

Running Eclipse

Table of Contents

<u>Running Eclipse</u>	1
<u>Setting the workspace location to be inside the current working directory</u>	1
<u>Setting a specific location for the workspace with -data</u>	1
<u>Setting the java VM using -vm</u>	1
<u>Advanced Topics in Running Eclipse</u>	1
<u>Running on Different VMs</u>	5
<u>Running Eclipse on J9</u>	5
<u>Running Eclipse on the IBM Developer Kit, Java(TM) Technology Edition VM</u>	5
<u>Upgrading Eclipse</u>	6
<u>Users who don't use "-data"</u>	6
<u>Users who do use "-data"</u>	7
<u>Adding third party plug-ins</u>	7
<u>Working with perspectives</u>	8
<u>Switching between perspectives</u>	9
<u>Specifying the default perspective</u>	10
<u>Opening perspectives</u>	11
<u>Changing where perspectives open</u>	12
<u>Configuring perspectives</u>	13
<u>Saving a user defined perspective</u>	14
<u>Deleting a user defined perspective</u>	15
<u>Resetting perspectives</u>	16
<u>Working with views and editors</u>	17
<u>Opening views</u>	18
<u>Moving and docking views</u>	19
<u>Rearranging tabbed views</u>	20
<u>Creating fast views</u>	21
<u>Working with fast views</u>	22
<u>Opening files for editing</u>	24

Table of Contents

<u>Associating editors with file types</u>	25
<u>Editing files outside the Workbench</u>	26
<u>Tiling editors</u>	27
<u>Maximizing a view or editor</u>	28
<u>Customizing the Workbench</u>	29
<u>Rearranging the main toolbar</u>	30
<u>Changing the key bindings</u>	31
<u>Changing fonts and colors</u>	32
<u>Fonts</u>	32
<u>Colors</u>	32
<u>Changing the placement of the tabs</u>	34
<u>Controlling single and double click behavior</u>	35
<u>Working with projects, folders and files</u>	36
<u>Creating a project</u>	37
<u>Closing projects</u>	38
<u>Deleting projects</u>	39
<u>Creating a folder</u>	40
<u>Creating a file</u>	41
<u>Creating linked resources</u>	42
<u>Moving resources</u>	44
<u>Copying resources</u>	45
<u>Renaming resources</u>	46
<u>Deleting resources</u>	47
<u>Viewing resource properties</u>	48

Table of Contents

<u>Navigating and finding resources</u>	49
<u>Finding a resource quickly</u>	50
<u>Searching for files</u>	51
<u>Searching for text within a file</u>	52
<u>Sorting resources in the Navigator view</u>	53
<u>Showing or hiding files in the Navigator view</u>	54
<u>Narrowing the scope of the Navigator view</u>	55
<u>Linking the Navigator view to the active editor</u>	56
<u>Bookmarks, tasks and other markers</u>	57
<u>Creating a bookmark within a file</u>	58
<u>Creating a bookmark for an entire file</u>	59
<u>Deleting a bookmark</u>	60
<u>Adding line items in the Tasks view</u>	61
<u>Associating a task with a resource</u>	62
<u>Deleting tasks</u>	63
<u>Filtering the Tasks view</u>	64
<u>Automatically fixing problems</u>	65
<u>Comparing resources</u>	66
<u>Setting preferences for comparing files</u>	68
<u>Understanding the comparison</u>	70
<u>Merging changes in the compare editor</u>	72
<u>Working with local history</u>	74
<u>Comparing resources with the local history</u>	75

Table of Contents

<u>Replacing a resource with local history</u>	76
<u>Restoring deleted resources from local history</u>	77
<u>Setting local history preferences</u>	78
<u>Importing</u>	79
<u>Importing existing projects</u>	80
<u>Importing resources from the file system</u>	81
<u>Importing resources from a ZIP file</u>	82
<u>Exporting</u>	83
<u>Exporting resources to the file system</u>	84
<u>Exporting resources to a ZIP file</u>	85
<u>Building resources</u>	86
<u>Performing builds automatically</u>	87
<u>Performing builds manually</u>	88
<u>Saving resources automatically before a manual build</u>	89
<u>Changing build order</u>	90
<u>Running external tools</u>	91
<u>Running Ant buildfiles</u>	92
<u>Modifying the Ant classpath</u>	93
<u>Using a different version of Ant</u>	94
<u>Adding new Ant tasks and types</u>	96
<u>Using the help system</u>	97
<u>Accessing context-sensitive help</u>	98
<u>Accessing and navigating online help</u>	99
Maximizing help frames.....	99
Printing online help.....	99

Table of Contents

<u>Searching online help</u>	100
<u>Refining the search results</u>	101
<u>Search index generation</u>	101
<u>Method 1: Using the context menu</u>	102
<u>Method 2: Using the synchronize action</u>	103
<u>From within the synchronize view</u>	104
<u>Setting help fonts and colors for accessibility</u>	105
<u>Changing the web browser used by the help system</u>	106
<u>Working in the team environment</u>	107
<u>Working with a CVS repository</u>	108
<u>Creating a CVS repository location</u>	109
<u>Discarding a CVS repository location</u>	111
<u>Refreshing the CVS Repositories view</u>	112
<u>Discovering branch and version tags</u>	113
<u>Changing the properties of a CVS repository location</u>	114
<u>Setting the content type of a file extension</u>	116
<u>Working with projects shared with CVS</u>	117
<u>Enabling the CVS resource decorations</u>	118
<u>Sharing a new project using CVS</u>	119
<u>Project checked out with another CVS tool</u>	120
<u>Checking out a project from a CVS repository</u>	121
<u>Checking out a module from a CVS repository</u>	122
<u>Checking out a folder into an existing project</u>	124
<u>Disconnecting a project from CVS</u>	125
<u>Setting the CVS keyword substitution mode</u>	126
<u>Filtering in the CVS Resource History View</u>	126

