workspace Name** field, enter a name for the new AccuRev workspace.

    Note: When the workspace is created in AccuRev, the suffix *_username* is appended to the workspace name that you enter in this field. For example, if your AccuRev username is *jsmith* and you enter *W1* as the workspace name, AccuRev creates the workspace name as *W1_jsmith*

10. In the **Workspace Path** field, click the **Browse** button and select an existing directory.

    A new directory is created based on the **Workspace Name** and **Workspace Path** you enter. For example, if you specify *c:\workspaces* as the workspace path, *W1* as the workspace name, and your username is *jsmith*, AccuRev creates the directory *c:\workspaces\W1* and this directory contains the workspace named *W1_jsmith.*

11. Optionally change the default values for settings for the new AccuRev workspace: locking options, workspace contents, and end-of-line (EOL) convention.

    Note: If you choose either Exclusive Locking or Anchor Required locking options, you will need to perform an **Anchor** command on a file before editing it. See *Anchor* on page 40 for more information.

12. Select the **Workspace Contents** option based on which files from the stream are to be included in the workspace. The following options are available:

    •   **Populate workspace with everything from basis stream** -- This option places all the files from the basis stream in the workspace and does not provide any AccuRev include rules. Use this option if you want to include all files from the stream in the project. (This option may take a while if you have a very large stream.)

- **Populate workspace with selected projects only** -- This choice updates the workspace with only the files from the folders you selected in *Step 7* on page 21. (AccuRev include rules are used to find the files belonging to those folders only. The other files in the basis stream will be excluded.) This is the default choice. Use this option if you only want to include only the files of the selected folders in your project.

13. Click **Next**.

    The Check Out As page appears. This page allows you to name the new Eclipse projects you are creating based on the names of the folders you selected.

    This page appears only if you selected more than one folder in step *Step 7*.
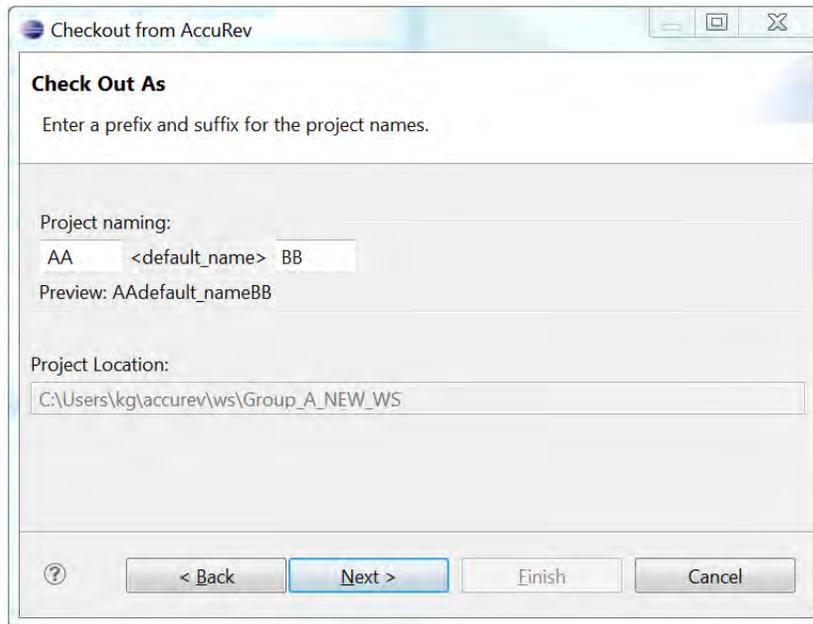
14. Optionally enter a prefix and a suffix to attach to the name of each folder. For example, if the prefix is AA and the suffix is BB, a folder named Folder_1 will result in an Eclipse project named AAFolder_1BB.

    Note: You can also leave the prefix and/or suffix fields empty, if desired.

15. Click **Next**.

16. The Checkout Projects page appears. A new project is listed for each folder you selected in *Step 7* on page 21.

    The new projects in the list have project names and project types as follows:

    - If no .project file is detected in the folder, the project name is based on the prefix and suffix you provided in *Step 14* (if any); the new project has the project type of Eclipse Project.

    - If a .project file is detected in the folder, the project name is taken from the <**name**> tag within the .project file and the Project Type is Defined Type. (See *How New Eclipse Project Names are Taken from Existing .project Files* on page 14 for more information)
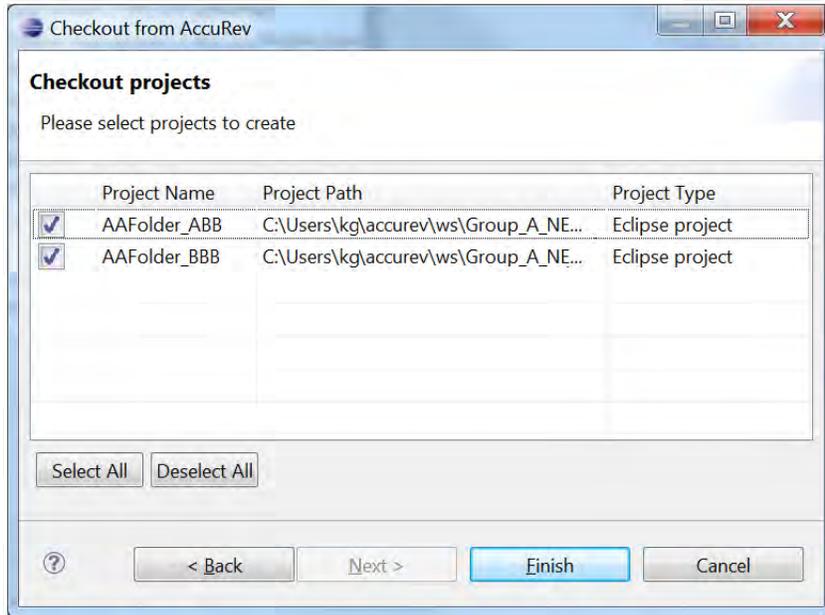
17. De-select any project that you do not want to create.

18. Click **Finish**.

An Eclipse project is created for each Eclipse project you selected.

Sharing an Eclipse Project with AccuRev

Sharing is the process of placing files associated with an Eclipse project under AccuRev control. You might want to share a project with AccuRev to use the configuration management capabilities of AccuRev for a project that already exists in Eclipse.

Note: Project sharing functionality is available for Eclipse 3.4 and later.

How Shared Eclipse Project Files Populate AccuRev Workspaces

The Sharing process places files from Eclipse projects in an AccuRev workspace in a way that is consistent with how the AccuRev workspace was initially created.

If your Eclipse project is not located within an AccuRev workspace prior to sharing, then a workspace is created for you during the Sharing process and you choose which files you want to place in the workspace. The new workspace is initially populated with only the files you select. Note that the basis stream of the workspace might include other files and an AccuRev **Update** command will populate the new workspace with all the files from the basis stream.

If your Eclipse project is already located within a AccuRev workspace, then the Sharing process depends on how the AccuRev workspace was initially created:

- If your Eclipse project is located within an AccuRev workspace that was created with the "Initial Contents – Same as Basis Stream" option, then *all* the files from the Eclipse project are promoted to the workspaces's basis stream during the Sharing process.

- If your Eclipse project is located within an AccuRev workspace that was created with the "Initial Contents – Pick and choose from Basis stream" option, then you choose the files from the Eclipse project that you want to promote to the basis stream of the workspace.

To share an Eclipse project with AccuRev:

1. In the Eclipse Navigator or Project Explorer view, right-click the project you want to share and select **Team** > **Share Project**.

   The Share Project dialog box appears.
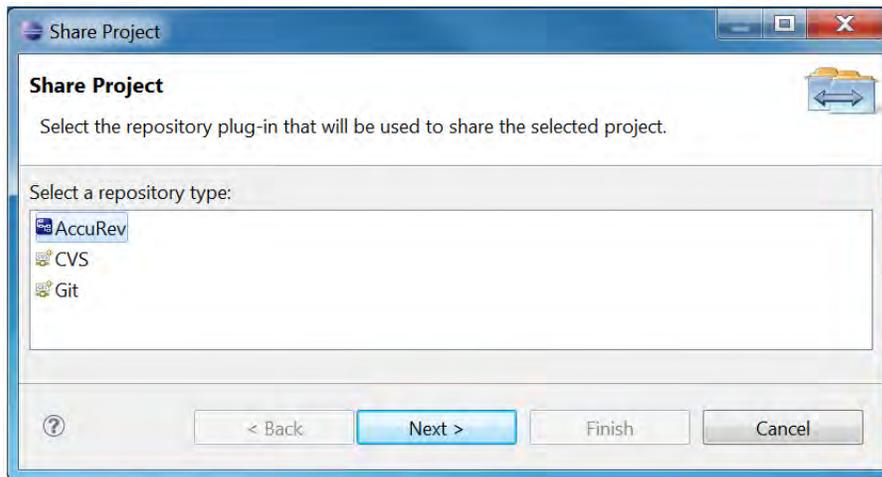
2. Select **AccuRev** and click **Next**.

   The Share Project with AccuRev page appears.
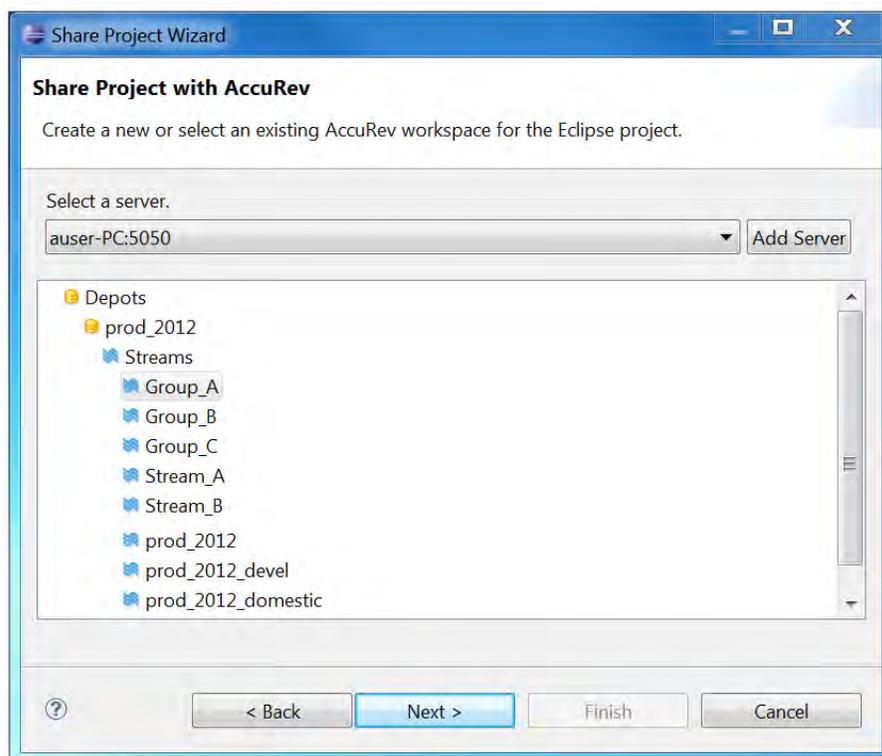
Tip: Place the focus on the tree view and then press F5 if you want to refresh the view.

3. Select the AccuRev stream in which you want to create the new workspace for your Eclipse project.

   Note: This stream should not contain any files that have the same filenames as files within the Eclipse project that you are sharing.

4. Click **Next**.

   The Create New Workspace page appears. The default workspace name and workspace path are taken from the Eclipse project.
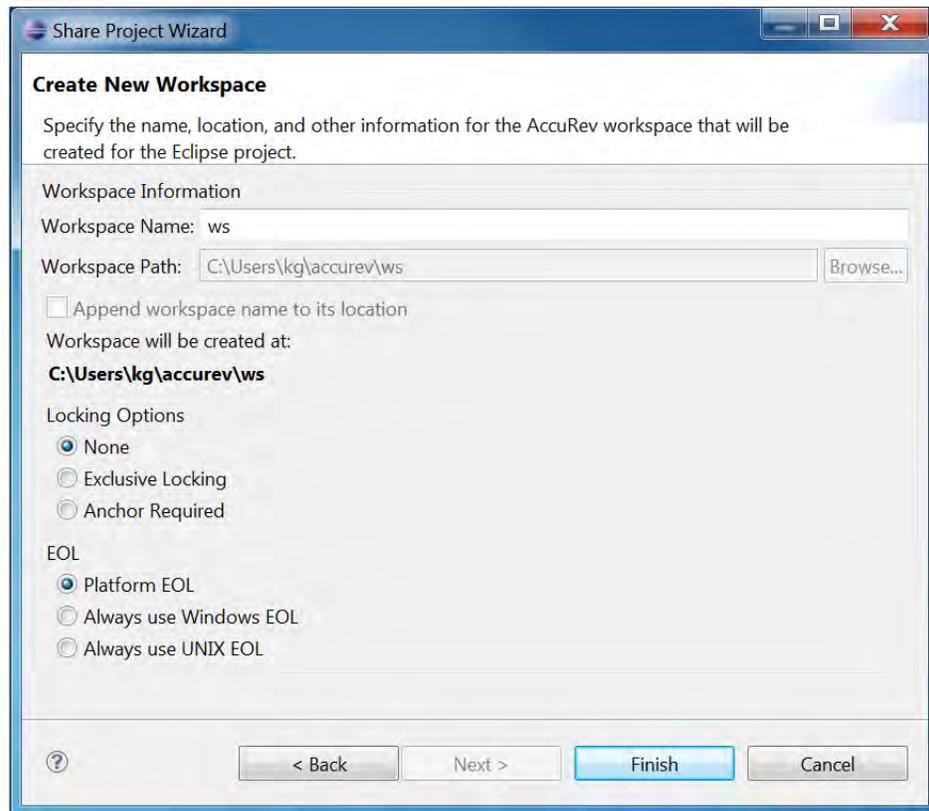
5. Optionally, change the default values for locking options and end-of-line (EOL) convention for the new workspace. If you choose either Exclusive Locking or Anchor Required locking options, you will need to perform an **Anchor** command on a file before editing it. See *Anchor* on page 40 for more information.

6. Click **Finish**.

   The workspace is created and the Promote dialog box appears. This dialog box allows you to promote the files in the shared project into the AccuRev depot.

   Note: See *Promote* on page 40 for more information about Promote.
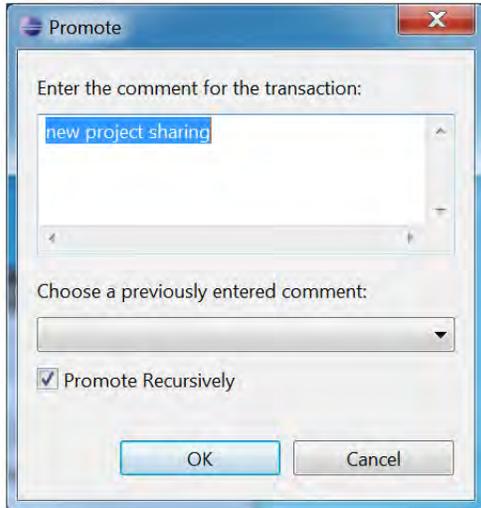
7. Enter a comment or select a comment from the **Choose a previously entered comment** box. Leave the **Promote Recursively** box checked, and click **OK**.

    The External Elements to Add and Promote dialog box appears with all elements selected.

8. Optionally de-select the elements that you do not want to add to the workspace.
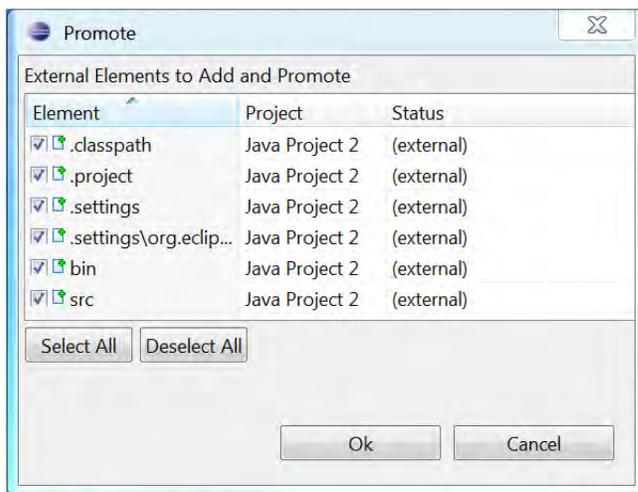
9. Click **OK**.

   If you do not have Change Packages implemented, the elements are promoted. If you have Change Packages implemented, the Select Issue dialog box appears.
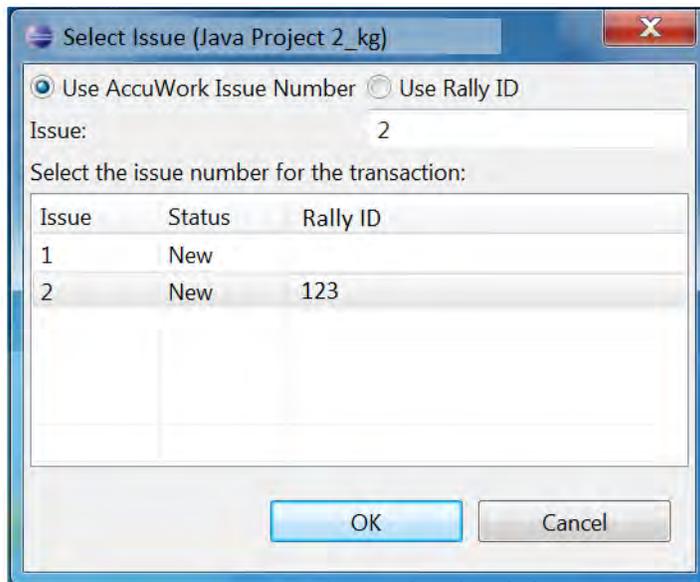
10. Select the issue against which you want to promote the project elements and click **OK**. The elements are promoted.

Adding an AccuRev Server

In the AccuRev Plug-In, you are asked to select the AccuRev server where your AccuRev files are stored. If your AccuRev server does not appear as a choice, you can add the server using this procedure.

1. Click the **New** button ( ) on the Eclipse toolbar.

   Alternative: Select **File** > **New** > **Project** from the Eclipse main menu or select **File** > **Import** from the Eclipse main menu.

   The New Project dialog box appears.
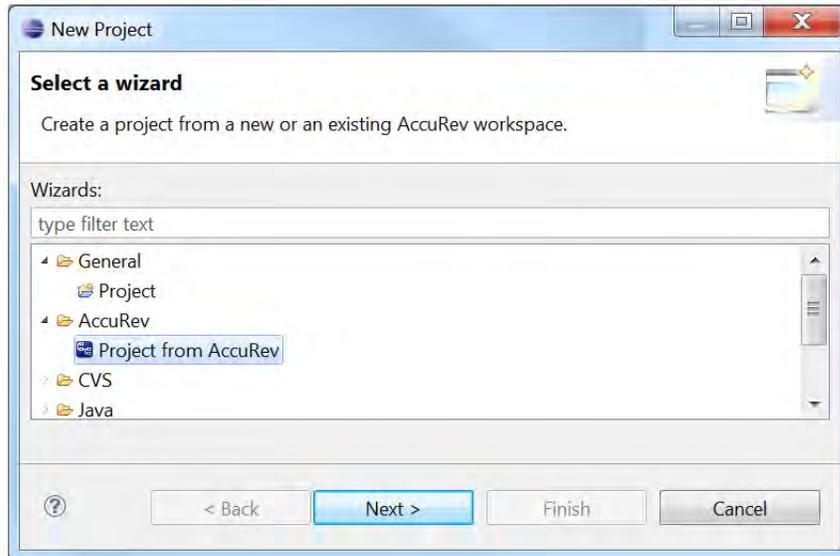
2. In the **Wizards** field, navigate to **AccuRev** > **Project from AccuRev**. (If you have many wizards, simply type **A** to go directly to AccuRev in the list.)

3. Click **Next**.

   Note: If you are not already logged in to AccuRev, you are prompted to do so.