Table of Contents

<u>Changing the sharing of a project</u>	126
<u>Synchronizing with the repository</u>	126
<u>Updating</u>	128
<u>Committing</u>	131
<u>Version control life cycle: adding and ignoring resources</u>	133
<u>Adding a file to version control</u>	133
<u>How may I ignore thee, let me count the ways</u>	133
<u>Creating a global ignore pattern</u>	135
<u>Authoring the CVS .cvsignore file</u>	136
<u>Overriding or removing resource ignore patterns</u>	137
<u>Resolving conflicts</u>	138
<u>Manually merging changes</u>	138
<u>Auto merging changes</u>	139
<u>Merge actions</u>	140
<u>Branching</u>	141
<u>Merging from a branch</u>	141
<u>Viewing a file's revision history</u>	143
<u>Versioning</u>	144
<u>Creating a version of a project</u>	145
<u>Versioning projects in the repository</u>	146
<u>Comparing resources with repository versions</u>	147
<u>Working with patches</u>	148
<u>To create a patch from a CVS project</u>	148
<u>To apply a patch</u>	148
<u>Options for applying a patch</u>	149
<u>Setting up Watches</u>	150
<u>Setting up a Project for Watch/Edit</u>	151
<u>Editing</u>	151
<u>Unediting</u>	151
<u>Moving a tag on a single file</u>	151
<u>Moving a tag from within the repositories view</u>	152
<u>Compatibility</u>	153

Table of Contents

<u>Working with patches</u>	
<u>Don't forget to refresh!</u>	154
<u>Caveats</u>	155
<u>Replacing resources in the Workbench</u>	156
<u>Finding out who's working on what: watch/edit</u>	156
<u>Changing CVS team settings</u>	156
<u>Restoring deleted files from the repository</u>	157
<u>Reverting a branch to a previous version</u>	157
<u>Moving version tags</u>	157
<u>Running the CVS command-line client outside of Eclipse</u>	157
<u>Updating features with the update manager</u>	159
<u>Inspecting the current configuration</u>	160
<u>Saving a configuration</u>	161
<u>Restoring a saved configuration</u>	162
<u>Picking up pending changes</u>	163
<u>Installing new features with the update manager</u>	164
<u>Installing several features at once using group updates</u>	166
<u>Searching for features</u>	168

Running Eclipse

After you install (unzip) the Eclipse driver in a directory (such as `c:\eclipse`), start the Workbench by running the Eclipse executable file found in the top level install directory. The executable file is called `eclipse.exe` on Windows systems and `eclipse` on Linux systems. **Note:** the following discussion describes setting up on Windows systems. Setup on Linux is analogous.

If you do not specify otherwise, the platform creates a default workspace directory as a sibling of the executable (for example, `c:\eclipse\workspace`). This workspace directory is used as the default content area for your projects as well as for holding any required metadata. For shared or multi-workspace installs you should explicitly state the location of your workspace rather than using the default. There are two ways to control the location of your workspace: using the current working directory or using the `-data` command line argument.

Setting the workspace location to be inside the current working directory

In this scenario, the workspace location will be a directory called `workspace` inside the current working directory.

Perhaps the easiest way of doing this is to create a shortcut using the following steps:

1. Navigate to `eclipse.exe` in the Windows Explorer and using a right button drag, create a shortcut to `eclipse.exe`.
2. Edit the properties of the shortcut such that the **Start in:** field identifies the parent directory of your workspace location (for example, `c:\users\robert`).
3. Close the properties dialog and double-click on the shortcut (if the provided directory was `c:\users\robert`, the workspace location would be `c:\users\robert\workspace`).

Of course you can get the same effect using a command prompt by changing directory to your workspace parent's directory and then running `eclipse.exe`.

Setting a specific location for the workspace with `-data`

To use the `-data` command line argument, simply add `-data your_workspace_location` (for example, `-data c:\users\robert\myworkspace`) to the **Target** field in the shortcut properties, or include it explicitly on your command line.

Setting the java VM using `-vm`

It is recommended that you explicitly specify which Java VM to use when running Eclipse. This is achieved with the `-vm` command line argument (for example, `-vm c:\jre\bin\javaw.exe`). If you don't use `-vm`, Eclipse will use the first Java VM found on the O/S path. When you install other products, they may change your path, resulting in a different Java VM being used when you next launch Eclipse.

Advanced Topics in Running Eclipse

The Eclipse executable and the platform itself offer a number of execution options of interest to people developing or debugging parts of Eclipse. The general form of running the Eclipse executable is:

Running Eclipse

eclipse [platform options] [-vmargs [Java VM arguments]]

Eclipse Startup Parameters

Command	Description	Since
<code>-arch</code> <i>architecture</i>	Defines the processor architecture on which the Eclipse platform is running. The Eclipse platform ordinarily computes the optimal setting using the prevailing value of <code>Java.os.arch</code> property. If specified here, this is the value that the Eclipse platform uses. The value specified here is available to plug-ins as <code>BootLoader.getOSArch()</code> . Example values: "x86", "sparc", "PA-RISC", "ppc".	2.0
<code>-application</code> <i>applicationId</i>	The application to run. Applications are declared by plug-ins supplying extensions to the <code>org.eclipse.core.runtime.applications</code> extension point. This argument is typically not needed. If specified, the value overrides the value supplied by the configuration. If not specified, the Eclipse Workbench is run.	1.0
<code>-boot</code> <i>bootJarURL</i>	<i>(Deprecated; replaced by <code>-configuration</code>; supported for 1.0 compatibility)</i> . The location of the Eclipse platform's boot plug-in code (<code>boot.jar</code>), expressed as a URL. If specified, it is used to set the classpath for the class loader that loads the Eclipse platform bootstrap class loader. Only required when changing the relative location of <code>startup.jar</code> and <code>boot.jar</code> . Note that relative URLs are not allowed.	*1.0
<code>-classloaderproperties</code> [<i>file</i>]	Activates platform class loader enhancements using the class loader properties file at the given location, if specified. The file argument can be either a file path or a URL. Note that relative URLs are not allowed. Click here for more details.	2.0.2
<code>-configuration</code> <i>configurationFileURL</i>	The location for the Eclipse Platform configuration file, expressed as a URL. The configuration file determines the location of the Eclipse platform, the set of available plug-ins, and the primary feature. Note that relative URLs are not allowed. The configuration file is written to this location when the Eclipse platform is installed or updated.	2.0
<code>-consolelog</code>	Mirrors the Eclipse platform's error log to the console used to run Eclipse. Handy when combined with <code>-debug</code> .	1.0
<code>-data</code> <i>workspacePath</i>	The path of the workspace on which to run the Eclipse platform. The workspace location is also the default location for projects. Relative paths are interpreted relative to the directory that Eclipse was started from.	1.0
<code>-debug</code> [<i>optionsFile</i>]	Puts the platform in debug mode and loads the debug options from the file at the given location, if specified. This file indicates which debug points are available for a plug-in and whether or not they are enabled. If a file	1.0

Running Eclipse

	location is not given, the platform looks in the directory that eclipse was started from for a file called ".options". Both URLs and file system paths are allowed as file locations.	
-dev [<i>classpathEntries</i>]	Puts the platform in development mode. The optional classpath entries (a comma separated list) are added to the runtime classpath of each plug-in. For example, when the workspace contains plug-ins being developed, specifying -dev bin adds a classpath entry for each plug-in project's directory named <code>bin</code> , allowing freshly generated class files to be found there. Redundant or non-existent classpath entries are eliminated.	1.0
-endsplash <i>params</i>	Internal option for taking down the splash screen when the Eclipse platform is up and running. This option has different syntax and semantics at various points along the splash screen processing chain.	2.0
-feature <i>featureId</i>	The ID of the primary feature. The primary feature gives the launched instance of Eclipse its product personality, and determines the product customization information used.	2.0
-keyring <i>keyringFilePath</i>	The location of the authorization database (or "key ring" file) on disk. This argument must be used in conjunction with the -password option. Relative paths are interpreted relative to the directory that Eclipse was started from.	1.0
-nl <i>locale</i>	Defines the name of the locale on which the Eclipse platform is running. The Eclipse platform ordinarily computes the optimal setting automatically. If specified here, this is the value that the Eclipse platform uses. The value specified here is available to plug-ins as <code>BootLoader.getNL()</code> . Example values: "en_US" and "fr_FR_EURO".	2.0
-nolazyregistrycacheloading	Deactivates platform plug-in registry cache loading optimization. By default, extensions' configuration elements will be loaded from the registry cache (when available) only on demand, reducing memory footprint. This option will force the registry cache to be fully loaded at startup.	2.1
-noregistrycache	Bypasses the reading and writing of an internal plug-in registry cache file.	2.0
-nosplash	Runs the platform without putting up the splash screen.	1.0
-os <i>operatingSystem</i>	Defines the operating system on which the Eclipse platform is running. The Eclipse platform ordinarily computes the optimal setting using the prevailing value of <code>Java os.name</code> property. If specified here, this is the value that the Eclipse platform uses. The value specified here is available to plug-ins as <code>BootLoader.getOS()</code> , and used to resolve occurrences of the <code>\$OS\$</code> variable in paths mentioned in the plug-in manifest file. Example values:	1.0