   The Checkout from AccuRev page appears.
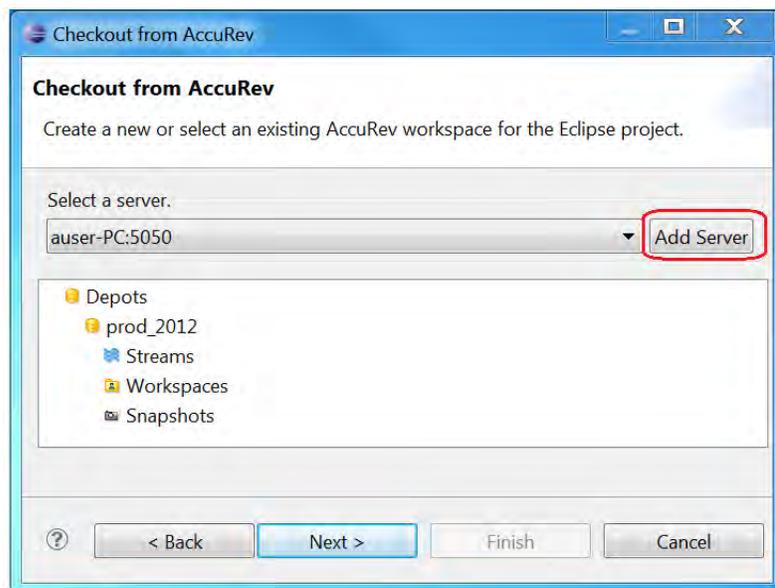
4. Click the **Add Server** button to add a server to the list of servers.

   The Available AccuRev Servers dialog box appears.
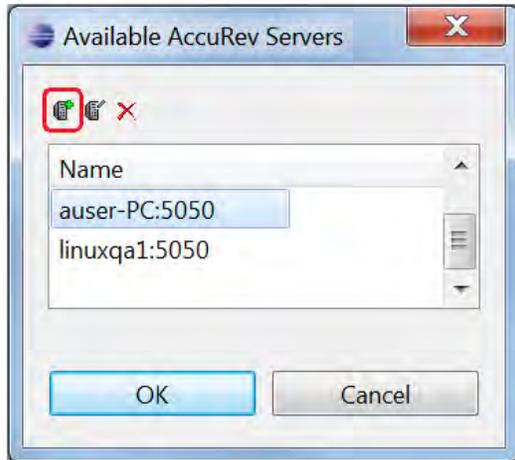
5. Click the Add Server button.

    The New Server dialog box appears.
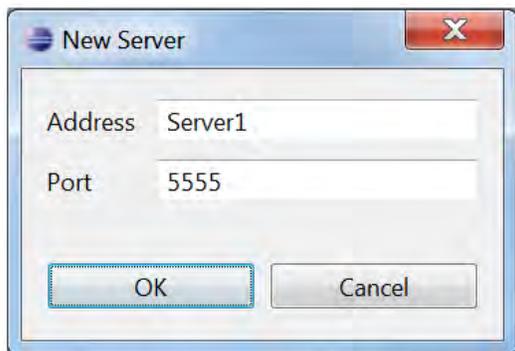
6. Complete the **Address** and **Port** fields and click **OK**.

    The new server is added to the list of servers.

Advanced Techniques for Using AccuRev Data

If you want to work with several distinct subtrees of an existing AccuRev workspace, you must create a separate Eclipse project for each subtree. When performing AccuRev commands that search through all the files in a project, you can choose to include all such "sibling" projects in a single search. See *Searching an Entire AccuRev Workspace* on page 59.

There is an alternative to creating several projects for several workspace subtrees: use AccuRev's include/exclude facility to configure the AccuRev workspace with just the files you want. Then, create an Eclipse project using the entire workspace.

Specifying UTF-8 Encoding

You can specify character encoding for the following objects in Eclipse:

- **Files.** You can specify the character encoding for individual files or for all files of a given content type.

- **Projects.** Specifying character encoding for a project sets the default for all new files created in that project. You can set character encoding for projects individually, or for all projects created in an Eclipse workspace.

- **Workspaces.** Specifying character encoding for a workspace sets the default for all new projects created in that workspace.

The remaining sections summarize the procedures you can use to specify UTF-8 character encoding in Eclipse. Refer to your Eclipse user documentation for additional information.

Setting Encoding for a File

To set the character encoding for an individual file:

1. In the Package Explorer, right-click the file whose encoding you want to set and the choose **Properties** from the shortcut menu.

2. On the Resource page, in the Text file encoding section, select Other and then choose UTF-8 from the drop-down list.

3. Click the **OK** button to save the settings and close the Properties dialog box.

Tip: You can also set the character encoding based on a file's content type. See *Other Preferences for Specifying Character Encoding* on page 33 for more information.

Setting Encoding for a Project

Setting the character encoding for a project sets the encoding for any new files created in that project.

To set the character encoding for a project:

1. In the Package Explorer, right-click the project whose encoding you want to set and then choose **Properties** from the shortcut menu.

2. On the Resource page, in the Text file encoding section, select Other and then choose UTF-8 from the drop-down list.
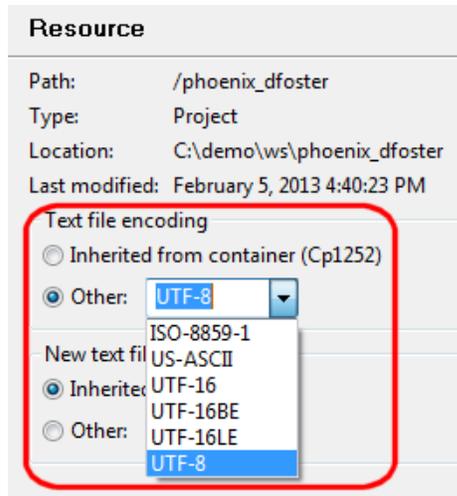
3. Click the **OK** button to save the settings and close the Properties dialog box.

Setting Encoding for a Workspace

Setting character encoding for a workspace sets the default for all new projects created in that workspace.

To set character encoding for a workspace:

1. From the main menu, select **Window** > **Preferences**.

   The Preferences dialog box appears.

2. Navigate **General** > **Workspace** to display the Workspace page.

3. In the Text file encoding group box, select Other and then choose UTF-8 from the drop-down list.
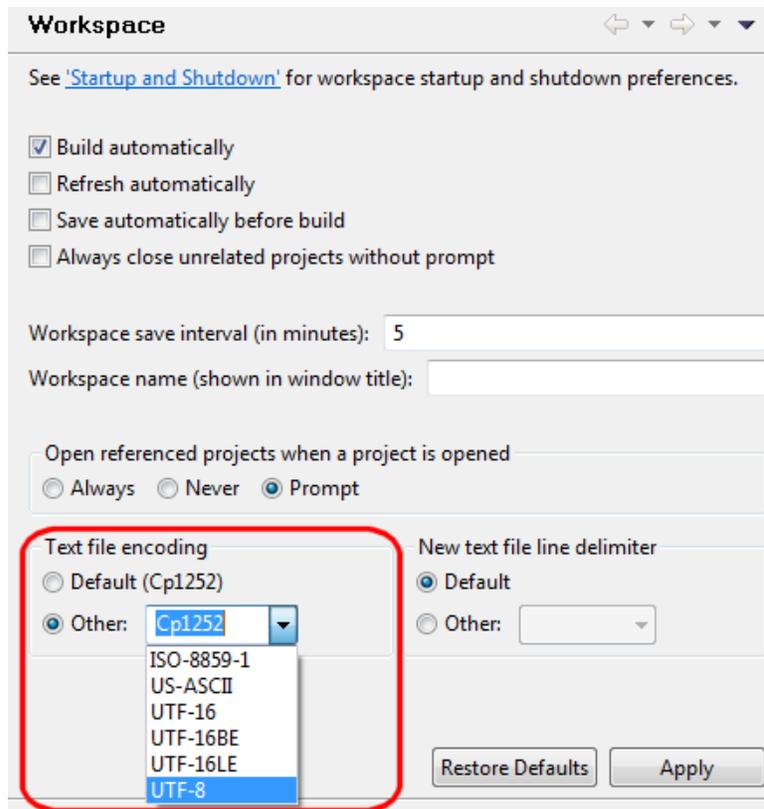
4. Click the **OK** button to save the settings and close the Properties dialog box.

Other Preferences for Specifying Character Encoding

In addition to using Eclipse preferences to set character encoding for workspaces, you can also use preferences to set character encoding based on content types, and for files associated with other Eclipse plug-ins you have installed (plug-ins for Web and XML application development, for example).

To locate all the places where you can specify preferences for character encoding, type "encoding" in the Preference dialog box filter field. As you type, titles of the preference pages containing "encoding" are displayed in the Preferences dialog box left pane:
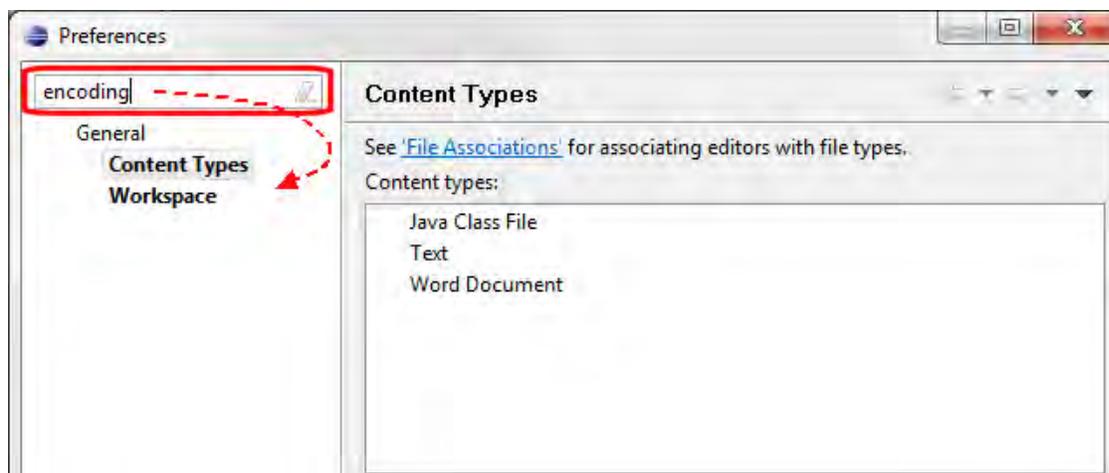
Navigate the preference pages for the plug-ins you have installed and review, and optionally set character encoding on those pages.

For example, to set character encoding based on a file's content type:

1. From the main menu, select **Window** > **Preferences**.

   The Preferences dialog box appears.

2. Navigate **General** > **Content Types** to display the Content Types page.
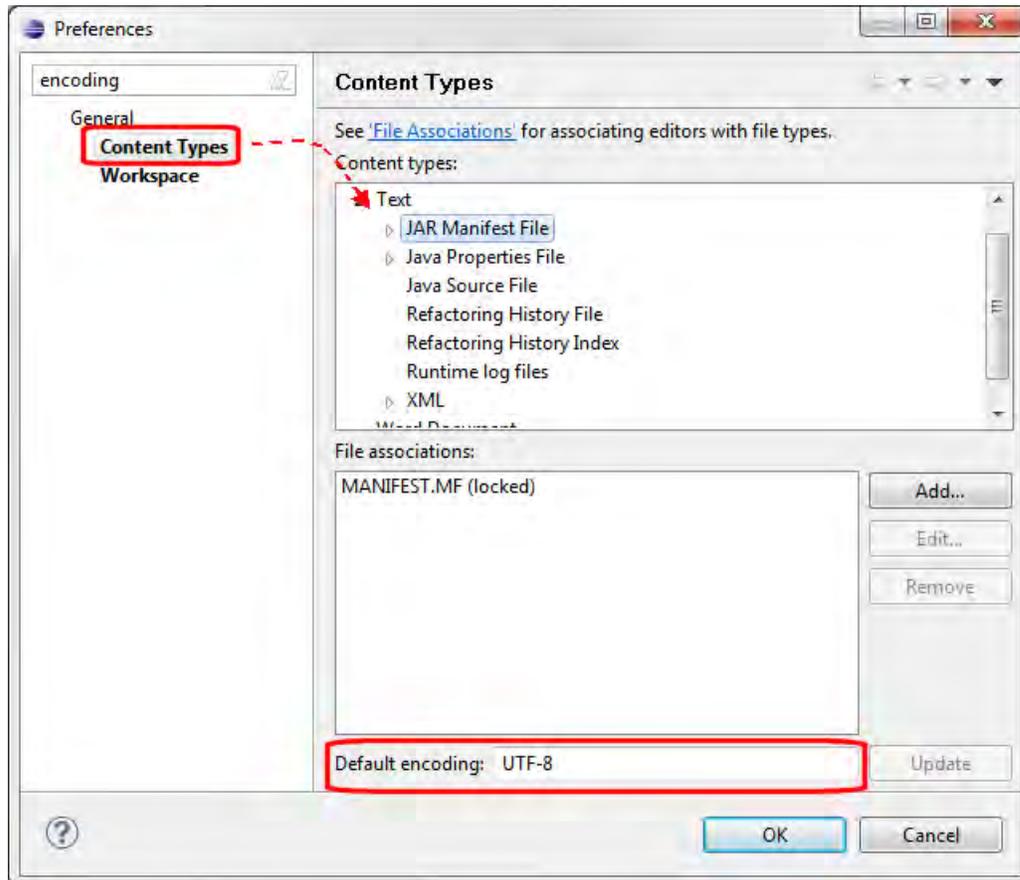
3. Scroll through the Content types field, review the corresponding file associations, and set the value in the Default encoding field as needed.

4. Click the **OK** button to save the settings and close the Preferences dialog box.

3. Working in the Eclipse Navigator View

The Navigator view shows all your Eclipse projects -- both projects associated with AccuRev and others. You access each project's files and directories (folders) using a familiar tree control.
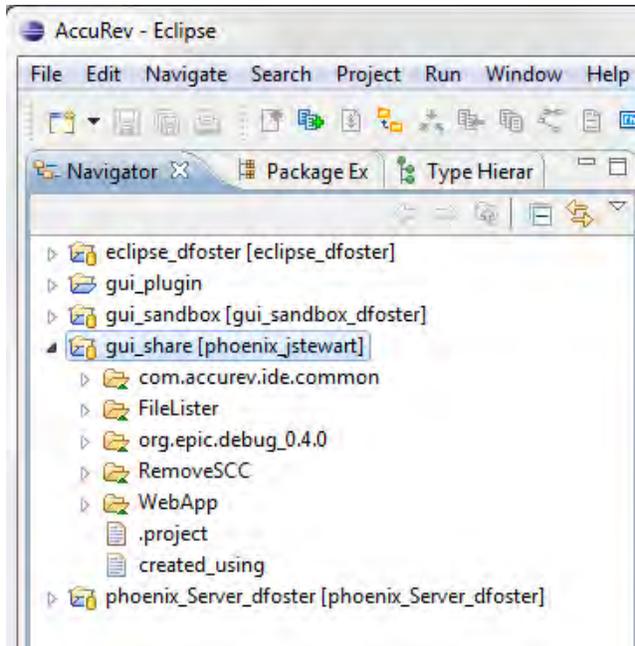
Note: For an Eclipse project consisting of AccuRev version-controlled files and directories, the Package Explorer view is equivalent to the Navigator view. In the rest of this document, descriptions relating to the Navigator view apply equally to the Package Explorer view.

Integrated Commands

The following AccuRev commands are integrated into the main context menu in Eclipse's Navigator view when you install AccuRev Plug-In:

- *Refactor*
- *Compare With*
- *Replace With*

To display the context menu, right-click a file or directory in the Navigator view.

Refactor

Eclipse supports two types of refactoring operations:

- Moving an object to a different directory
- Renaming an object

When you complete either of these operations, the AccuRev **Rename** command is invoked to record the change of pathname in the repository. The results are displayed in the *AccuRev Console View* on page 57.

> **Note:** The Eclipse **Delete** command removes an object from local disk storage, but does not perform any AccuRev command. (In particular, it does not perform a **Defunct** command). If you **Delete** a version-controlled element, its status becomes **(missing)**.

Note: Do not use the **Move** or **Rename** commands when you are not logged in to an AccuRev Server. Doing so causes changes to the Eclipse workspace but not to the AccuRev workspace. Visual cues indicate that you are not logged in to the AccuRev server, including a greyed-out repository decoration on the root of the Package Explorer tree ( ) and missing decorations on files and directories. See *Setting Label Decorations* on page 6 for more information on decorations.

Move

Eclipse supports moving an object to a different directory, using either of these methods:

- Right-click the object, and select **Refactor** > **Move** from the context menu.
- Drag-and-drop the object from one directory to another.

Note: Do not use the **Move** command to move an object from one Eclipse project to a project that is part of a different AccuRev workspace. The object will be moved, but the AccuRev **Rename** command will not be invoked. The object will have **(missing)** status in the source location and **(external)** status in the destination location.

Rename

Eclipse supports renaming an object (without changing its directory location), using either of these methods:

- Right-click the object, and select **Refactor** > **Rename** from the context menu.
- Select the object, and press function key **F2**.

Compare With

Eclipse supports commands to compare your version of a file with prior "local history" versions of that file. AccuRev Plug-In adds the following commands which perform comparisons with versions that were created with *Keep* and stored in the AccuRev repository:

- **Most Recent Version**
  Backed Version
  File On Disk
  Basis Version

  These commands are the same as those on the **Team** > **Diff Against** > menu. See *Diff Against* on page 44.

- **AccuRev Revision** -- Compares the file in the AccuRev workspace with the version created by a particular AccuRev transaction. A File History tab appears, containing a table of transactions involving the file. (This is the same as the upper table in the **AccuRev History** display; see *History View* on page 59.)

  Double-click a transaction to choose it for the comparison; this launches Eclipse's Compare Editor.
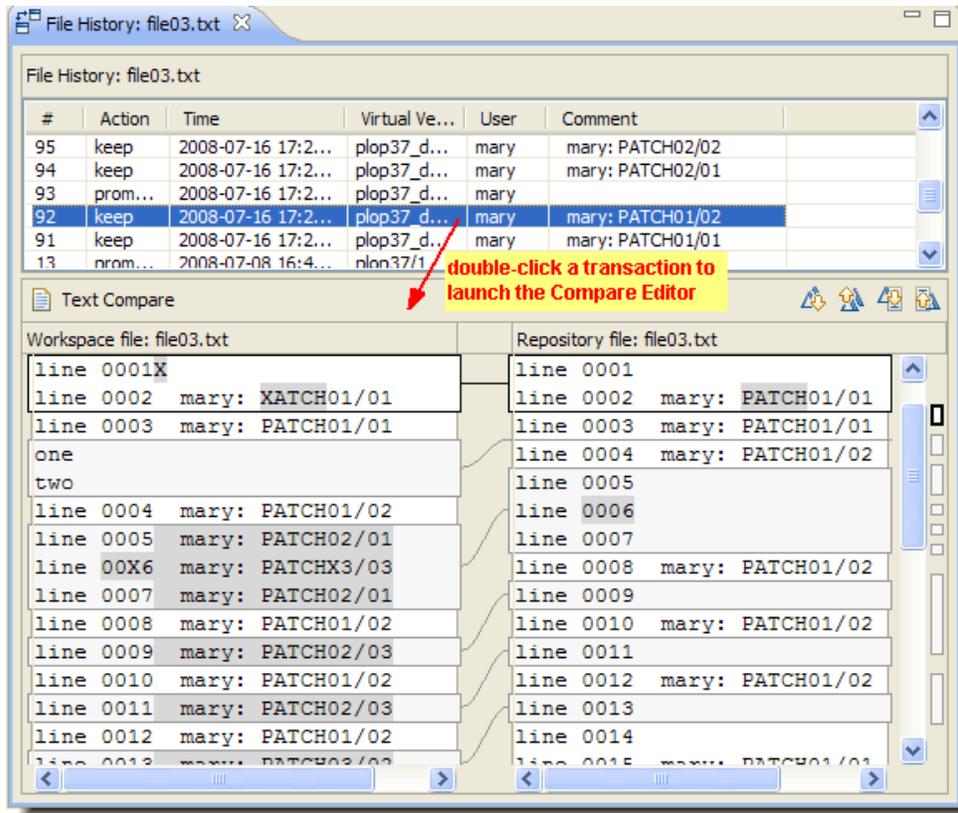
  See the Eclipse help system for more information on the Compare Editor--in particular, the topics *Compare Editor* and *Understanding the Comparison*.

Replace With

Eclipse supports commands to replace your version of a file with a prior "local history" version. AccuRev Plug-In adds the following commands which replace a file with a version that was created with *Keep* and stored in the AccuRev repository:

- **Most Recent Version**
  Backed Version

  These commands are the same as those on the **Team** > **Revert To** > menu. See *Revert To* on page 44.

- **AccuRev Revision** – Replaces the file in the AccuRev workspace with the version created by a particular AccuRev transaction. As with the **Compare With** > **AccuRev Revision** command, a File History tab appears, containing a list of transactions involving the file.

  Double-click a transaction to choose it for the comparison; this launches Eclipse's Compare Editor.

To proceed with the replacement operation, click the **Replace** button below the Compare Editor. To cancel the operation, just close the File History tab.

```
line 0009
line 0010
line 0011
```

Replace
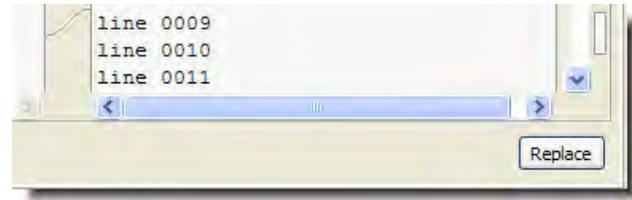
Team Menu Commands

The following table summarizes the AccuRev commands that are integrated into the Team context menu in Eclipse's Navigator or Package Explorer view:

Synchronize with AccuRev Repository	*Login*	*Logout*
Add to AccuRev Depot	*Keep*	*Anchor*
Promote	*Merge*	*Version Browser*
Annotate	*Populate*	*Defunct*
Revert To	*Diff Against*	*Copy as Link/Paste as Link*
Synchronize Time	*AccuRev Workspace Information*	*Update AccuRev Workspace*
AccuRev History	*AccuRev Statuses*	*AccuRev Searches*
Show Properties	*Refresh*	*Web UI*
Share/Unshare Project	*Create Patch/Apply Patch*	

To display the Team menu, right-click a file or directory in the Navigator view.

Each of these commands is described in the following sections.

Synchronize with AccuRev Repository

Performs an Eclipse "synchronize with the repository" operation on the current selection--which can include anything from multiple Eclipse projects to a single file. Synchronization results are displayed in the Synchronize view. (**Exception:** If you invoke this command on a single file, a simple file comparison is performed; the Synchronize view is not involved.)

For each object, the synchronize operation determines the differences between the version in your project--that is, in the AccuRev workspace--and the version in the backing stream. In the Synchronize view, you can invoke commands to resolve the differences: you can "promote" changes from your project to the repository and/or "update" changes from the repository to your project.

For details, see *Synchronize View* on page 63.

Login

You must be logged in to AccuRev in order perform any AccuRev operation on a repository file or AccuRev workspace.

If the login is invalid or the session expires, then no AccuRev operation can be performed on the open Eclipse project files.

See *Logging In to AccuRev* on page 1.

Logout

The **Logout** command logs the current user out of AccuRev. While logged out, AccuRev commands are not available, and the AccuRev server is not contacted. The icon on the project folder is greyed out, and label decorations are not shown for files in the workspace. Files in workspaces are still available, and their contents can be modified freely. Performing a *Synchronize with AccuRev Repository* operation after logging back in is recommended.

Add to AccuRev Depot

The **Add to AccuRev Depot** command converts one or more of the files in the development project into AccuRev version-controlled elements. The directory containing the files is also converted to an element, if necessary.

Keep

The **Keep** command saves the changes you have made to one or more files as "private" versions in the AccuRev repository. These versions are visible only in your workspace--not in the "public" backing stream or in other users' workspaces.

The **Keep** command is also available for directories. Performing this command on a directory recursively searches it for files with **(modified)** and **(external)** status to keep.

Do not confuse the versions of a file created by **Keep** with the "local history" copies of the file created when you invoke **File** > **Save** in an Editor pane. Local history copies are maintained by Eclipse itself in the local file system; versions created with **Keep** are saved permanently in the AccuRev repository.

Advanced Keep Options

You can use advanced options when keeping files. Click the **Advanced** button in the Keep dialog box to display these choices:

- **Choose File Type** – By default, AccuRev determines the element type for the newly created version automatically (text or binary). You can override the default by specifying either of these types or ptext (a variant of text).

- **Choose Lock Type** – Choosing **Enable File Locking** places a lock on the element to enforce serial development in sibling workspaces for that element. This ensures that users in sibling workspaces will not have to merge their work on this element. The lock remains on the element until you (or someone else) changes this value to **Disable File Locking** in a subsequent **Keep** command.

See your AccuRev user documentation for more information.

Anchor

Many version control systems force you to perform a "check out" command that makes a file writable, so that you can edit it. The AccuRev default is that your files are *always* writable.

The AccuRev **Anchor** is similar to a typical checkout. However, anchoring a file does not change whether it is writable. Rather, it "activates" the file in your workspace, so that it will be found by the **AccuRev Searches** > **Default Group** search.

One effect of anchoring a file is that it ensures that the file is not overwritten by an **Update AccuRev Workspace** command. Under normal circumstances, you rarely need to invoke the **Anchor** command.

If the AccuRev workspace uses the AccuRev "exclusive file locking" or "anchor-required" feature, files are initially read-only. Before editing a file, you *must* **Anchor** it, making it writable.

Promote

The **Promote** command converts "private" versions into "public" versions. That is, it takes versions that you previously created in your workspace with **Keep**, and sends them to the backing stream. If you have modified a file, this command first performs a **Keep**, then promotes the kept file.

Promote and Issue Management Systems

Invoking the **Promote** command can activate either or both of the integrations between AccuRev's configuration management functionality and its issue management (AccuWork™) functionality. The former uses change packages as the point of integration; AccuWork uses a particular issue-record field as the point of integration. Both of them record information about the **Promote** transaction in a user-specified AccuWork issue record. For more information, see *Promote-Based Integrations with Issue Management* on page 54.

Recursive Promote

You can select individual files or folders for the **Promote** command. If you have selected a folder for the **Promote** command, you can choose to perform a recursive promote: the current folder and all subfolders for pending and external elements.
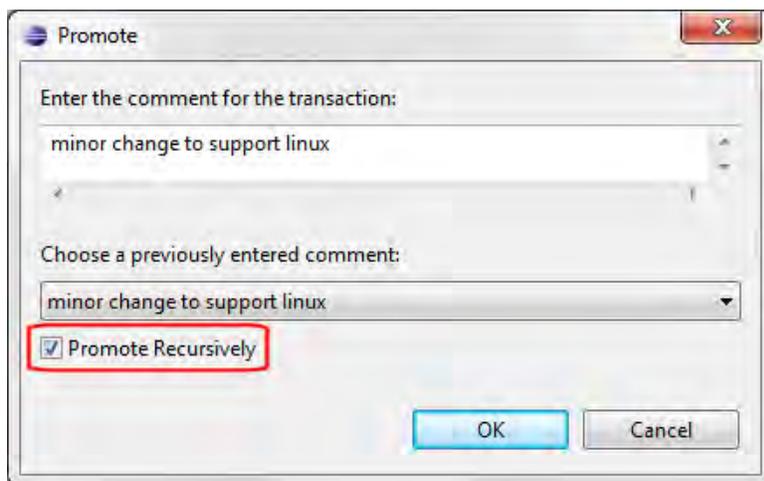
As shown in the following illustration, the Promote dialog box displays all pending elements and external elements:
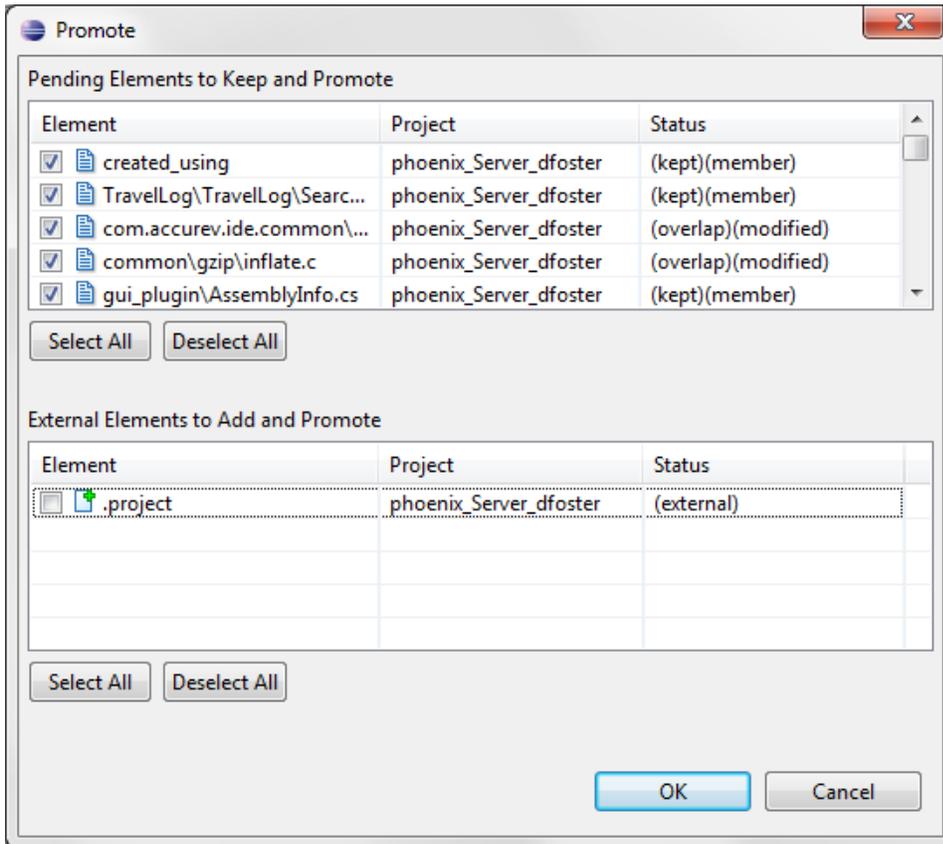
Pending elements are selected by default; you need to choose any external elements you want to promote.

Merge

The **Merge** command is enabled only when a file's status is **(overlap)**. It merges your version of the file with the version in the backing stream. For text files, AccuRev uses the merge tool you have configured--this can be AccuRev's own merge tool, the Eclipse tool, or a third-party tool that you specify. See *Using Alternative Diff and Merge Tools* on page 3 for more information.

Binary data is not merged. If a file is in binary format, the **Merge** command prompts you to choose whether to use your version (that is, the version in your AccuRev workspace / Eclipse project) or the version in the AccuRev backing stream.
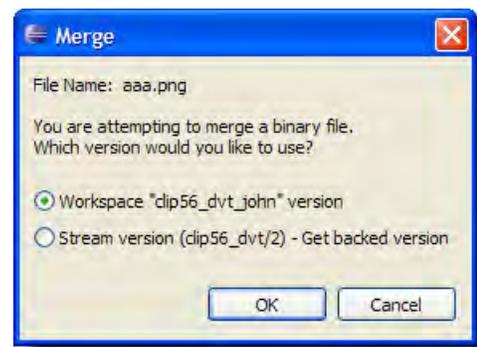
The element to be merged can be a directory. This occurs when the directory has been renamed or moved, both in your AccuRev workspace / Eclipse project and in the backing stream. In this situation, the **Merge** command prompts you to choose which name change to use. You can also choose to reject both changes, and use the directory's "original" name in the common-ancestor version.
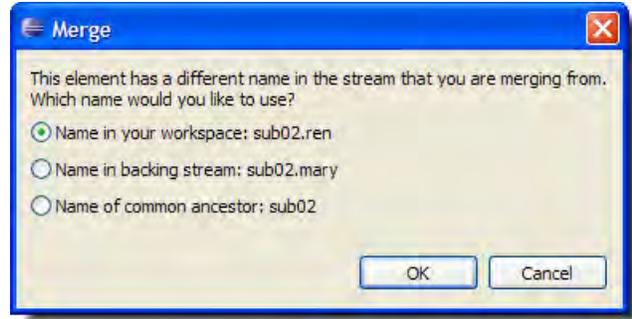
The **Merge** command can be invoked, perhaps multiple times, when you invoke the **Commit All Outgoing Changes** and the **Update All Incoming Changes** commands in the *Synchronize View*.

Version Browser

Shows the ancestry for each element you select in the Version Browser. It displays the current version of the selected element, the real ancestors of that version, and the virtual versions that have one of those ancestors as a parent. By default, data for 20 transactions is shown in the drop-down list of filters on the Version Browser toolbar.

The Version Browser timeline shows when each transaction took place, as well as the number and user associated with the transaction. Note that the timeline proceeds left-to-right, therefore the most recent versions will be visible on the right side of the Version Browser.

Note: The Version Browser is available only if AccuRev Plug-In has been configured to provide access to the AccuRev Web Interface (Web UI). See the *AccuRev Plug-In for Eclipse Installation and Release Notes* for more information on this process. See the *AccuRev Web Interface User's Guide* for more information on the Version Browser.

Annotate

The **Annotate** command shows the contents of the selected version of a text file, annotating each line with:

- The timestamp of the version in which that line was added to the file

- The number of the transaction in which that version was created

- The user who performed that transaction

- A line number

You can use the annotations to explore how the file has changed over time. When you click on an item in the *Date*, *#*, or *User* columns, all the lines whose value in that column matches the value of the item you clicked are highlighted. This makes it easy to:

- Find all the lines added or changed in a particular version (click in the *Date* or *#* column).

- Find all the lines created by a particular user (click in the *User* column).

A timeline is shown above the file contents. Each transaction involving the annotated file is represented on the timeline. Click a transaction to show the file as of that transaction. The lines of the file that have been added or changed by that transaction are highlighted.

Note: Annotate is available only if AccuRev Plug-In has been configured to provide access to the AccuRev Web Interface (Web UI). See the *AccuRev Plug-In for Eclipse Installation and Release Notes* for more information on this process. See the *AccuRev Web Interface User's Guide* for more information on Annotate.

Populate

Restores a file with **(missing)** status to your project. To locate such files, use the *AccuRev Search/Status View* on page 58.

The **Populate** command copies the version currently in the workspace stream (in the AccuRev repository) to the workspace. For more information, see "What's the Difference between Populate and Update?" in the *AccuRev Technical Notes*.

Invoking the **Populate** command displays a dialog with two options:

- *Recursive* – For each item in the list that is a *directory*, perform a **Populate** command on that element and on all elements below it.

- *Overwrite* – For each item in the list that is a *file*, replace the file (if any) currently in your workspace. If this checkbox is not selected, no files will be overwritten.

Click **OK** to proceed with populating files.

Defunct

Deletes a file from your disk, and also marks it as having **(defunct)** status in the AccuRev workspace. Defuncted files are not displayed in the Navigator view because it only displays files that are on your disk. However, defuncted files are displayed in certain AccuRev Searches: Defunct, Pending, and Default Group. Defuncted files disappear entirely from the AccuRev workspace when you **Promote** them to the backing stream.

You can also **Defunct** a directory. But before doing so, consult the description of the **defunct** command in the *AccuRev CLI User's Guide*.

Note: The Eclipse **Delete** command does not invoke the AccuRev **Defunct** command. It simply removes the file from local disk storage; no change is made to the AccuRev repository. If you invoke the Eclipse **Delete** command on an AccuRev-controlled file, its AccuRev status becomes **(missing).**

Revert To

The commands on the **Revert To** submenu allow you to discard changes to a file in your workspace, replacing them with another version. See *Replace With* on page 37 if you want more flexibility in selecting the replacement version.

- **Revert to > Most Recent Version**

  Replaces your file with the version in the workspace stream (in the AccuRev depot). Typically, the workspace stream contains the version you most recently created with *Keep*. Or it might contain the version you brought into your workspace with *Update AccuRev Workspace*.

  Note: This command does not discard namespace changes.

- **Revert to > Backed Version**

  Replaces your file with the version that was in the backing stream at the time of your most recent *Update AccuRev Workspace*, or the last version you *Promote*d, whichever is more recent.

Diff Against

Most **Diff Against** commands allow you to compare a text file in your workspace with another version of the same file. AccuRev uses the diff tool you have configured--this can be AccuRev's own diff tool, the Eclipse tool, or a third-party tool that you specify. See *Using Alternative Diff and Merge Tools* on page 3 for more information.

- **Diff Against > Backed Version**

  Compares your file with the version currently in the workspace's backing stream.

- **Diff Against > Most Recent Version**

    Compares your file with the version in the workspace stream (in the AccuRev depot). Typically, the workspace stream contains the version you most recently created with *Keep*, so this command helps answer the question, "What have I changed since my last Keep?". The workspace stream might contain the version you brought into your workspace with *Update AccuRev Workspace*, or the version you recently *Promote*d; in these cases, the command is equivalent to **Diff Against > Backed Version**.

- **Diff Against > Basis Version**

    Compares your file with the version you most recently brought into your workspace with *Update AccuRev Workspace*. However, if you have *Promote*d the file since the update, the comparison is with your most recently promoted version.

- **Diff Against > File on Disk**

    Allows you to compare a text file in your workspace with some other file (either on the hard disk, or available via a networked machine).

Ignore Options for AccuRev Diff

If you are using the AccuRev Diff tool, you can specify whether or not you want AccuRev to ignore one or more of the following:

- Whitespace (whitespace is ignored by default)

- Changes in whitespace

- Case

Ignore options for AccuRev Diff are specified on the AccuRev page of the Preferences dialog box. See *Setting AccuRev Preferences in Eclipse* on page 2 for more information.

Copy as Link/Paste as Link

Allows you to create an element link or a symbolic link from a file. To create a link:

1. Right-click the file for which you want to create a link and choose **Team** > **Copy as Link** from the context menu.

2. Navigate to the project in which you want to create the link.

3. Right-click the appropriate project folder and choose **Team** > **Paste as Link** from the context menu.

    The Paste Link dialog box appears:
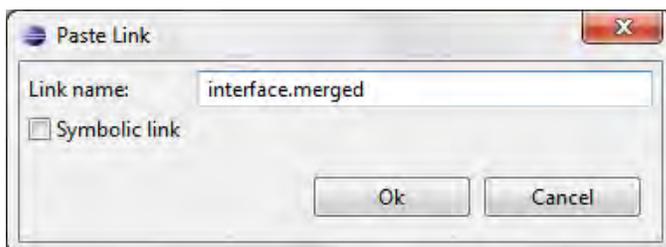
    By default, the link object you are creating has the same name as the target object to which it refers.

4. Optionally, change the link name.

5. Select the **Symbolic link** field if you want to create a symbolic link. Otherwise, AccuRev Plug-In creates an element link.

6. Click OK.

Synchronize Time

The **Synchronize Time** command changes a client machine's system clock to match the clock on the AccuRev server.

AccuRev Workspace Information

The **AccuRev Workspace Information** command displays a message box containing information about the current AccuRev context: your user name, your AccuRev workspace name and location, and so on. It also indicates whether the integrations between AccuRev's configuration management and issue management capabilities are enabled. See *Promote-Based Integrations with Issue Management* on page 54.

Update AccuRev Workspace

The **Update AccuRev Workspace** > **Entire Workspace** command copies versions from your workspace's backing stream into your workspace. This has the effect of incorporating other people's changes, which they have promoted to the backing stream, into your workspace.

The **Update AccuRev Workspace** > **Preview** command shows which files have changed and would be copied to your workspace if you issue the **Update AccuRev Workspace** > **Entire Workspace** command.

AccuRev History

Displays a table in the History view listing the AccuRev transactions (**keep**, **promote**, and so on) involving the selected element. The previous contents of this view, if any, are discarded. See *History View* on page 59.

This command is disabled if multiple elements are selected.