Running Eclipse

	"win32", "linux", "hpux", "solaris", "aix".	
<code>-password password</code>	The password for the authorization database. Used in conjunction with the <code>-keyring</code> option.	1.0
<code>-perspective perspectiveld</code>	The perspective to open in the active workbench window on startup. If this parameter is not specified, the perspective that was active on shutdown will be opened.	1.0
<code>-plugincustomization propertiesFile</code>	The location of a properties file containing default settings for plug-in preferences. These default settings override default settings specified in the primary feature. Relative paths are interpreted relative to the directory that eclipse was started from.	2.0
<code>-plugins pluginsFileURL</code>	<i>(Deprecated; replaced by <code>-configuration</code>; supported for 1.0 compatibility)</i> . The location of the file that specifies where the Eclipse platform finds plug-ins, expressed as a URL. The file is in property file format where the keys are arbitrary user defined names and the values are comma separated lists of either explicit paths to plugin.xml files, or paths to directories containing plug-ins. Note that relative URLs are not allowed. If specified, this option causes the creation of a suitable temporary configuration.	*1.0
<code>-refresh</code>	Option for performing a global refresh of the workspace on startup. This will reconcile any changes that were made in the file system since the platform was last run.	1.0
<code>-showlocation</code>	Option for displaying the location of the workspace in the window title bar. In release 2.0 this option only worked in conjunction with the <code>-data</code> command line argument.	2.0
<code>-showsplash params</code>	Internal option for showing the splash screen (done by the executable Eclipse platform launcher). This option has different syntax and semantics at various points along the splash screen processing chain.	2.0
<code>-vm vmPath</code>	The location of Java Runtime Environment (JRE) to use to run the Eclipse platform. If not specified, the JRE is at <code>jre</code> , sibling of the Eclipse executable. Relative paths are interpreted relative to the directory that eclipse was started from.	1.0
<code>-ws windowSystem</code>	Defines the window system on which the Eclipse platform is running. The Eclipse platform ordinarily computes the optimal setting using the prevailing value of Java <code>os.name</code> property. If specified here, this is the value that the Eclipse platform uses. The value specified here is available to plug-ins as <code>BootLoader.getWS()</code> , used to configure SWT, and used to resolve occurrences of the <code>\$ws\$</code> variable in paths mentioned in the plug-in manifest file. Example values: "win32", "motif", "gtk".	1.0

All arguments following (but not including) the `-vmargs` entry are passed directly through to the indicated Java VM as virtual machine arguments (that is, before the class to run). **Note:** If an Eclipse startup argument, such as `-data`, is provided after the Java vm arguments (`-vmargs`), Eclipse will not start and you will receive a "JVM terminated. Exit code=1" error.

Running on Different VMs

Running Eclipse on J9

When running Eclipse on J9 version 1.5, it is recommended that you use the following VM options:

```
eclipse.exe [eclipse arguments] -vm path_to_j9w.exe  
-vmargs -ms:32 -mm:2048 -mo:32768 -moi:32768 -mca:32 -mco:128 -mx:2000000
```

When running Eclipse on J9 version 2.0, the default arguments chosen by J9W should be suitable. However, to override the parameters which are automatically set internally by the Eclipse executable, you must specify `-vmargs` with no following arguments as follows:

```
eclipse.exe [eclipse arguments] -vm path_to_j9w.exe -vmargs
```

Please refer to the J9 VM documentation and help for further information.

Running Eclipse on the IBM Developer Kit, Java(TM) Technology Edition VM

The default VM settings for IBM Developer Kit, Java(TM) Technology Edition 1.3 Linux work well for initial exploration, but are not sufficient for large scale development. For large scale development you should modify your VM arguments to make more heap available. For example, the following setting will allow the Java heap to grow to 256MB:

```
-vmargs -Xmx256M
```

© Copyright IBM Corporation and others 2000, 2003.

Upgrading Eclipse

If you are upgrading to a newer release of Eclipse from an older release, there are simple steps to follow to migrate your workspace to the new release. Your workspace is the directory on disk that contains all of your project files, as well as meta-data such as preferences you may have customized. The steps to follow for upgrading depend on whether or not you used the "-data" command line argument when starting Eclipse. The "-data" argument is recommended because it clearly specifies the location of your workspace. If this argument is not used, Eclipse will place the workspace in the current working directory at the time Eclipse was launched.

Tip: It doesn't hurt to make a backup of your workspace before upgrading. After you've upgraded your workspace, you won't be able to use it again with an older version of Eclipse. If you ever want to go "back in time" to an earlier release, you'll need that backup!

Users who don't use "-data"

If you weren't previously using "-data" to specify your workspace, follow these steps to upgrade:

1. Find the workspace directory used by your old version of Eclipse. Typically this is located inside the eclipse install directory in a sub-directory called "workspace". If you were using a shortcut or script to launch Eclipse, then it will be under the current working directory of that shortcut or script in a sub-directory called "workspace". Windows users, this is specified by the "Start in" argument in your shortcut properties.
2. Copy this workspace directory into a new, empty location outside of any Eclipse install directory.
3. Install the new version of Eclipse in a new location, separate from any old version of Eclipse.
4. Start this new version of Eclipse, using the "-data" command line argument to point to the workspace location.

Example:

Old Eclipse install directory:

```
c:\eclipse2_0
```

New Eclipse install directory:

```
c:\eclipse2_1
```

Workspace location:

```
c:\data\EclipseWorkspace
```

Java VM location:

```
c:\jre
```

Command line to start Eclipse:

```
"c:\eclipse2_1 -data c:\data\EclipseWorkspace -vm d:\jre\bin\javaw"
```

Tip: It's generally a good idea to explicitly specify which Java VM to use when running Eclipse. This is achieved with the "-vm" command line argument as illustrated above. If you don't use "-vm", Eclipse will look on the O/S path. When you install other products, they may change your path, resulting in a different Java VM being used when you next launch Eclipse.