AccuRev Statuses

Displays a table listing information about each element in the current selection:

*   Element's pathname within the depot

*   AccuRev status (one or more indicators)

*   Version ID of this version

*   Element ID

The result table is displayed in the AccuRev Search/Status view.

The previous contents of this view, if any, are discarded. See *AccuRev Search/Status View* on page 58.

AccuRev Searches

Searches the current folder in the AccuRev workspace for elements that satisfy the selected search type.

Note: The **Stranded** search searches the entire workspace.

To search the entire workspace, use the **Entire AccuRev Workspace** option available from the tab. See *AccuRev Search/Status View* on page 58.
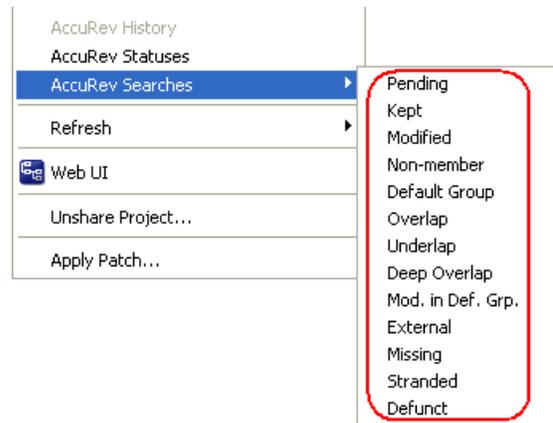
Show Properties

Displays the Properties dialog box for the currently selected element.
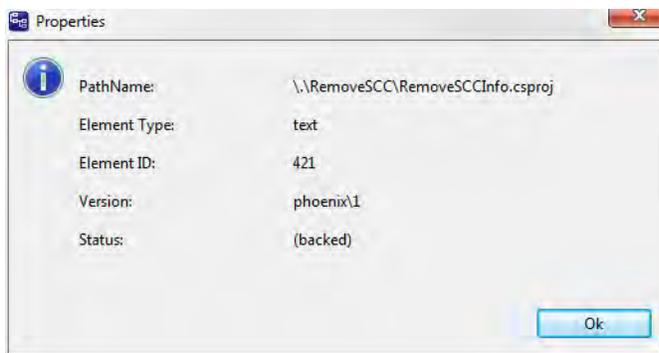
Properties listed for the element include the:

- Path and element name

- Element type

- Element ID

- Version

- Status

Refresh

Executes the AccuRev refresh operation, which updates the AccuRev status symbol for elements displayed in the Navigator view. You can perform the **Team** > **Refresh** operation for all projects or currently selected projects.

Tip: Pressing F5 (or **File** > **Refresh**) also performs the AccuRev refresh operation if you have enabled this preference. See *General AccuRev Settings* on page 3 for more information. The Refresh choice on the Navigator context menu, however, performs only the Eclipse refresh operation.

Web UI

Displays the AccuRev Web UI, providing a convenient way to view information from the AccuRev repository. See *AccuRev Web UI* on page 49 for details.

Note: The Web UI command is available only if AccuRev Plug-In has been configured to provide access to the AccuRev Web Interface (Web UI). See the *AccuRev Plug-In for Eclipse Installation and Release Notes* for more information on this process.

Share/Unshare Project

(Eclipse 3.4 and later only) Invoke **Share Project** to create an internal Eclipse association (mapping) of an Eclipse project to AccuRev.

Invoke **Unshare Project** to remove the selected projects from AccuRev's control. The selected projects will no longer be associated with AccuRev workspaces. All AccuRev decorations are removed from the files and folders in that project, and the **Team** menu no longer allows access to AccuRev commands.

Use the **Team** > **Share Project** command to restore the project to AccuRev control.

See *Sharing an Eclipse Project with AccuRev* on page 24 for more information about sharing projects.

Create Patch/Apply Patch

Create Patch creates a portable Eclipse patch file based on one or more files in your AccuRev workspace. The **Create Patch** command filters files in your workspace, displaying only those that have been modified. A file under AccuRev control is considered *modified* if it has one of the following statuses:

- **(modified)**
- **(modified)(member)**
- **(overlap)(modified)**
- **(overlap)(modified)(member)**
- **(kept)(member)**

Creating a Patch

To create a patch from modified files under AccuRev control:

1. In the Eclipse Navigator view (from the Project Explorer, for example), right-click the file you want to use to create a patch and choose **Team** > **Create Patch** from the context menu.

   Tip: You can select multiple files.

   The Create Patch dialog box appears.
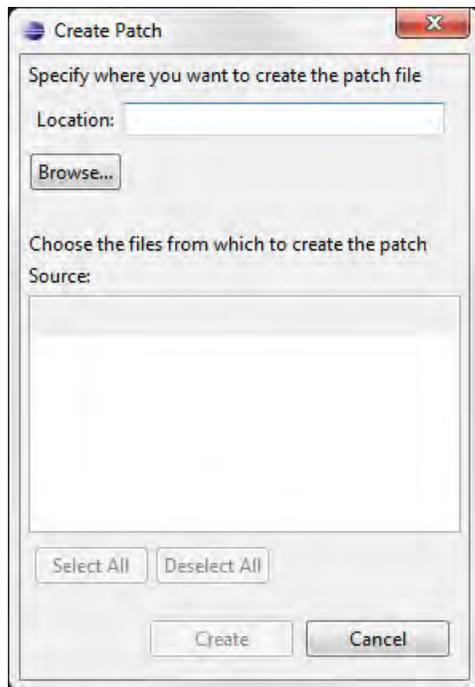
2. Specify the full path of the location where you want to create the patch file. Optionally, specify a file name. If you do not specify a file name, the patch is created as ac_patch.txt.

   Note: One patch file is created, regardless of the number of files you choose in *Step 1*.

3. Expand the Eclipse project folder.

4. Select the files from which you want to create a patch.

5. Click **Create**.

   The patch file is created using the name and location you specified in *Step 2*.

Applying a Patch

You use the **Apply Patch** command to apply a patch created using the **Create Patch** command.

Refer to your Eclipse documentation for more information about applying patches.

AccuRev Web UI

Note: The AccuRev Web UI is available only if AccuRev Plug-In has been configured to provide access to the AccuRev Web Interface (Web UI). See the *AccuRev Plug-In for Eclipse Installation and Release Notes* for more information on this process.

The Web UI lets you access both AccuRev and AccuWork™ data. Note that the Web UI only interacts with data from the AccuRev Server. Data from user workspaces that has not been kept or promoted is not accessible from the Web UI.

Starting the Web UI

To start the Web UI:

1. Click the Open Web UI button on the Eclipse toolbar:

   Alternative: Choose **Team** > **Web UI** from the Navigator view context menu.

   The Login to AccuRev Server dialog box appears.

2. Enter a user name and password.

   Tip: Select Remember my login on this computer to create a session token. Once created, you will not have to log in to the AccuRev Server when launching the Web UI in subsequent AccuRev Plug-In sessions.

3. Click **Ok**.

   The Web UI appears on a new tab.

Getting Started with the Web UI

To get started in the Web UI, choose an action from the toolbar to view or open streams or issues. The status bar at the bottom of the window displays information about the AccuRev Server and depot (if any) you are currently accessing.

For more information, see the Web UI online help.

Web UI Toolbar

The toolbar at the top of the tab contains general AccuRev commands:

- **StreamBrowser** – Displays stream options that allow you search and filter streams based on various criteria, and display them in either tabular or graphical displays.

- **New Query** – Displays the New Query tab.

- **Issue Queries** – Displays the Query Browser.

- **New Issue** – Creates a new AccuWork issue.

- **Issue Search** – Opens an existing AccuWork issue (for searches by issue number) or list of issues (for full-text searches).

- **Refresh** – Updates the information shown in the active tab to reflect recent AccuRev activity.

- **User:** – Displays the user's name. Clicking the down-arrow displays the Change Password dialog box.

- **Login** – Displays the Login dialog box, allowing you to log in as a different user.

- **Logout** – Logs out the current user.

- **Help** ( ) – Displays the online help in a Web browser.

Toolbar Access to AccuRev Commands

To provide easy access to commonly-used AccuRev features, several of the AccuRev commands described above are also available from icons that appear on the Eclipse toolbar:

The icons from left to right are:

- Add to Depot
- Keep
- Anchor
- Promote
- Merge
- Revert to Backed
- Show History
- Version Browser*
- Annotate*

- Rename

- Delete

- Defunct

- Show Properties

- Synchronize

- Update

- Open Web UI*

- Open Stream Browser*

- Create New Issue*

- Open Issue Queries*

An asterisk (*) identifies a feature that is available only if you have configured access to the AccuRev Web Interface (Web UI).

Running Commands in the Background

Long-running commands -- **AccuRev Searches**, **AccuRev History**, **Refresh**, **Populate, Diff Against**, and **Merge** -- can be run in the background. When the command begins execution, a pop-up window appears:
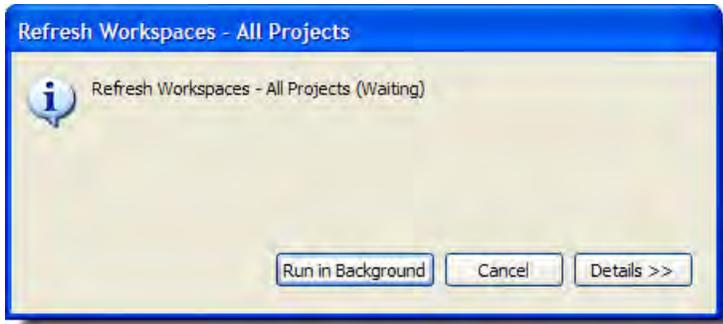
- Clicking **Run in Background** closes the pop-up window and places a "show progress" button in the lower right corner of the Eclipse window.
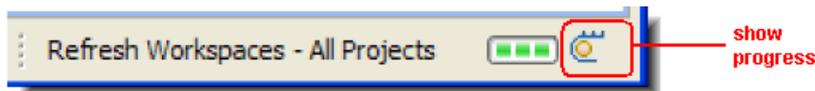
  Clicking the "show progress" button opens a standard Eclipse Progress view, which provides information on all background commands. You can cancel a background command in the Progress view, by clicking its red button.
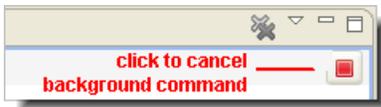
- If you click **Details** in the pop-up window, the command continues to run in the foreground. A progress view, as described above, is added to the pop-up window.

Promote-Based Integrations with Issue Management

AccuRev defines two integrations between its configuration management functionality and its issue management (AccuWork) functionality:

- Change-package-level
- Transaction-level

The integrations are enabled separately, through AccuRev commands that are not available in AccuRev Plug-In. One or both of the integrations are triggered when you invoke a **Promote** command on a set of elements. You are prompted to specify an AccuWork issue record in a pop-up window.

If only the transaction-level integration is enabled *and* you do not have a default query defined for the issues database, the prompt window contains a text field into which you type an issue number (or several numbers, separated by SPACE characters).

Otherwise, the prompt window offers a set of existing issue records, from which you can select one or more. You can also enter an issue in the *Issue #* field. When a "3pty ITS Key" is set in the AccuRev Schema editor, that key is displayed instead of the issue number.
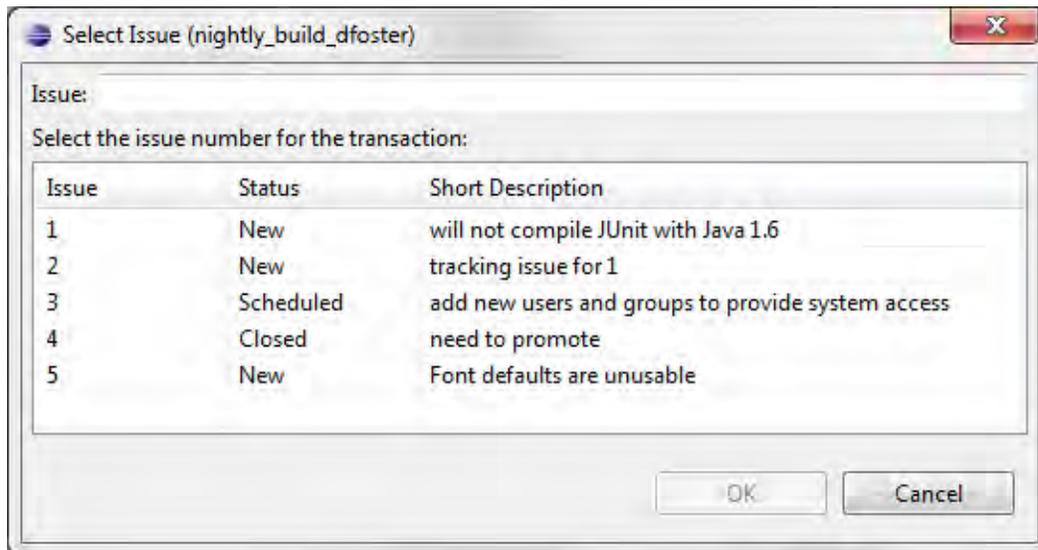
Once you select one or more issues to associate with your changes and click **OK**, AccuRev Plug-In displays a dialog box that allows you to enter an optional comment that documents the nature of the change or the reason for promoting your changes:
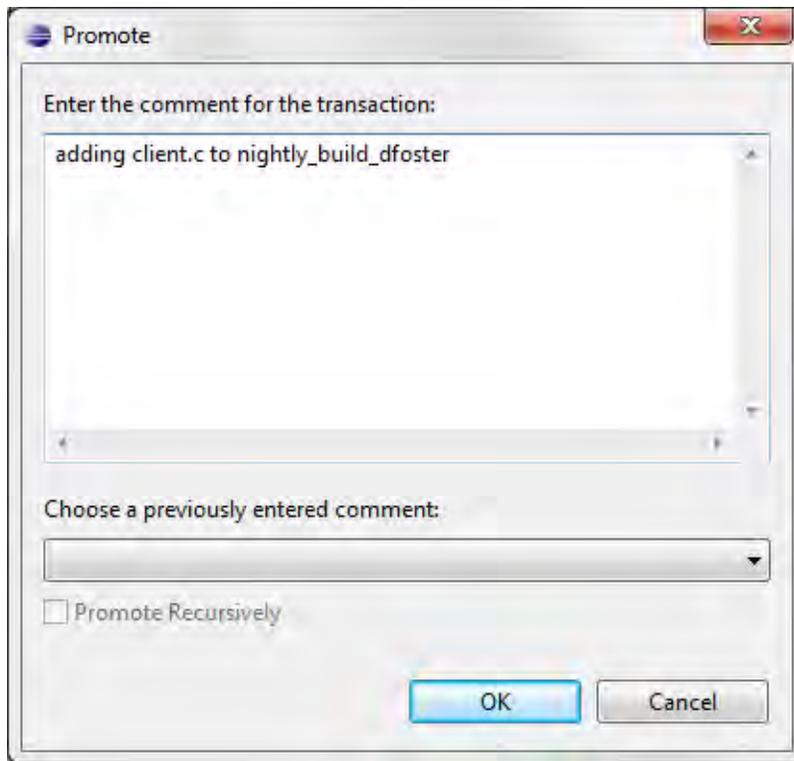
When you click OK to complete the **Promote** operation:

- The change-package-level integration (if enabled) records the promoted versions on the Changes tab of the specified issue record(s).

- The transaction-level integration (if enabled) records the **Promote** transaction number in the **affectedFiles** field of the specified issue record(s).

Note: If you use AccuRev to establish one of these integrations or to set a default query for the issues database while an Eclipse session is active, you might need to restart Eclipse to see the effect of this change.

For more information, see the section titled *Integrations Between AccuRev and AccuWork* in the *AccuRev Administrator's Guide*.

4. Working in the AccuRev Perspective

An Eclipse *perspective* defines a set of *views* (subwindows) that can be displayed in the Workbench window. AccuRev® Plug-In for Eclipse defines an **AccuRev** perspective, which includes these views:

- *AccuRev Console View*

- *AccuRev Search/Status View*

- *History View*

- *Synchronize View*

To open the AccuRev perspective, select **Window** > **Open Perspective** > **Other** from the Eclipse main menu. Select **AccuRev** in the Open Perspective dialog box and click **OK**.

AccuRev Console View

Many AccuRev commands generate informational messages. When you invoke these commands using AccuRev® Plug-In for Eclipse, the messages are sent to the AccuRev Console view. This view automatically appears (if it is not already visible) whenever an AccuRev command produces a user message. You can also display the console view using the Show View dialog box (**Window** > **Show View** > **Other**). Choose **AccuRev Console** under the **AccuRev** node on the dialog.

Each time you invoke an AccuRev command through the **Team** context submenu, a notice is appended to the contents of the AccuRev Console view. For example:

```
---- AccuRev command started: Keep
---- AccuRev command finished: Keep
```

If the command produces output, it is sent to the console:

```
---- AccuRev command started: Promote
Validating elements.
Promoting elements.
```
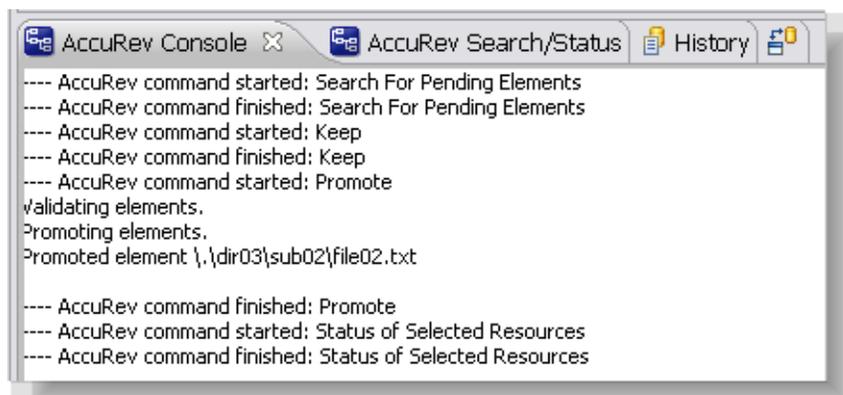
```
Promoted element \.\dir03\sub02\file02.txt
---- AccuRev command finished: Promote
```

Some commands (for example, **Team** >) both write to the AccuRev Console view and write to another view in the AccuRev perspective (in this example, the History view).

The data displayed in the AccuRev Console view is discarded when you close the view. It is not discarded if you switch to another view.

AccuRev Search/Status View

When you perform an AccuRev search (for example, **Team** > **AccuRev Searches** > **Modified**), the results are shown in a table in the AccuRev Search/Status view. By default, the results of all searches except the Stranded search display only the elements from the selected files and/or directories. (The Stranded search displays **(stranded)** elements from the entire workspace.)
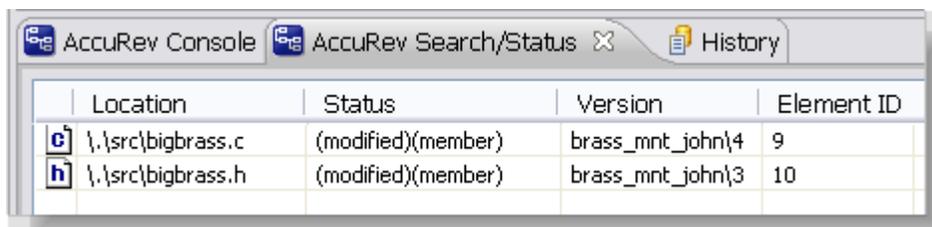
Each object is listed by its pathname relative to the AccuRev workspace's top-level directory. This is called a depot-relative pathname in AccuRev, and it always begins with /./ (Unix/Linux) or \.\ (Windows).

Working with the Results Table

The table in the AccuRev Search/Status view works in the usual way:

• Resize a column by dragging its right-hand column separator.

• Resize a column to accommodate the longest value by double-clicking its right-hand column separator.

• Sort column data by clicking on a column header to sort the rows on that column's values. A second click reverses the sort order.

The Search/Status view is *pinned* by default--that is, results for every search are displayed in a new instance of the Search/Status view. The data displayed in the AccuRev Search/Status view is discarded when you close or unpin the view.

Invoking Commands on Objects Displayed in the Results Table

You can invoke AccuRev commands on the objects displayed in the AccuRev Search/Status view. The context menu, including the **Team** submenu, provides access to the same AccuRev commands as in the Navigator view. This makes it easy to perform such operations as:

• Promoting some or all of the elements that are pending promotion.

• Placing under version control some or all of the workspace's external files.

You may prefer to use Eclipse's native "synchronize with the repository" interface to perform such operations. See *Synchronize View* on page 63.

Searching an Entire AccuRev Workspace

An Eclipse project can include just part of an AccuRev workspace, called a *subtree*. You might create multiple such subtree projects that, as a group, span the entire AccuRev workspace. On the other hand, you might choose to omit some portions of the AccuRev workspace from the group of Eclipse projects.

To return elements that meet the search criterion throughout the entire AccuRev workspace, select **Entire AccuRev Workspace** from the drop-down menu near the minimize and maximize controls.

Use the drop-down menu to toggle the **Entire AccuRev Workspace** setting. This setting persists as long the AccuRev Search/Status View remains open.

Example: Suppose you have loaded just a workspace's **src** directory -- not the **doc** or **tools** directory -- into an Eclipse project. Invoking the command **Team** > **AccuRev Searches** > **Pending** displays the elements in the Eclipse project (that is, in the **src** directory) that are pending promotion. Then, turning on the **Entire AccuRev Workspace** setting expands the display to include all pending-promotion elements in the AccuRev workspace, including those that are not loaded into the Eclipse project.

History View

The History view displays a table containing some or all of the AccuRev transactions involving a selected element.

When you select a transaction in the upper table, the lower table is populated with data on all the elements involved in that transaction. The data for each element includes its pathname within the depot, the AccuRev version identifiers of the version this transaction created, identifiers of the versions from which this version was derived, and any associated AccuWork issue record numbers.
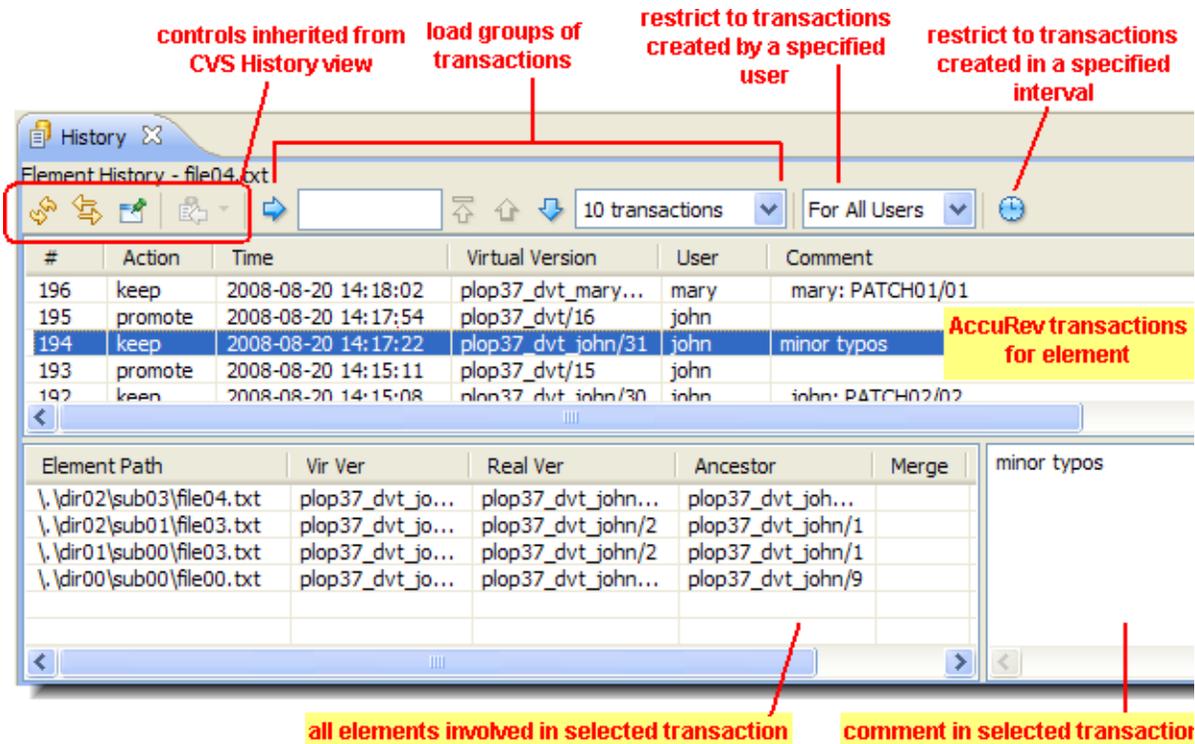
You can manipulate the table in the usual way -- see *Working with the Results Table* on page 58. This view is a variant of the CVS History View, and so inherits some of its controls:

- **Refresh** -- Checks for new transactions and displays them.

- **Link with Editor and Selection** -- A toggle switch; when linking is enabled, selecting an element in the Package Explorer / Navigator or switching to an element's Editor tab automatically displays the element's history. Up to 15 of these histories are cached in the History view; you can redisplay a cached element history with the **Show element history** toolbar button.

- **Pin this history view** -- Ensures that the currently displayed element history does not get overlaid. Invoking the **AccuRev History** command again creates a new instance of the History view.

  Note: When the **Link** toggle is enabled, the **Pin** toggle is automatically disabled, and vice-versa.

- **Show element history** -- Displays the next element history in the cache enabled by the **Link with Editor and Selection** toggle, or opens a drop-down menu with all the cache entries.

The element history displayed in an AccuRev History view is discarded when you close the view. So is any element history cache you have accumulated. This data is also discarded if you invoke another command when the **Pin** toggle is off.

Navigating in the Transaction Table

If an element has many transactions, you can use the AccuRev toolbar controls at the top of the tab to assist in navigating among them.

Initially, the most recent set of transactions is loaded into the table. Use the drop-down box to control how many transactions are included in the set.

To load a set beginning with a particular transaction into the table, type the transaction number into the input box, and click the right-arrow button.
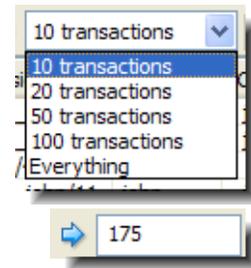
To scroll to the next or previous set of transactions, use the ⬇ or ⬆ button. The ⬆̄ button loads the first (most recent) set of transactions.

By default, the transaction table includes all transactions that involved the selected element. You can restrict the display to:

- The transactions performed by a specific user (by choosing from the drop-down list of users)

- The transactions created in a specific interval:
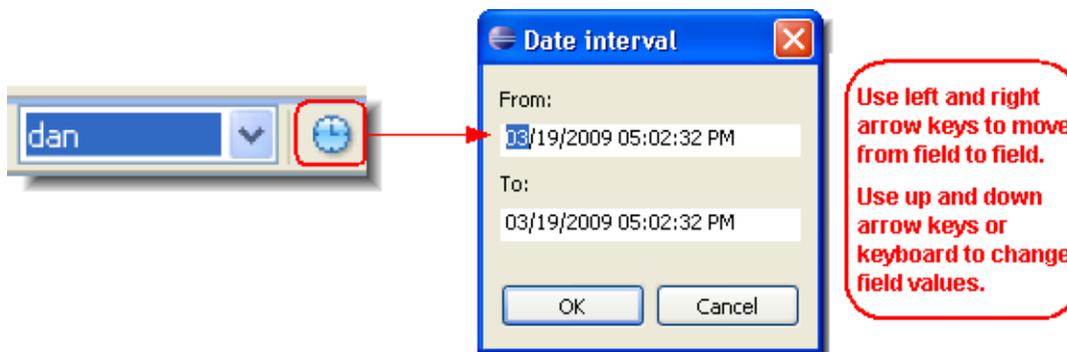
History View Commands

You can select a *transaction* in the upper table and right-click to invoke commands that operate on the version created by the selected transaction:

- **Open** -- Opens an Editor tab on the file.

- **Export** -- Copies the file to a location you specify.

- **Merge From** -- Opens a Merge tool to merge the selected version into your workspace.

- **Patch From** -- Opens the Eclipse Compare editor, which allows you to copy (patch) changes from one version of a file to another.

  Note: In order to perform a Patch From command, the merge tool must be set to Eclipse in the AccuRev preferences. See *Using Alternative Diff and Merge Tools* on page 3 for more information on setting preferences.

- **Diff Against** >

  - **Other Version** *(enabled when two transactions are selected)* -- Compares the versions created by the two selected transactions.

- **File in the Workspace** *(enabled when a single transaction is selected)* -- Compares the version associated with the transaction with the version in the workspace.

You can configure the file-comparison tool; see *Using Alternative Diff and Merge Tools* on page 3.

You can select an *element* in the lower table and right-click to invoke commands that operate on the version that this transaction created:

- **Open** -- Opens an Editor tab on the file.

- **Export** -- Copies the file to a location you specify.

- **Diff Against** > **Previous Transaction** -- Compare the selected version with the version that was in its workspace or stream just before the version was created.

- **Properties** -- Displays the depot-relative pathname of the selected version, as well as the element type and ID.

Patching From the History View

Use the following procedure to patch a file using changes from another version of that file. The target for the patch operation is the version in your Eclipse project (your AccuRev workspace).

Before you begin, make sure AccuRev® Plug-In for Eclipse is using the Eclipse Merge tool. See *Using Alternative Diff and Merge Tools* on page 3 for more information on setting this preference.

To patch from the History view:

1. In the Navigator view, right-click the file you want to patch and choose **Team** > **AccuRev History**.

   The History view appears. The Element History section displays all of the transactions that involve the file you selected.

2. Review the transactions to find the one that contains the changes you want to patch into your version of the file.

   File versions associated with that transaction are displayed in the lower part of the Element History section.

3. Right-click the transaction and choose **Patch From** from the context menu.

   The Eclipse Compare editor appears. The Present File pane displays the version of the file in your Eclipse project; the Workspace file pane displays the version whose changes you want to patch into that file.
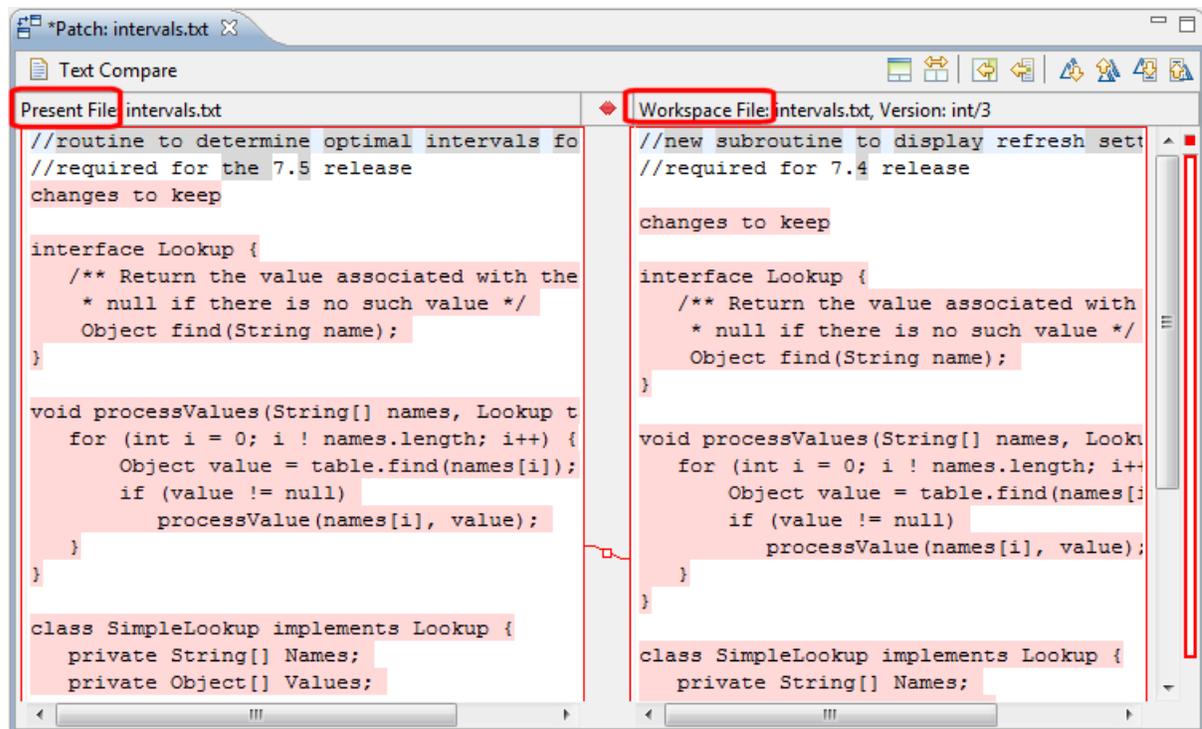
4. Scroll through the Workspace File to review changes. Use the:

- **Copy All Non-Conflicting Changes from Right-to-Left** button ( ) to copy *all* changes to your version of the file.

- **Copy Current Change from Right to Left** button ( ) to copy *individual* changes one at a time.

  Note: Changed text must be selected for this button to be active.

5. Save your changes (**File** > **Save**).

   Tip: Closing the Compare editor prompts you to save changes if you have not already.

   AccuRev® Plug-In for Eclipse prompts you to **Keep** the patched file.

6. Click **Yes** to **Keep** the patched file; otherwise, click **No**.

Synchronize View

The AccuRev perspective includes the standard Eclipse Team Synchronize view. Most of the controls and commands available in this view work the same way with AccuRev as they do with the CVS version-control system. This means you can use the Eclipse online help facility to learn about synchronization in general and about the specific facilities of this view. This section describes the AccuRev-specific aspects of using the Synchronize view.

An Eclipse *synchronization* is a file-by-file listing that answers the question, "What are the current differences between my projects and the repository?".

With AccuRev, a synchronization actually takes into account three locations:

- The AccuRev "project" or workspace tree. The AccuRev workspace is typically located on your machine's hard drive.

- The AccuRev "repository" or depot. The AccuRev depot contains two locations corresponding to a given project:

  - Your AccuRev workspace stream, which tracks your private versions -- for example, those created by **Keep**.

  - Your AccuRev workspace's backing stream, which tracks versions that you have made public with **Promote**.
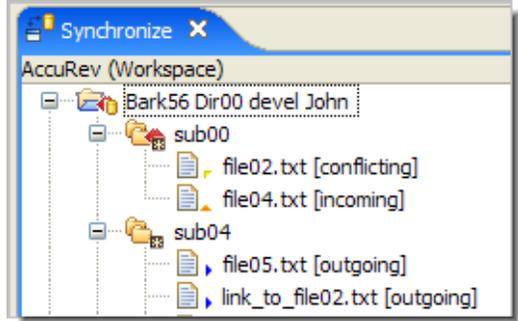
Since AccuRev also version-controls directories and links, a synchronization can list these objects, as well.

Eclipse Annotations

Eclipse annotates each object in a synchronization, to indicate the type of difference:

[incoming]

> The object changed in the repository, but not in the project.

[outgoing]

> The object changed in the project, but not in the repository.

[conflicting]

> The object changed in both the project and the repository.

The following table summarizes how Eclipse annotations correspond to AccuRev statuses:

Eclipse Annotation	AccuRev Status
[incoming]	(stale)
[outgoing]	(modified) (kept) (member) (external)
[conflicting]	(overlap)

You can view an object's AccuRev status by invoking the **Show Properties** command from its context menu or from the AccuRev toolbar.

Notes:

- The AccuRev command **Anchor** changes a file's AccuRev status to **(member)**, but does not affect the file's contents. Such a file is classified as **[outgoing]** in a synchronization, even though there might be no difference between the file's contents in the project and in the repository.

- AccuRev icon label decorations are included for each object in the synchronization. These decorations can help you identify the various kinds of outgoing objects -- for example, to distinguish the **(overlap)** files from the **(modified)** files. See *Setting Label Decorations* on page 6 for more information.

- Icons for files with a status of **(external)** are displayed without any decoration.

Performing Operations in the Synchronize View

You can work in the Synchronize view by invoking commands from its toolbar or from the context menu of an object (or a selected set of objects). Many of these commands work with AccuRev in exactly the same way as they do with CVS -- for example, the context menu commands **Open in Compare Editor** and **Remove from View**, and the toolbar commands **Pin Current Synchronization**, **Mode**, and **Presentation**.

The following images provide a visual quick-reference for the standard Synchronize view commands. For detailed help, use the Eclipse **Help** menu.

The same AccuRev commands that are available in the Navigator view (**Keep**, **Promote**, **Defunct**, etc.) also appear on object context menus in the Synchronize view. These commands are described in *Team Menu Commands* on page 38.
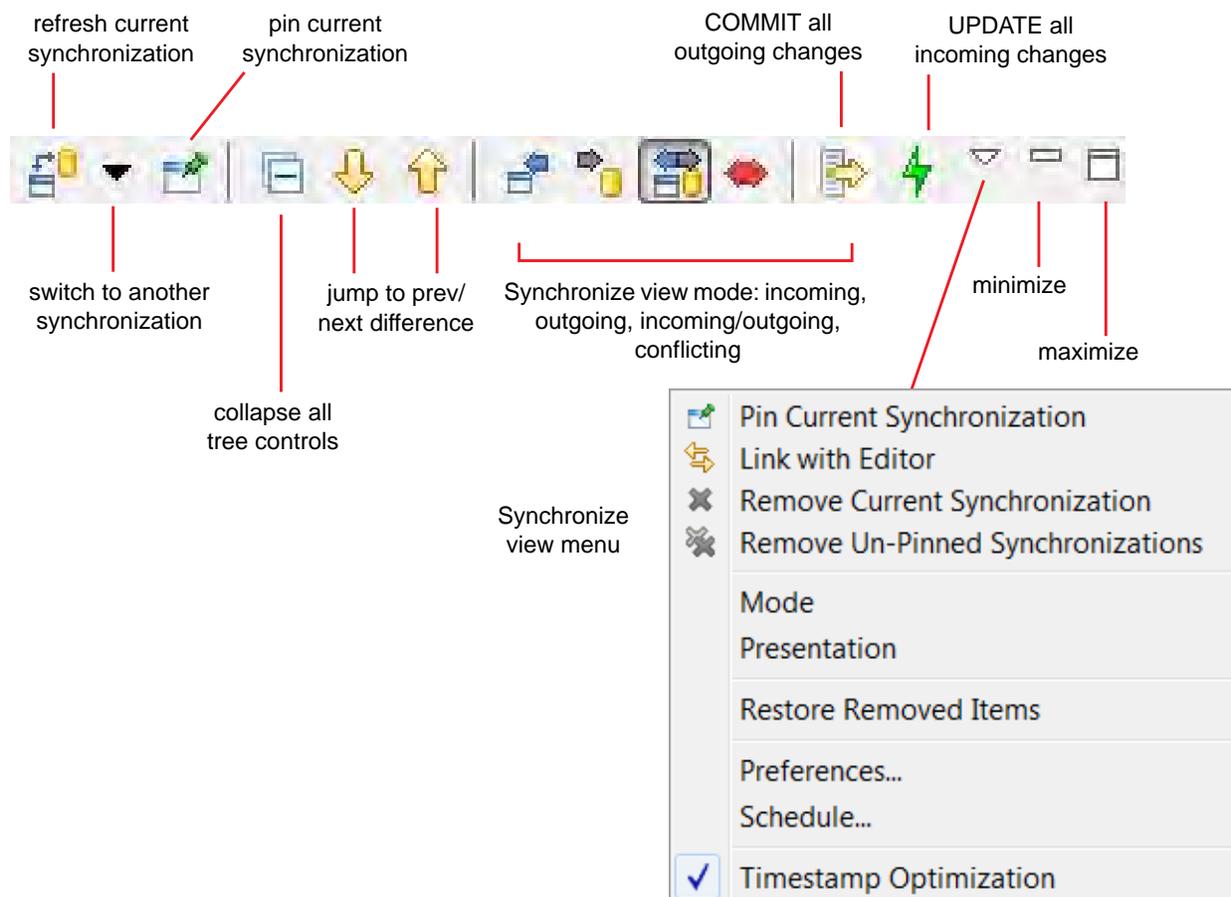
A common operation in the Synchronize view is comparing the project's version of a file with the repository's version. You can do this using Eclipse or AccuRev tools:

- To invoke the Eclipse Compare Editor, double-click the file or select **Open in Compare Editor** from its context menu.