Users who do use "--data"

If you were previously using the "--data" argument to start Eclipse, your upgrade path is much easier:

1. Install the new version of Eclipse in a new location, separate from any old version of Eclipse.
2. Start this new version of Eclipse, using the "--data" command line argument to point to your old workspace location.

See the example in the previous section for an illustration.

Adding third party plug-ins

If you have installed extra plug-ins in your Eclipse environment, you will need to add these new plug-ins to each new build or version of Eclipse you install. Before you do this, refer to the documentation for those plug-ins to ensure they are compatible with the version of Eclipse you are moving to. There are several ways to add these extra plug-ins to your new Eclipse install:

1. Copy the directories for each plug-in into the "plugins" directory of your new Eclipse version.
2. Use an Eclipse update site to re-install those extra plug-ins or features in the new version of Eclipse.
3. If you are using product extensions, simply copy the "links" directory into the new Eclipse version install directory. For more details on product extensions, see the documentation in the Platform Plug-in Developer Guide, under **Programmer's Guide > Packaging and delivering Eclipse based products > Product extensions**.

© Copyright IBM Corporation and others 2000, 2003.

Working with perspectives

Perspectives define the initial set and layout of views in the Workbench window. They provide a set of capabilities aimed at accomplishing a specific type of task or working with specific types of resources.

See the Related tasks links for more details.

■ Related concepts

[Perspectives](#)

[Views](#)

[Fast views](#)

■ Related tasks

[Switching between perspectives](#)

[Specifying the default perspective](#)

[Opening perspectives](#)

[Changing where perspectives open](#)

[Configuring perspectives](#)

[Saving a user defined perspective](#)

[Deleting a user defined perspective](#)

[Resetting perspectives](#)

© Copyright IBM Corporation and others 2000, 2003.

Switching between perspectives

Open perspectives are represented by icons on the shortcut bar (the toolbar at the left of the Workbench window). When you have more than one perspective open, you can switch between them by clicking the icons on the shortcut bar.

■ Related concepts

[Perspectives](#)

■ Related tasks

[Opening perspectives](#)

[Changing where perspectives open](#)

© Copyright IBM Corporation and others 2000, 2003.

Specifying the default perspective

The default perspective always shows up on the **Window > Open Perspective** menu. The Resource perspective is the pre-defined default perspective.

To change the default perspective:

1. From the main menu bar, select **Window > Preferences**.
2. Expand the **Workbench** category on the left and select **Perspectives**. The Perspectives preferences page opens.
3. Select the perspective that you want to define as the default, and click **Make Default**. The default indicator moves to the perspective that you selected.
4. Click **OK**.

■ Related concepts

[The Workbench Perspectives](#)

■ Related tasks

[Opening perspectives](#)
[Changing where perspectives open](#)
[Configuring perspectives](#)

© Copyright IBM Corporation and others 2000, 2003.

Opening perspectives

Perspectives provide combinations of views and editors that are suited to performing a particular set of tasks. For example, you would normally open the Debug perspective to debug a Java program.

To open a new perspective:

1. Click the **Open Perspective** button  on the shortcut bar on the left side of the Workbench window. (This provides the same function as the **Window > Open Perspective** menu on the menu bar.)
2. To see a complete list of perspectives, select **Other** from the drop-down menu.
3. Select the perspective that you want to open.

When the perspective opens, the title bar of the window it is in changes to display the name of the perspective. In addition, an icon is added to the shortcut bar, allowing you to quickly switch back to that perspective from other perspectives in the same window.

By default, a perspective will open in the same window. If you would rather it opened in a new window, change the setting in **Window > Preferences > Workbench > Perspectives**.

■ Related concepts

[Perspectives](#)

■ Related tasks

[Opening views](#)

[Changing where perspectives open](#)

[Specifying the default perspective](#)

[Switching between perspectives](#)

© Copyright IBM Corporation and others 2000, 2003.

Changing where perspectives open

You can change the default behavior for how perspectives are opened in the Workbench.

1. From the main menu bar, select **Window > Preferences**.
2. Expand the **Workbench** category on the left and select **Perspectives**. The Perspectives preferences page opens.
3. Select either **In the same window** or **In a new window** from the **Open a new perspective** group.
4. Click **OK**.

■ Related concepts

[Perspectives](#)

■ Related tasks

[Opening perspectives](#)

[Specifying the default perspective](#)

[Switching between perspectives](#)

[Configuring perspectives](#)

© Copyright IBM Corporation and others 2000, 2003.

Configuring perspectives

In addition to configuring the layout of your perspective you can also control several other key aspects of a perspective. These include:

- The options available on the **File > New** submenu.
- The options available on the **Window > Open Perspective** submenu.
- The options available on the **Window > Show View** submenu.
- Action sets (buttons and menu options) that show up on the toolbar and menu bar.

To configure a perspective:

1. Switch to the perspective that you want to configure.
2. Select **Window > Customize Perspective**.
3. Expand the item that you want to customize.
4. Use the check boxes to select which elements you want to see on drop-down menus in the selected perspective. Items you do not select will still be accessible by clicking the **Other** menu option.
5. Click **OK**.