- To invoke the AccuRev Diff Tool, select **Diff Against** > **Backed Version** from its context menu. (You can also access **Diff Against** from within the Commit All dialog box as described below.) You can configure this command to invoke the Eclipse Compare Editor, third party, and custom Diff tools. See *Using Alternative Diff and Merge Tools* on page 3.

The following sections describe the most important Synchronize view operations: increasing the level of synchronization between your project and the repository, either by "committing" (transferring data from the project to the repository) or by "updating" (transferring data from the repository to the project).

Committing Outgoing Changes

Eclipse uses the CVS term "commit" to mean the transferring of changes from your project to the repository. You initiate a commit operation by clicking the **Commit All Outgoing Changes** button in the Synchronize view toolbar. This launches a multiple-page wizard, which leads you through the invocation of some or all of these AccuRev operations:

- **Add to Depot** – creates new versions of objects with **(external)** status.

- **Keep** – creates new versions of elements with **(modified)** status.

- **Merge** – creates new versions of elements with **(overlap)** status.

- **Promote** – sends the project's **(kept)** or **(modified)** versions -- including versions created during this wizard session--to the backing stream.

These operations are organized into a series of pages. On each page, you can select none, some, or all of the eligible [outgoing] elements to be processed. The AccuRev operations take place when you click **Finish** on the wizard's final page. The following example shows the wizard screen to keep and promote **(modified)** elements.
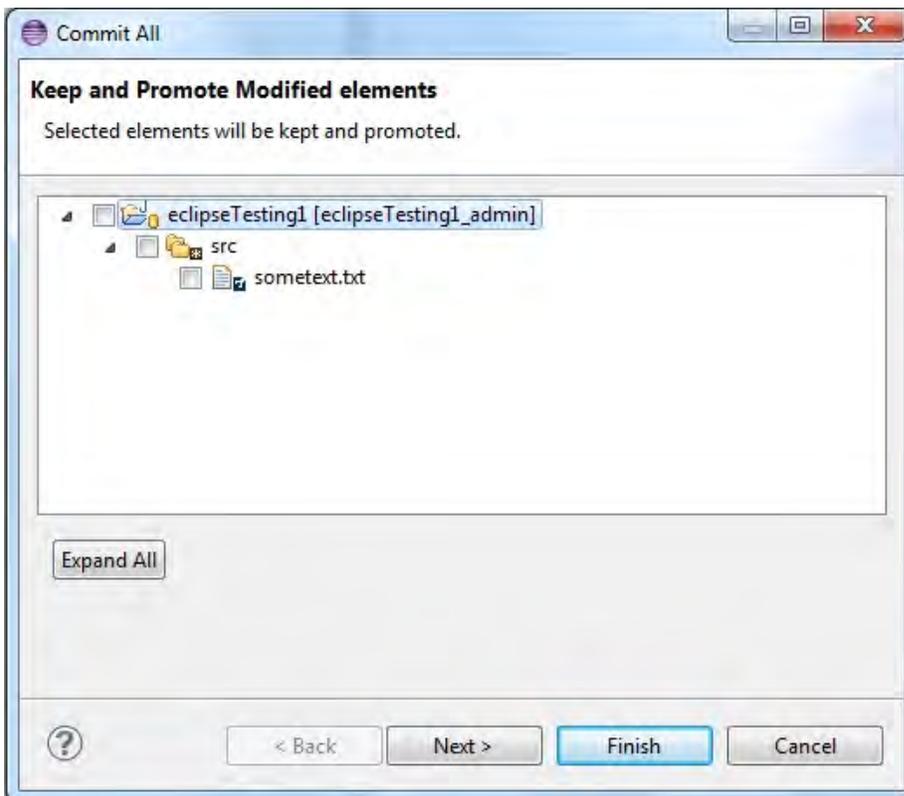
You can right-click individual elements in the Commit All Outgoing Changes dialog box to access a **Diff Against** context menu.

To add and promote external files, click **Next**.
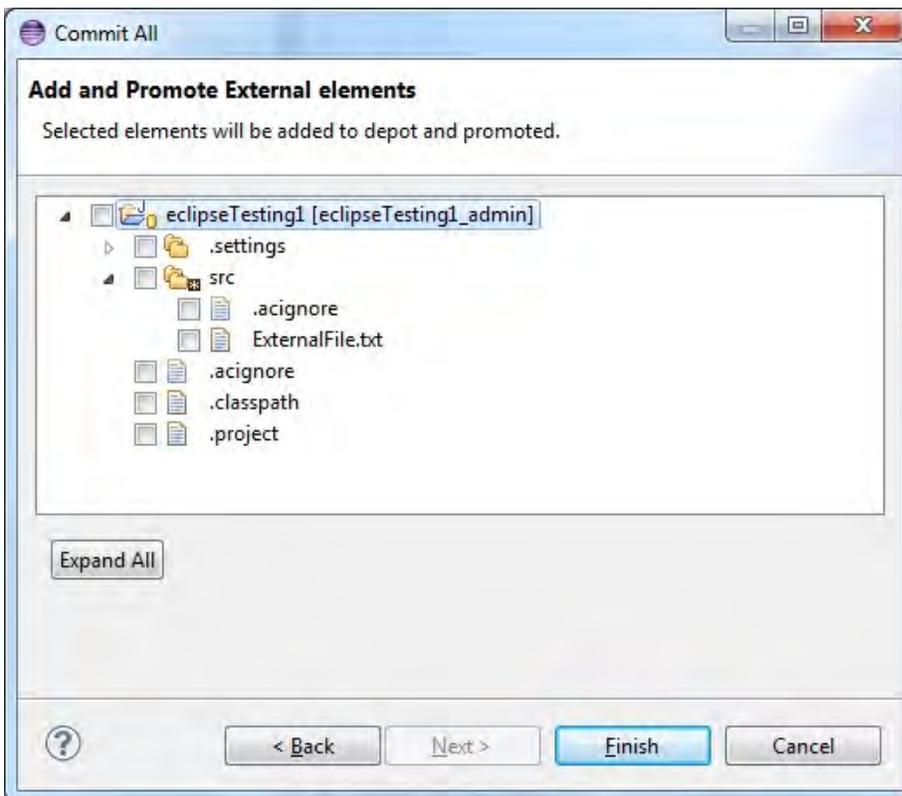
One invocation of **Commit All Outgoing Changes** can result in many AccuRev commands being performed. If your synchronization spans multiple projects in multiple AccuRev workspaces, separate AccuRev transactions are performed for each workspace. The wizard prompts you just once for a comment string, and attaches it to each AccuRev transaction that accepts comments.

Updating Incoming Changes

Eclipse uses the CVS term "update" to mean the transferring of changes from the repository to your project. AccuRev also uses the term "update". Sometimes, clicking the **Update All Incoming Changes** button in the Synchronize view toolbar simply performs an AccuRev **Update** command. But it sometimes launches a multiple-page wizard, which leads you through some operations to be performed before the AccuRev **Update** command proceeds.

The wizard's first page handles the following "AccuRev Update restriction":

> An AccuRev **Update** cannot proceed if any element has **(modified)** status but does not also have **(member)** status.

That is, the offending elements have not been made "active" in the workspace with the **Anchor** or **Keep** command. You can locate such elements in the Navigator view with the command **Team > AccuRev Searches > Non-member**.

(This page does not appear if the synchronization contains no such "non-member" elements.)
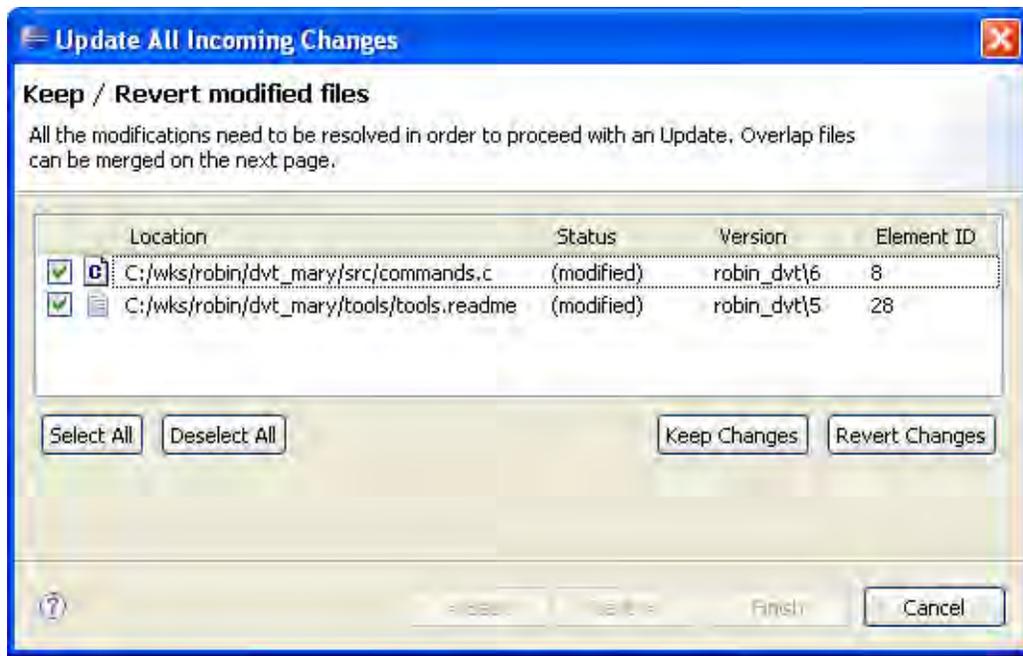
On this page, you *must* process all elements whose status is **(modified)**. You can process **(overlap)(modified)** elements here, too -- or you can wait to process some or all of them on the next page.

You can resolve the **(modified)** status of elements in either of these ways:

- **Keep Changes** performs a **Keep** command on all the selected elements. These elements will not be overwritten in the coming **Update** operation.

- **Revert Changes** performs a **Revert to Backed** command on all the selected elements, causing them to have **(backed)** status. In the coming **Update** operation, these elements will be overwritten by the current version in the backing stream.

When the list has no more elements whose status is **(modified)** only, you can:

- Click **Next** to proceed to the second wizard page, on which you specify **Merge** operations to be performed before the **Update**.

- Click **Finish** to proceed directly to the **Update**, without doing any merge work.

- Click **Cancel** to exit the wizard without performing the **Update** at all. (This does not cancel your **Keep Changes** and **Revert Changes** work, which has already taken place.)

The wizard's second page handles files with **(overlap)** status. If you chose not to handle one more files with **(overlap)(modified)** status on the preceding page, they appear here.
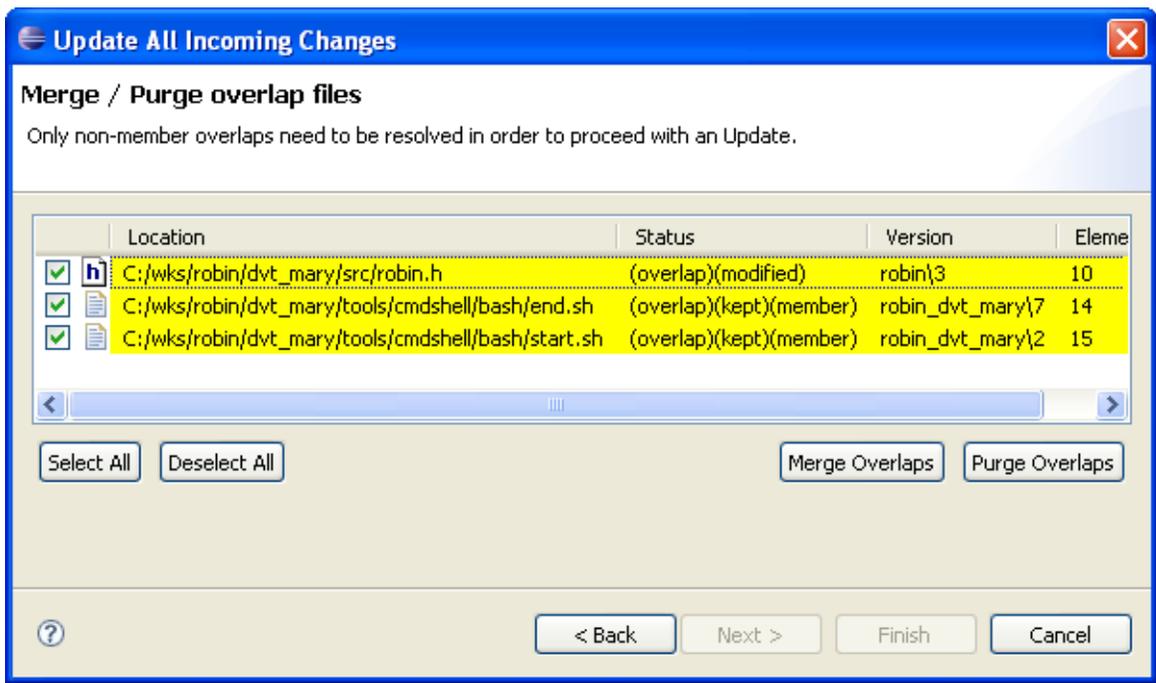
You must resolve files with **(overlap)(modified)** status, because of the AccuRev Update restriction discussed at the *beginning of this section*. Resolving files with **(overlap)(kept)(member)** status is optional, because the Update operation will not try to overwrite them.

You can resolve the **(overlap)** status of elements in either of these ways:

- **Merge Overlaps** performs a **Merge** command on all the selected elements, combining the changes in the repository (backing stream) with your project's changes. This **Keep**s a new version of each element. These elements will not be overwritten in the coming **Update** operation -- but that is OK, since you have already incorporated the changes from the repository (backing stream).

- **Purge Overlaps** performs a **Revert to Backed** command on all the selected elements, causing them to have **(backed)** status. In the coming **Update** operation, these elements will be overwritten by the current version in the backing stream.

When the list has no more elements whose status is **(overlap)(modified)**, you can:

- Click **Finish** to proceed to the **Update**.

- Click **Cancel** to exit the wizard without performing the **Update** at all. (This does not cancel your **Merge Overlaps** and **Purge Overlaps** work, which has already taken place.)

AccuRev Plug-In for Eclipse Integration with Mylyn

AccuRev Plug-In for Eclipse supports an integration between AccuWork and Mylyn. Mylyn is an Eclipse plug-in that manages user tasks. You can use the AccuRev Plug-In for Eclipse integration with Mylyn to perform AccuWork procedures within the Eclipse interface, such as creating and managing AccuWork queries and issues.

Note: The Mylyn plug-in is not included when you install AccuRev Plug-In for Eclipse. If you want to use the integration between AccuWork and Mylyn, you must separately install the Mylyn plug-in.

When you install AccuRev Plug-In for Eclipse version 2013.5 or later, the integration with Mylyn is automatically active if the Mylyn plug-in has been installed. The procedures described in the following sections assume that you have installed the Mylyn plug-in.

Adding an AccuWork Task Repository

Before using the AccuRev Plug-In for Eclipse integration with Mylyn, you must add an AccuWork task repository.

To add an AccuWork task repository:

1. In Eclipse, open the Mylyn view by selecting **Window** > **Show View** > **Other** > **Mylyn** > **Task Repositories**.

    The Task Repositories tab is active.

2. Click the Add Task Repository 🔵 button.

    The **Select a task repository type** page of the Add Task Repository wizard appears:
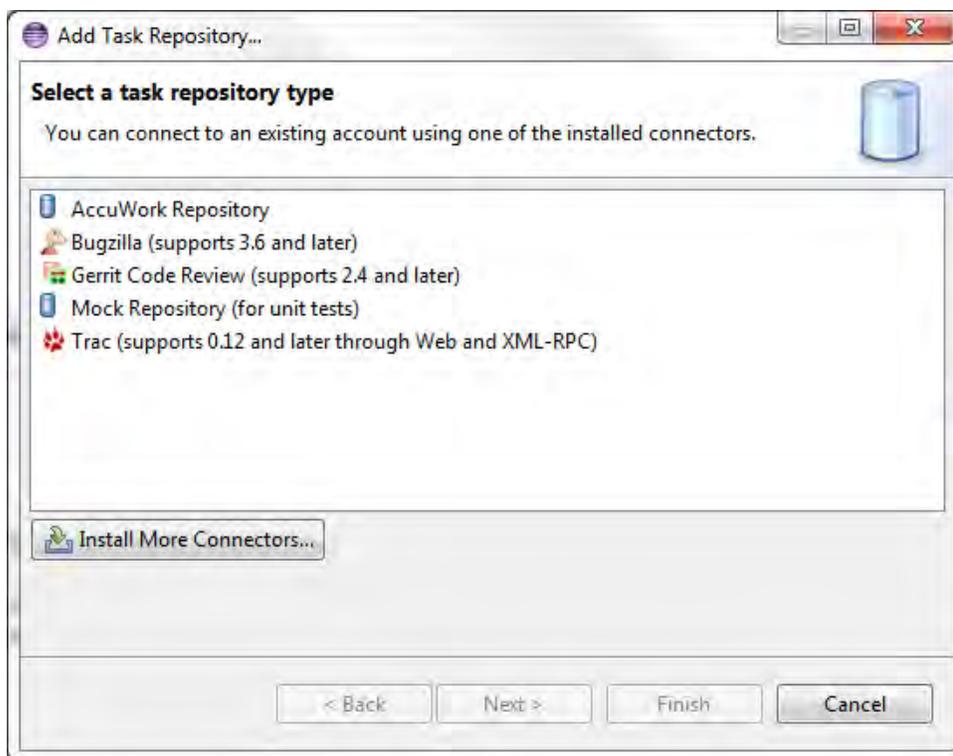
3. Select the **AccuWork Repository** option and click **Next**.

    The **AccuWork Repository Settings** page of the Add Task Repository wizard appears:
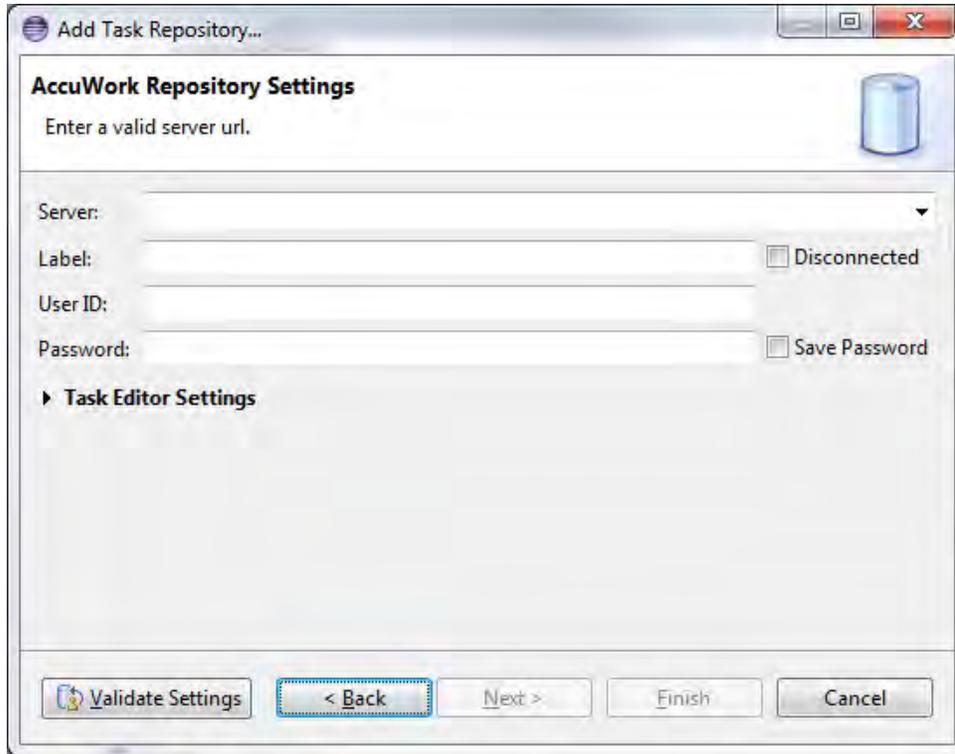
Specify settings as described in the following table:

Option Name	Description
Server	Your AccuRev server. The server information you enter should take the following form: *<server_name>:<port_number>* Note that the server must be one of the AccuRev servers that is already available to you through the AccuRev Plug-In for Eclipse.
Label	A name you designate for your AccuRev server.
Disconnected	The default state of the AccuRev server is disconnected.
User ID	Your AccuRev user name.
Password	Your AccuRev password.
Save Password	Indicates that you want Eclipse to remember your password between sessions.
Task Editor Settings	Optional settings that specify the look and feel of the Task Editor.