■ Related concepts

[Perspectives](#)

■ Related tasks

[Changing where perspectives open](#)

[Specifying the default perspective](#)

[Saving a user defined perspective](#)

© Copyright IBM Corporation and others 2000, 2003.

Saving a user defined perspective

If you have modified a perspective by adding, deleting, or moving (docking) views, you can save your changes for future use.

1. Switch to the perspective that you want to save.
2. Click **Window > Save Perspective As**.
3. Type a new name for the perspective into the **Name** field.
4. Click **OK**.

The name of the new perspective is added to the **Window > Open Perspective** menu.

■ Related concepts

[Perspectives](#)

[Views](#)

■ Related tasks

[Resetting perspectives](#)

[Deleting a user defined perspective](#)

[Opening views](#)

[Moving and docking views](#)

© Copyright IBM Corporation and others 2000, 2003.

Deleting a user defined perspective

You can delete perspectives that you defined yourself, but not those that are delivered with the Workbench.

1. On the main menu bar, click **Window > Preferences**. The Preferences window opens.
2. Expand the **Workbench** category and click **Perspectives**.
3. From the **Available perspectives** list, select the one that you want to delete and click **Delete**.
4. Click **OK**.

■ Related concepts

[Perspectives](#)

■ Related tasks

[Saving a user defined perspective](#)

[Resetting perspectives](#)

© Copyright IBM Corporation and others 2000, 2003.

Resetting perspectives

To restore a perspective to its original layout:

1. Click **Window > Preferences**.
2. Expand **Workbench** and choose **Perspectives**.
3. From the **Available perspectives** list, select the perspective you want to restore.
4. Click **Reset**.
5. Click **OK**.

■ Related concepts

[Perspectives](#)

■ Related tasks

[Saving a user defined perspective](#)

© Copyright IBM Corporation and others 2000, 2003.

Working with views and editors

Views and editors are the main visual entities which appear in the Workbench. In any given perspective there is a single editor area, which can contain multiple editors, and a number of surrounding views which provide context.

The Workbench provides a number of operations for working with views and editors. See the Related tasks links for more details.

■ Related concepts

[Views](#)

[Editors](#)

[Fast views](#)

[Perspectives](#)

■ Related tasks

[Opening views](#)

[Moving and docking views](#)

[Rearranging tabbed views](#)

[Creating fast views](#)

[Opening files for editing](#)

[Associating editors with file types](#)

[Editing files outside the Workbench](#)

[Tiling editors](#)

[Maximizing a view or editor](#)

© Copyright IBM Corporation and others 2000, 2003.

Opening views

Perspectives offer pre-defined combinations of views and editors. To open a view that is not included in the current perspective, select **Window > Show View** from the main menu bar.

You can create *fast views* to provide a shortcut to views that you use often.

After adding a view to the current perspective, you may wish to save your new layout by clicking **Window > Save Perspective As**.

■ Related concepts

[Views](#)

[Fast views](#)

[Perspectives](#)

■ Related tasks

[Moving and docking views](#)

[Creating fast views](#)

[Maximizing a view or editor](#)

[Resetting perspectives](#)

© Copyright IBM Corporation and others 2000, 2003.

Moving and docking views

To change the location of a view in the current perspective:

1. Drag the view by its title bar. Do not release the left mouse button yet.
2. As you move the view around the Workbench, the mouse pointer changes to one of the *drop cursors* shown in the table below. The drop cursor indicates where the view will be docked if you release the left mouse button. To see the drop cursor change, drag the view over the left, right, top, or bottom border of another view or editor.
3. When the view is in the location that you want, relative to the view or editor area underneath the drop cursor, release the left mouse button.
4. (Optional) If you want to save your changes, select **Window > Save Perspective As** from the main menu bar.
5. Note that a group of stacked views can be dragged using the empty space to the right of the view tabs.

You can also move a view by using the pop-up menu for the view. (Left-click on the icon at the left end of the view's title bar, or right-click anywhere else in the view's title bar).

Drop cursor	Where the view will be moved to
	Dock above: The view is docked above the view underneath the cursor.
	Dock below: The view is docked below the view underneath the cursor.
	Dock to the right: The view is docked to the right of the view underneath the cursor.
	Dock to the left: The view is docked to the left of the view underneath the cursor.
	Stack: The view is docked as a Tab in the same pane as the view underneath the cursor.
	Restricted: You cannot dock the view in this area.

■ Related concepts

[Views](#)

[Fast views](#)

[Perspectives](#)

■ Related tasks

[Opening views](#)

[Creating fast views](#)

[Maximizing a view or editor](#)

[Saving a user defined perspective](#)

[Resetting perspectives](#)

© Copyright IBM Corporation and others 2000, 2003.

Rearranging tabbed views

In addition to dragging and dropping (docking) views inside the Workbench, you can rearrange the order of views within a tabbed notebook.

1. Click on the tab of the view that you want to move and drag it to where you want it. A stack symbol  appears as you drag the view across other view tabs.
2. Release the mouse button when you have the view tab in the desired location. The view that you selected is now moved.

■ Related concepts

[Views](#)

■ Related tasks

[Moving and docking views](#)

© Copyright IBM Corporation and others 2000, 2003.

Creating fast views

Fast views are hidden views that can be quickly opened and closed. They work like other views except they do not take up space in your Workbench window.

To create a fast view:

1. Click the title bar of the view that you want. Hold the mouse button down.
2. Drag the view to the shortcut bar at the far left of the window and release the mouse button.

The icon for the view that you dragged now appears on the shortcut bar. You can look at the view by clicking its icon on the shortcut bar. As soon as you click somewhere else outside the view, it is hidden again.

To restore the view to its original location (to delete the fast view), click on the Fast View icon  in the title bar of the view.

You can also create and restore fast views by selecting **Fast View** from the context menu that opens when you click the icon at the left side of the view's title bar.

■ Related concepts

[Views](#)

[Fast views](#)

[Perspectives](#)

■ Related tasks

[Working with fast views](#)

[Opening views](#)

[Moving and docking views](#)

[Maximizing a view or editor](#)

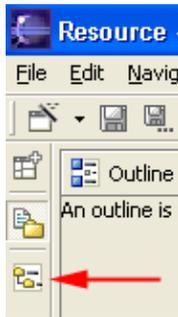
[Saving a user-defined perspective](#)

[Resetting perspectives](#)

© Copyright IBM Corporation and others 2000, 2003.

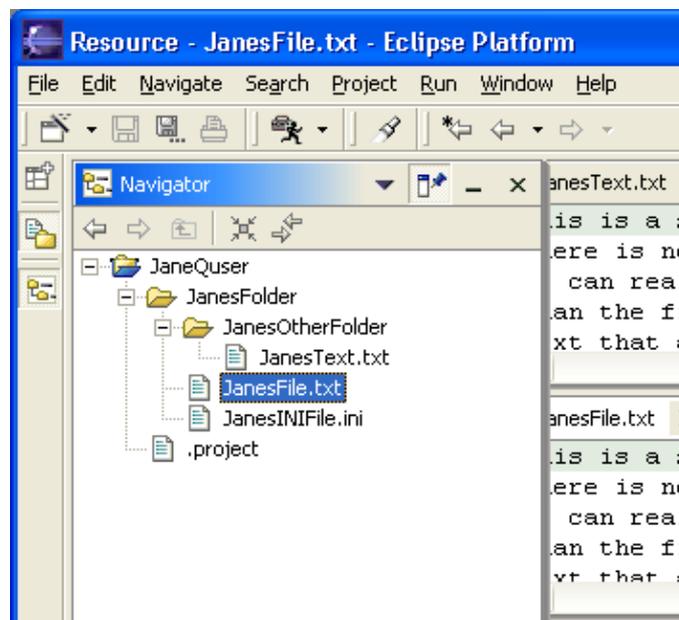
Working with fast views

If you have converted the Navigator to a fast view it will appear in the shortcut bar as shown below.

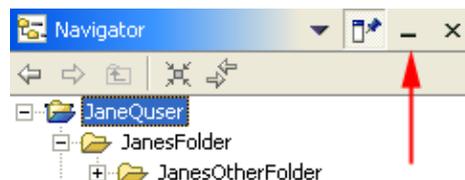


To work with a fast view proceed as follows.

1. In the shortcut bar click on the Navigator fast view button.
2. Observe the Navigator view slides out from the shortcut bar.



3. You can use the Navigator fast view as you would normally.
4. To hide the fast view simply click off of it or click on the Minimize button on the fast view's toolbar



Running Eclipse

Note: If you open a file from the Navigator fast view, the fast view will automatically hide itself to allow you to work with the file.

To convert a fast view back to a regular view you can either:

- Click the Restore button on the fast view's toolbar.



- Choose Restore from the context menu of the icon in the top left corner of the view.
- Drag the fast view's title bar (or close the fast view and then drag its button from the shortcut bar) and drop it on the workbench like a normal view.

■ Related concepts

[Views](#)

[Fast views](#)

[Perspectives](#)

■ Related tasks

[Creating fast views](#)

[Opening views](#)

[Docking views](#)

[Maximizing a view or editor](#)

[Saving a user defined perspective](#)

[Resetting perspectives](#)

© Copyright IBM Corporation and others 2000, 2003.

Opening files for editing

You can launch an editor for a given file in several ways.

- By right-clicking the file in the Navigator view and then selecting **Open** from the pop-up menu.
- By double-clicking the file in the Navigator view
- By double-clicking a bookmark that is associated with that file, in the Bookmarks view
- By double-clicking a problem, error, warning, or task record that is associated with that file, in the Tasks view

All of the above alternatives open the file in the default editor for that type of file. To open it in a different editor, select **Open With** from the file's pop-up menu.

■ Related concepts

[Editors](#)

[External editors](#)

■ Related tasks

[Associating editors with file types](#)

[Editing files outside the Workbench](#)

[Linking the Navigator view to the active editor](#)

[Tiling editors](#)

[Changing the placement of editor tabs](#)

[Comparing resources](#)

© Copyright IBM Corporation and others 2000, 2003.