4. Click **Validate Settings** to confirm the accuracy of the AccuRev server settings.

5. Click **Finish** to apply the current settings and dismiss the dialog box.

Logging In to an AccuWork Task Repository

You must be connected to an AccuWork task repository before you can create and run AccuWork queries.

To log in to an AccuWork Task repository:

1. In Eclipse, open the Mylyn view by selecting **Window** > **Show View** > **Other** > **Mylyn** > **Task Repositories**.

   The Task Repositories tab is active.

2. Right-click the name of your AccuRev Task repository and select **Properties**.

   The Properties for Task Repository window appears:
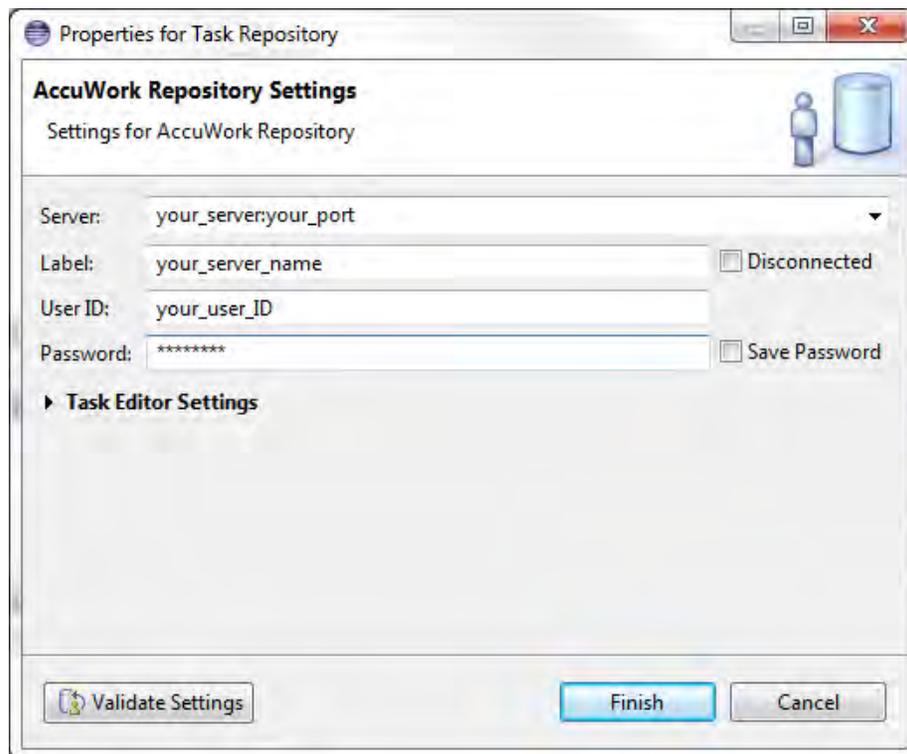
3. Click **Validate Settings** to confirm the accuracy of the AccuRev server settings.

4. Click **Finish** to log in and dismiss the dialog box.

Removing an AccuWork Task Repository from Your Mylyn View

If you are no longer using an AccuWork task repository, you can remove it from your Mylyn view. Note that the following procedure only removes the repository from your view. The repository still exists and you can reconnect to it at a later time. See *Adding an AccuWork Task Repository* on page 71 for more information.

To remove an AccuWork task repository from your Mylyn view:

1. In Eclipse, open the Mylyn view by selecting **Window** > **Show View** > **Other** > **Mylyn** > **Task Repositories**.

   The Task Repositories tab is active.

2. Right-click the name of the AccuWork task repository and select **Delete**.

The task repository no longer appears in your Mylyn view. The repository still exists and you can reconnect to it at a later time. See *Adding an AccuWork Task Repository* for more information.

Working with AccuWork Queries

The following sections contain information about creating, running, and editing AccuWork queries when using Mylyn tasks.

Before running AccuWork queries in a Mylyn view, you must specify a default query in the AccuWork Query Browser by right-clicking an existing query and selecting **Set as Default** from the context menu.

> **Note:** The query results represent the AccuWork issues returned by both your Mylyn query and your AccuWork default query.

Creating an AccuWork Query

You can create and run AccuWork queries from within your Eclipse development environment by using the AccuRev Plug-In for Eclipse integration with Mylyn.

> **Note:** Before creating an AccuWork query in Mylyn, you must specify a default query in the AccuWork Query Browser by right-clicking an existing query and selecting **Set as Default** from the context menu.

To create an AccuWork query:

1. In Eclipse, open the Mylyn view by selecting **Window** > **Show View** > **Other** > **Mylyn** > **Task List**.

   The Task List tab is active.

2. On the Mylyn Task List tab, click the **New Task** option list and select **New Query**.

   The New Query dialog box appears:
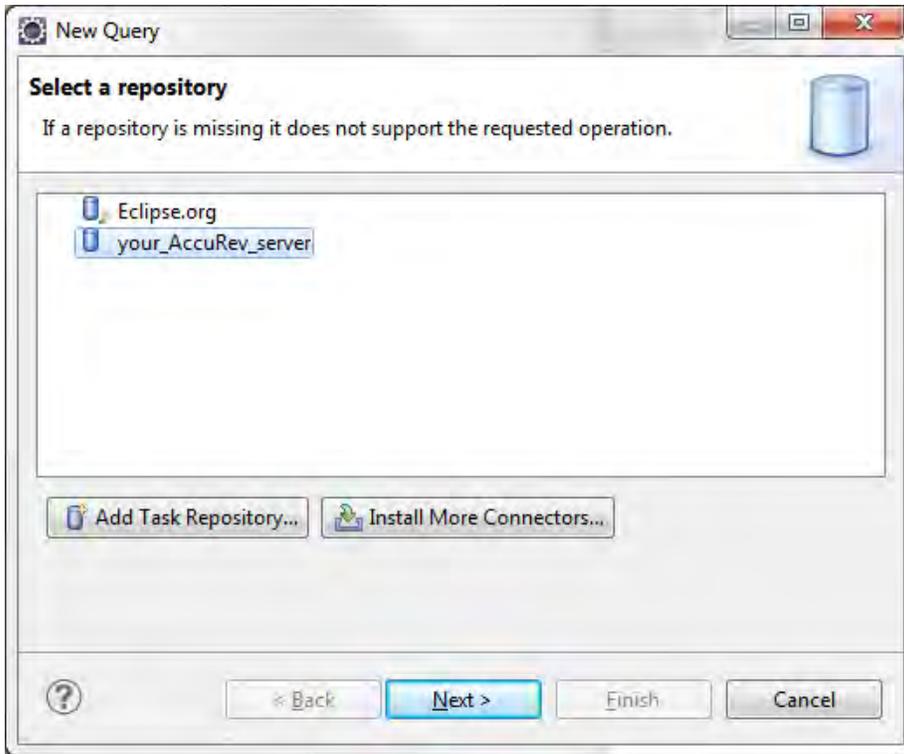
3. Select the name of your AccuRev server and click **Next**.

   The Edit Query dialog box appears:
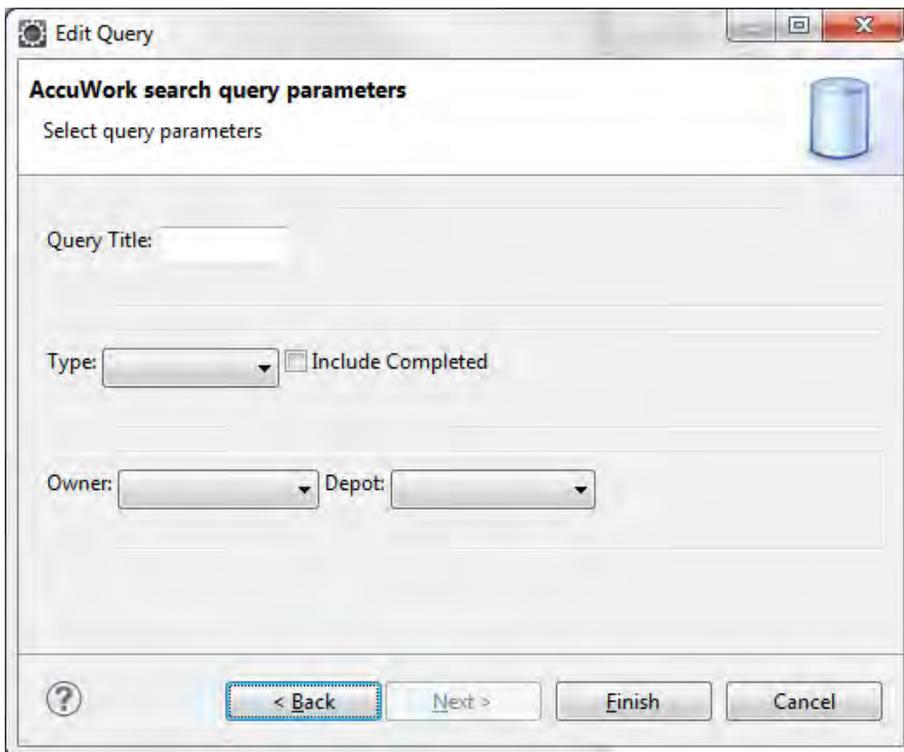
4. Specify settings in the Edit Query dialog box as described in the following table:

Option Name	Description
Query Title	A name for your new query.
Type	The kind of query you want to construct, such as: **defect**, **task**, **enhancement**, **story**, or **requirement**.
Include Completed	Indicates that you want the query to return defects, tasks, enhancements, stories, or requirements that have already been completed.
Owner	The owner of the query. Typically, you would select your own user name, unless you are assigning queries to other users.
Depot	The name of the AccuRev depot that you are querying.

5. Click **Finish**.

The new query (in this case, **query1**) appears in the Task List tab in your Mylyn view, followed by the name of your AccuRev server:

Running an AccuWork Query

To run an AccuWork query in a Mylyn view:

1. In Eclipse, open the Mylyn view by selecting **Window** > **Show View** > **Other** > **Mylyn** > **Task List**.

The Task List tab is active:
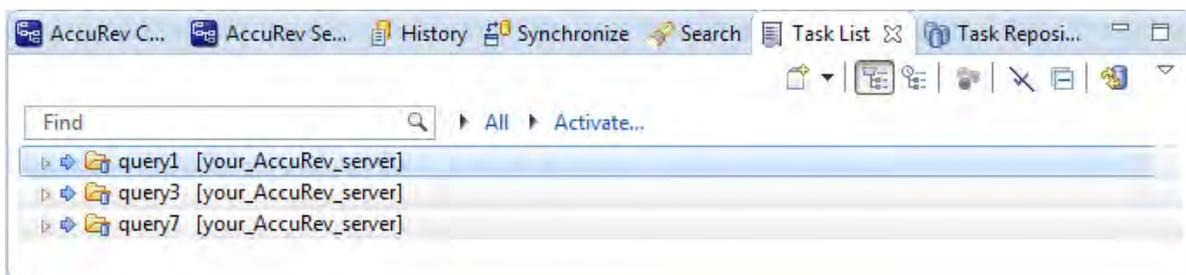
2. Select the query you want to run and press **Enter**, or double-click the name of the query.

If the query conditions are satisfied, the results of the query appear in the Mylyn Task List tab beneath the query:
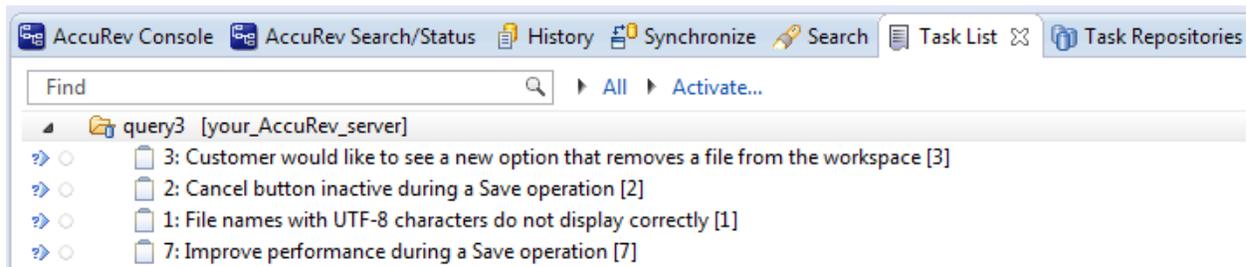
3. You can open any item that is returned by your query by selecting the item and pressing **Enter** or by double-clicking the item. You can also right-click the item and select an option from the context menu to perform a Mylyn Task operation.

Editing an AccuWork Query

To edit an AccuWork query:

1. In Eclipse, open the Mylyn view by selecting **Window** > **Show View** > **Other** > **Mylyn** > **Task List**.

   The Task List tab is active.

2. On the Mylyn Task List tab, right-click the name of the query and select **Properties** from the context menu.

   The Edit Query dialog box appears:
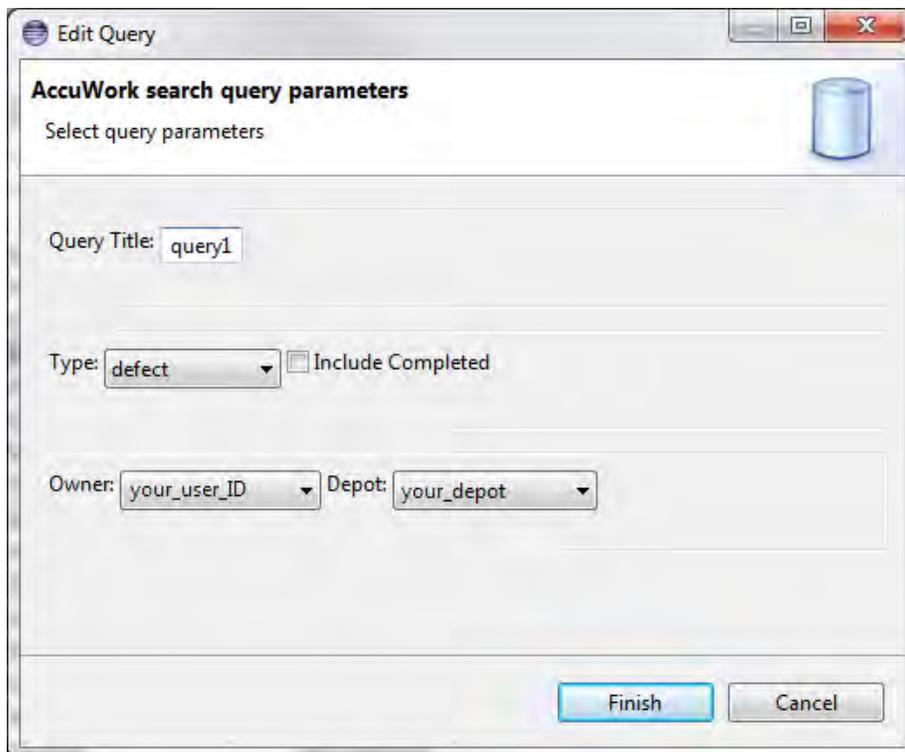
3. Change the settings to edit the query.

4. Click **Finish** to commit your changes.

A. The AccuRev Usage Model

This chapter introduces the basic AccuRev concepts and procedures that will enable you to be comfortable and productive using AccuRev on day one. More detailed information is always just a click away: pressing F1 or clicking the help button ( ) anywhere in the AccuRev GUI provides you with access to the entire AccuRev documentation set in both HTML and PDF formats.

The AccuRev® Usage Model

AccuRev's flexibility makes it easy to use for a variety of development scenarios. But like every software system, AccuRev has usage models that were foremost in the minds of its architects. This section describes the most common usage model.

AccuRev is a *software configuration management* (SCM) system, designed for use by a team of people who are developing a set of files. This set of files might contain source code, images, technical and marketing documents, audio/video tracks, or any other digital content the user puts into the system. The files and directories in AccuRev are said to be "version-controlled" or "under source control".

For maximum productivity, the team's members must be able to work independently of each other -- sometimes for just a few hours or days, other times for many weeks. Accordingly, each user has his own private copy of the version-controlled files. The private copies are stored on the user's own machine (or perhaps in the user's private area on a public machine), in a directory tree called a *workspace*. We can picture the independent workspaces for a three-user team comprised of John, Mary, and Derek as follows:

This set of users' workspaces uses the convention of having like names, suffixed with the individual usernames. AccuRev enforces this username-suffix convention. In this example, **talon_dvt** might mean "development work on the Talon product"; **john**, **mary**, and **derek** would be the users' login names.

From AccuRev's perspective, development work in this set of workspaces is a continual back-and-forth between getting "in sync" and "out of sync":

• Initially, the workspaces are completely synchronized: each workspace has copies of the same set of version-controlled files.

• The workspaces become unsynchronized as each user makes changes to some of the files.

• Periodically, users share their changes with each other. When **john** incorporates some or all of **mary**'s changes into his workspace, their two workspaces become more closely (perhaps completely) synchronized.

You might assume that the workspace synchronization process involves the direct transfer of data from one workspace to another. But this is not the way AccuRev organizes the work environment. Instead of transferring data directly between private areas (that is, between users' workspaces), AccuRev organizes the data transfer into two steps:

1. One user makes his changes public -- available to all the other members of his team. This step is called *promoting*.

2. Whenever they wish, other team members incorporate the public changes into their own workspaces. This step is called *updating*.
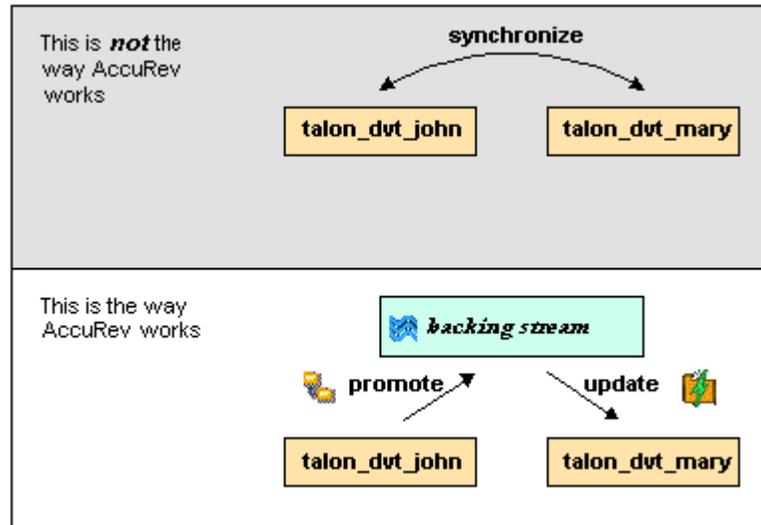
The first step involves a public data area, called a stream. A *stream* is an AccuRev structure with two very important features that support and simplify parallel development. First, as implied by the preceding illustration, numerous individual contributors can create workspaces on a single stream, allowing them to easily share their work with others, and to get changes from them when they wish. As you will see later, AccuRev has numerous safeguards that prevent one user from overwriting another user's changes, and tools that help you resolve conflicting changes when they occur. Second, in a stream hierarchy, *inheritance* lets child streams automatically inherit changes from the parent streams above them, reducing the need for tedious merges that might otherwise be required when several developers are working on the same code.

AccuRev has several kinds of streams, but the most common one is the backing stream. Later, we will show you how the data in this public stream "is in back of" or "provides a backstop for" all the private workspaces of the team members.

Change and Synchronization: The Four Basic Commands

With the usage model described above, you will be able to accomplish most of your AccuRev work with four simple commands: **Keep**, **Promote**, **Update**, and **Merge**. These commands are described in the following sections.

Keep: Preserving Changes in Your Private Workspace

An AccuRev workspace is just a normal directory tree, in which you make changes to version-controlled files. You can work with the files using text editors, build and test tools, IDEs, and so on, just as if the files were not version-controlled at all. For example, you might edit a source file and invoke the editor's **Save** command a dozen times over the course of an hour or two. These operations do not involve AccuRev at all -- the operating system changes the contents and the timestamp of the file in your workspace.

You do not need to perform a "check out" operation or otherwise get permission from AccuRev before editing a file in your workspace. (Some legacy SCM systems do impose such a regimen, and AccuRev can be configured to require checkouts, if your organization requires them.)

Every so often, you want AccuRev to preserve the current contents of the file as an official new *version* of the file. You accomplish this using AccuRev's **Keep** command. This figure shows how to invoke the **Keep** command from the File Browser toolbar; **Keep** is also available from the **Actions** and context (right-click) menus. Note that you can also keep new files, like **requirements.txt** in the preceding illustration -- its status is (external) because it has not yet been added to AccuRev.
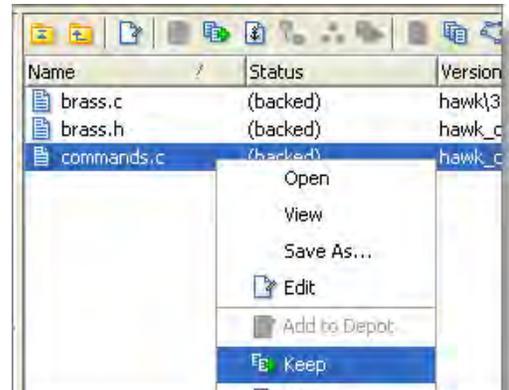
Tip: The File Browser helps organize your workspace using Outgoing Changes (your work), Incoming Changes (others' work), and Conflicts modes (conflicting changes in your work and that of a teammate).

You can continue modifying the file, using **Keep** to preserve the latest changes, as often as you like. Other team members will not complain about "thrashing" because these new versions stay within your workspace and do not affect any other user's workspace.

AccuRev retains all the versions that you **Keep**. This makes it possible for you to roll back to any previous version you created.

Several other operations are similar to **Keep**, in that they create a new version of a file in your workspace, without affecting any other user's workspace. The most important of these are:

- **Rename** and **Move**: You can rename a file or move it to a different directory (or both), using AccuRev commands. Other users will continue to see the file at its original pathname in their workspaces.

- **Defunct**: You can remove a file from your workspace with the AccuRev command **Defunct**. When you promote a defuncted file, other users will continue to see the file until they update their workspaces, at which point the file will be removed. (**Defunct** differs from a simple **Delete** in that **Defunct** removes the file from your workspace, while **Delete** removes the physical file from your local directory.)

More About Keep

We said earlier that AccuRev "retains all the versions that you **Keep**". But where? Each time you **Keep** a file, its current contents are copied to the AccuRev repository, located on the machine where the AccuRev Server runs. You do not need to care about the name and precise location of this copy. Each version you create has a *version-ID*, such as **velo_dfoster\2** (which translates as "the **2**nd version of this file created in workspace **velo_dfoster**").

AccuRev keeps track of the *status* of each file in a workspace. After you **Keep** a file, the Status column in the AccuRev File Browser contains the indicator **(kept)**. It also contains the indicator **(member)**, meaning that the file belongs to the set of files you are actively working on. (See *Active and Inactive Files* for more information.) The Version column displays the version-ID.

A change to the data within a file, recorded by **Keep**, is termed a *content change*; the change made by **Rename**, **Move** or **Defunct** is termed a *namespace change*. (Many SCM systems do not handle namespace changes at all, or have very limited capabilities in this area.) As noted previously, AccuRev saves a new copy of the file in the repository whenever you make a content change. But it does not need to copy the file when you make a namespace change; rather, the AccuRev Server just records the change in its database.

To perform version control on directories, AccuRev only needs to keep track of namespace changes -- renaming, moving, or deleting a directory. Unlike some legacy SCM systems, AccuRev does not need to record a new directory version when you make a content change -- for example, adding a new file to the directory.

Promote: Making Your Private Changes Public

At some point, after you have used **Keep** to create one or more new, private versions of a file in your workspace, you typically want to share the changes you have made with the other team members. To make your (most recent) new version "available to the public", you *promote* it. This figure shows how to invoke the **Promote** command from the File Browser toolbar; **Promote** is also available from the **Actions** and context (right-click) menus.

Tip: You can **Promote** files that you have not yet kept -- AccuRev performs the necessary **Keep** action for you. In the case of new files, AccuRev first adds the file to the AccuRev repository.

Promoting your new version of a file does not automatically "push" it into the workspaces of the other team members. When a user decides that she is ready to incorporate versions of files that other team members have promoted, she "pulls" them into her workspace with the **Update** command. This process is described in the following section, *Update: Incorporating Others' Changes into Your Workspace* on page 84.
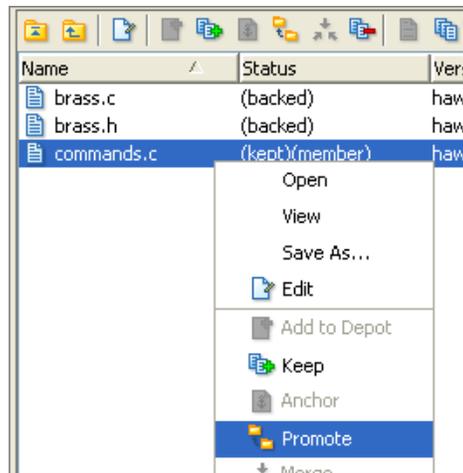
Streams

The **Promote** command sends data to -- and the **Update** command gets data from -- an AccuRev data structure called a *stream*. The stream (named **talon_dev** in this illustration) acts as a central data exchange for the set of workspaces used by a
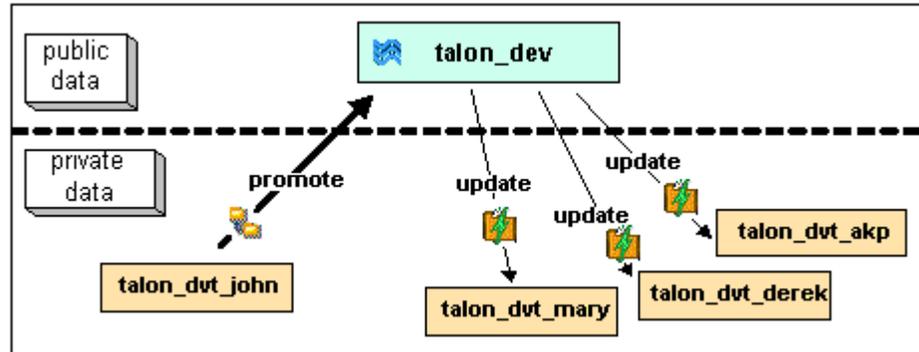

development team. A stream also has a bit of "traffic controller" built in, preventing team members' efforts from colliding and providing other mechanisms to control the flow of data.

A stream is not, as you might initially suppose, a set of copies of promoted files. Rather, it is more like a list of version-IDs.

- The 4th version created in workspace **talon_dvt_akp** of file **logic.xml**

- The 7th version created in workspace **talon_dvt_mary** of file **matrix.ini**

- and so on...

In SCM vernacular, a stream is a *configuration* of a collection of version-controlled files. The term "stream" is apt, because it implies the ongoing changes happening in a development project. Each time a user promotes a version of file **logic.xml**, the stream configuration changes for that file -- for example, from "the 5th version created in workspace **talon_dvt_derek**" to "the 7th version created in workspace **talon_dvt_mary**".

Promotion and Parallel Development

Sometimes, AccuRev does not allow you to promote a file to the development team's stream, because another team member has already promoted the same file (after modifying it in their own workspace). AccuRev prevents you from overwriting your colleague's change to the team's shared stream. This situation is called an *overlap*: two users working at the same time on the same goal, to create the stream's next version of a particular file.

Before you can promote your changes to the stream, you must first perform a *merge* on the file that has an overlap. This command is described in *Merge: When Changes Would Collide* on page 85.

Active and Inactive Files

As you work with a file using the commands described above, AccuRev considers the file to alternate between being *active* in your workspace and *inactive*:

- The file is initially *inactive*.

- When you create a new version in your workspace, using **Keep**, **Rename**, **Move**, or **Defunct**, the file becomes *active*.

- When you make your private version public, using the **Promote** command, the file becomes *inactive* again.

Later, you might restart this cycle, making the file active again by creating another new version of it. Alternatively, updating your workspace might overwrite your inactive file with a newer version that another team member promoted.

AccuRev keeps track of the set of active files in your workspace. Officially, this set is called the *default group*. You might find it easier to think of it as the workspace's "active group".

More About Promote

The **Promote** command does not copy the promoted version to the AccuRev repository. It does not need to. Promotion just gives an additional name to a version that *already exists* in the repository -- having been placed there by a previous **Keep** command (or **Rename**, **Move**, or **Defunct**). For example, promoting "the 7th version created in workspace **talon_dvt_mary**" might give that version the additional name "the 3rd version promoted to stream **talon_dvt**".

Just to emphasize the previous point: a stream does not reside in the file system, but in the database used by the AccuRev Server. Promoting a version to a stream does not create a copy of a file; it just creates an additional file-reference in the database.

It might seem strange at first that deleting a file with the **Defunct** command makes the file *active*. The File Browser continues to list the file -- with a **(defunct)** status -- even though the file has been removed from your workspace's disk storage. This design feature enables AccuRev to implement the file-deletion operation using the same private-change/public-change scheme as all other changes.

We have discussed *the* stream that is the basis for a set of workspaces. But a typical development project has many streams, organized into a hierarchy. Promoting a version to a higher-level stream from a lower-level stream makes that version "even more public" -- for example, available to users outside your local development team.

Update: Incorporating Others' Changes into Your Workspace

As users work independently of each other, the contents of their workspaces increasingly diverge. Typically, some of the differences between workspaces are inconsistencies. For example, changes that John makes in a report-library routine might cause errors in the report program that Mary is writing. To minimize the time and effort required to resolve inconsistencies during the integration phase of a project, it makes sense to have users synchronize their workspaces on a regular basis.

With AccuRev, synchronization does not mean incorporating data into your workspace directly from one or more other workspaces. Instead, synchronization involves copying data into the workspace from the stream to which all team members **Promote** their changes. This figure shows how to invoke the **Update** command from the File Browser toolbar; **Update** is also available from the **File** and context (right-click) menus.

Tip: You can **Update** your workspace from any File Browser mode, but the Incoming Changes mode shows you exactly how your workspace will change.
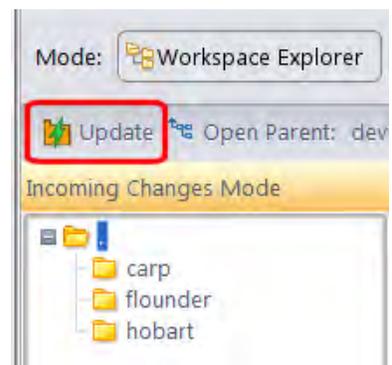
> **Note***: the stream's role as a provider of data -- through **Update** operations -- to a set of workspaces motivates the term *backing stream*. Think of restocking a store's shelves with merchandise retrieved from "the back room".

So updating your workspace copies versions of certain files from the backing stream to the workspace, overwriting/replacing the files currently in the workspace. But which files? **Update** changes a file if:

- There is a newer version in the backing stream, and

- The file is *not* currently active in your workspace.

Update will not overwrite an active file, even if there is a new version of it in the backing stream. No matter how good someone else's code is, you do not want his changes to wipe out the changes that you have been making! This situation is another instance of an *overlap*, which was introduced in *Promote: Making Your Private Changes Public* on page 82. You can encounter an overlap during a promote (if you are trying to make your private changes public), or during an update (if you are trying to bring already-public changes into your private workspace). You can use the *merge* operation to resolve all such overlap situations. See *Merge: When Changes Would Collide* on page 85.

Update handles namespace changes as well as content changes. Thus, if your colleague renamed a file and promoted the change, an update will cause the file to be renamed in your workspace. And if your colleague removed a file using the **Defunct** command, an update will cause the file to disappear from your workspace.

More About Update

Here is how AccuRev prevents an update from overwriting your changes: the first thing **Update** does is to analyze your workspace, determining whether each version-controlled file is "active" or "inactive". Initially, all the files in a workspace are inactive -- each one is a copy of some version in the repository. (For each version-controlled file, AccuRev keeps track of *which* particular version.)

A file is deemed to be active in your workspace if you have created a new version of it, using the **Keep**, **Rename**, **Move**, or **Defunct** command. (There are other AccuRev commands that "activate" a file.) When **Update** copies versions from the repository into your workspace, it skips over all active files.

> Note*:* **Update** can tell if you have modified a file but have not yet stored the changes in the repository as a new **Keep** version. It uses timestamps and checksums to determine this. The presence of such files prevents the update from proceeding if updating would overwrite one or more of them with the backing-stream version. You can use the **Anchor** command to activate such files, enabling **Update** to do its work.

Merge: When Changes Would Collide

The preceding sections on the **Promote** and **Update** commands both discuss the situation in which two users concurrently work on the same file. Their changes to the file are said to *overlap*. Both **Promote** and **Update** decline to process a file with overlap status, because doing so would cause one user's changes to overwrite another's changes.

For example:

- Team members John and Mary both Keep one or more new private versions of **logic.xml** in their respective workspaces.

- Mary Promotes her latest new version of **logic.xml** to the backing stream.

- At this point, AccuRev:

  - Will not allow John to **Promote** his version of **logic.xml** to the backing stream (there is no reason to assume that John's changes should take precedence over the changes Mary has already made and promoted).

  - Will not overwrite John's copy of **logic.xml** when he updates his workspace (again, AccuRev does not assume that a version in the backing stream should take precedence over changes made to the

version in the workspace). The **Update** command skips over this file, but continues its work on other files.

- "Flags" the overlap condition by highlighting **logic.xml** in yellow.

Before John can either promote or update **logic.xml**, he must incorporate, or merge, the version in the backing stream -- which contains Mary's changes -- into his own copy of the file. The **Merge** command is essentially a text editor, which combines the contents of two versions of a text file. The resulting "merged version" replaces the file in John's workspace.

This figure shows how to invoke the **Merge** command from the File Browser toolbar; **Merge** is also available from the **Actions** and context (right-click) menus.

Tip: Use the File Browser Conflicts mode to view overlap and other conditions that prevent files from being updated and promoted.

Often, a merge operation is unambiguous, and so can be performed automatically. For example, suppose Mary's changes to file **logic.xml** all occur in lines 30–50, and all of John's changes occur in lines 125–140. In this case, merging the two versions involves replacing some or all of John's 20 lines with Mary's. Now, the edited version of **logic.xml** in John's workspace contains both users' changes.
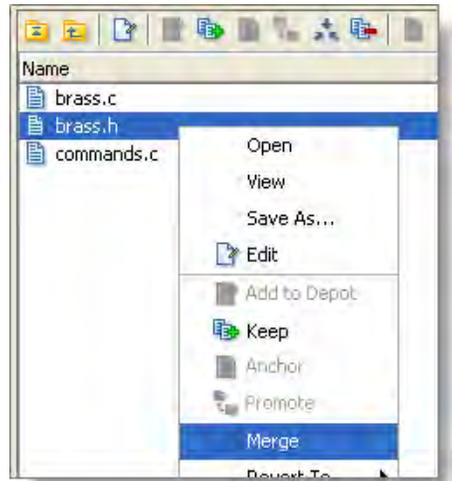
> **Note***: We do not claim that the two sets of changes are semantically consistent with each other. That is what the build-and-test cycle is for!

If both John and Mary have made changes to the same part of the file -- say, lines 2-10 -- then John must decide how to resolve this *conflict*. The graphical **Merge** tool makes this easy:
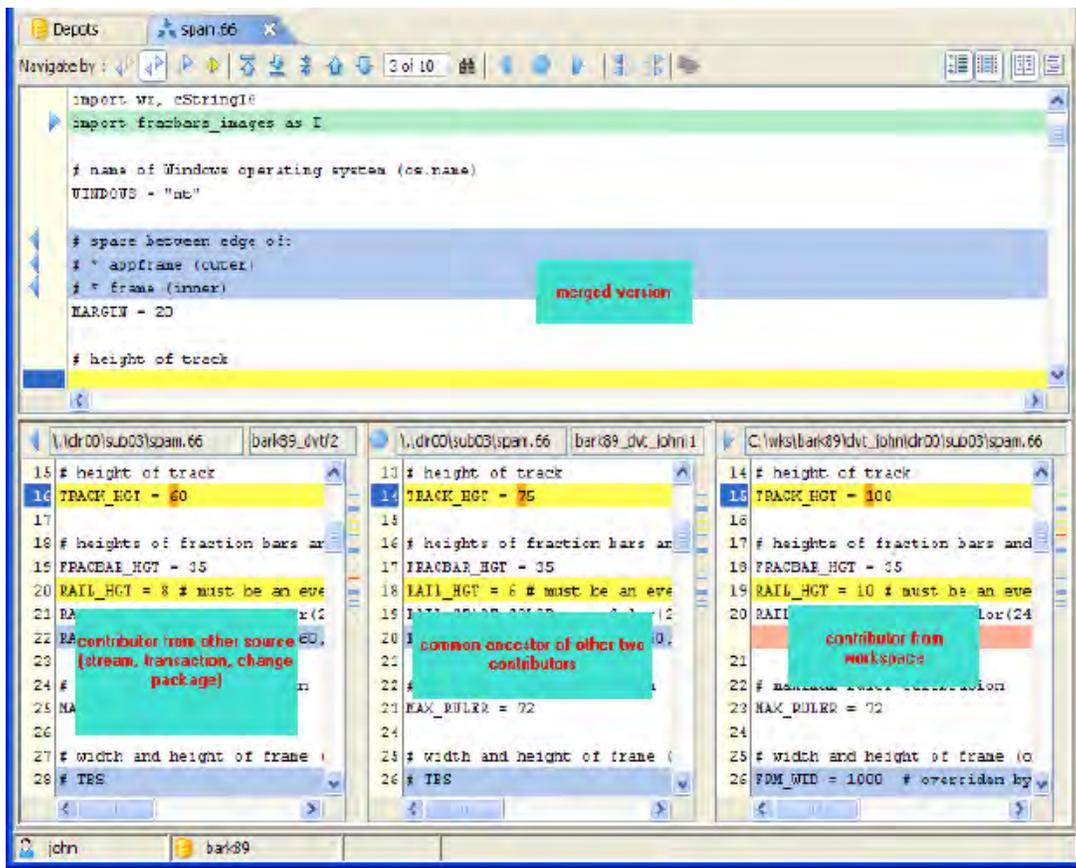
After performing a merge, AccuRev automatically Keeps the merged version to preserve the results of the merge operation. You can then Promote the merged version to the backing stream. After that, other team members can use Update -- perhaps in conjunction with Merge -- to bring all the changes into their workspaces.

More About Merge

The graphical Merge tool performs a "three-way merge", which uses the common ancestor of the two versions being merged. This algorithm helps to automate the merge operation, often completely eliminating the need for human intervention. AccuRev performs merge operations on text files only. Binary files are "merged" by choosing which version to take.

AccuRev keeps track of all merge operations. This greatly simplifies subsequent merge operations on files that have been merged previously: you do not need to resolve the same conflicts over and over again.

The most common overlap situation happens when AccuRev prevents you from promoting a file, because someone else "got there first" in creating a version in the backing stream. AccuRev can also detect *deep overlaps*, in which another user "got there first" in creating a version in the *parent* of the backing stream, or in other higher-level streams.

AccuRev® Plug-In for Eclipse User's Guide