Associating editors with file types

To associate editors with various file types in the Workbench:

1. From the main menu, select **Window > Preferences**.
2. In the Preferences window, expand the Workbench category and select **File Associations**.
3. Select the file type from the **File types** list, or click **Add** to add a type that is not already on the list.
4. In the **Associated editors** list, select the editor that you want to associate with that file type. To add an editor to the list:
 - a. Click **Add**. The Editor Selection dialog box opens.
 - b. Select **Internal Editors** or **External Programs**, depending on whether the editor that you want was built for the Workbench or runs outside the Workbench.
 - c. If you select **External Programs**, you can click the **Browse** button to browse the file system.
 - d. Select the editor from the list and click **OK**.
5. Click **OK** to finish associating the editor with the selected file type.

When you associate an internal editor with a file type, that editor opens in the editor area of the Workbench. For example, if you double-click a file in the Navigator or an entry in the Bookmarks or Tasks view it opens in the editor area.

WIN Editors that support OLE document mode can also run in the editor area of the Workbench.

Tip: You can choose to override your default editor selections by selecting **Open With** from the pop-up menu for any resource in the Navigator view.

■ Related concepts

[Editors](#)

[External editors](#)

■ Related tasks

[Opening files for editing](#)

[Editing files outside the Workbench](#)

[Linking the Navigator view to the active editor](#)

© Copyright IBM Corporation and others 2000, 2003.

Editing files outside the Workbench

To edit a Workbench resource outside the Workbench:

1. Navigate in the file system to the Workbench's installation directory. Go into the workspace directory and open the file that you want to edit with the external editor.
2. Edit the file as needed. Save and close it as usual.
3. *Important:* Go back to the Workbench, right-click the edited file in the Navigator view, and select **Refresh** from the pop-up menu. The Workbench will perform any necessary build or update operations to process the changes that you made outside the Workbench.

■ Related concepts

[Editors](#)

[External editors](#)

■ Related tasks

[Opening files for editing](#)

[Associating editors with file types](#)

© Copyright IBM Corporation and others 2000, 2003.

Tiling editors

The Workbench allows you to have multiple files open in multiple editors. Unlike views, editors cannot be dragged outside the Workbench to create new windows. However, you can tile editor sessions within the editor area, in order to view source files side by side.

1. With two or more files open in the editor area, select one of the editor tabs.
2. Holding down the left mouse button, drag that editor over the left, right, top or bottom border of the editor area. Notice that the mouse pointer changes to a "drop cursor" that indicates where the editor session will be moved when you release the mouse button.
3. (Optional) Drag the borders of the editor area or each editor, to resize as desired.

This is a similar operation to moving and docking views inside the Workbench, except that all editor sessions must be contained within the editor area.

■ Related concepts

[Editors](#)

■ Related tasks

[Opening files for editing](#)

[Changing the placement of the tabs](#)

[Comparing resources](#)

© Copyright IBM Corporation and others 2000, 2003.

Maximizing a view or editor

You can temporarily maximize a view or editor so that it fills the Workbench window.

1. Open the menu for the view or editor by clicking the icon at the far left of its title bar. (You can get the same effect by right-clicking anywhere in the title bar.)
2. Select **Maximize**.

To restore the view or editor to its previous position and size:

1. Open the menu again.
2. Select **Restore**.

Tip: You can also maximize or restore a view or editor by double-clicking its title bar.

■ Related concepts

[Views](#)

[Fast views](#)

[Perspectives](#)

■ Related tasks

[Opening views](#)

[Moving and docking views](#)

© Copyright IBM Corporation and others 2000, 2003.

Customizing the Workbench

Many aspects of the appearance and behavior of the Workbench can be customized to suit your individual needs. For example, you can:

- Rearrange where items appear in the main toolbar.
- Change the key bindings used by editors.
- Change the fonts and colors which are used.

See the Related tasks links for more details.

■ Related concepts

[Views](#)

[Editors](#)

[Workbench](#)

■ Related tasks

[Rearranging the main toolbar](#)

[Changing the key bindings](#)

[Controlling single and double click behavior](#)

[Changing fonts and colors](#)

[Changing the placement of the tabs](#)

© Copyright IBM Corporation and others 2000, 2003.

Rearranging the main toolbar

You can rearrange sections of the main toolbar. Toolbar sections are divided by a thin vertical line.

1. Make sure the toolbar is unlocked. The toolbar is unlocked if it has thick vertical bars next to the thin vertical toolbar dividers.



If it is locked, unlock the toolbar by selecting the **Window > Lock the Toolbars** menu item.

2. Grab the section of the toolbar you want to rearrange by moving the mouse over the thick vertical line on the left side of the desired segment. The mouse cursor changes its shape to indicate that you can click to move the toolbar section.
3. Click and hold the left mouse button to grab the toolbar section.
4. Move the section left and right or up and down. Release the mouse button to place it in the new location.
5. To prevent accidental changes to the toolbar lock it again by selecting the **Window > Lock the Toolbars** menu item.

■ Related concepts

[Toolbars](#)

© Copyright IBM Corporation and others 2000, 2003.

Changing the key bindings

The function of the keyboard can be extensively customized in Eclipse.

Select **Window > Preferences > Workbench > Keys** for the Keys preference page, where you can assign key sequences to many of the commands in Eclipse.

■ Related concepts

Keys

Navigating the user interface by using the keyboard

© Copyright IBM Corporation and others 2000, 2003.

Changing fonts and colors

By default, the Workbench uses the fonts and colors provided by the operating system. However, there are a number of ways that this behavior can be customized.

Fonts

The Workbench lets you directly configure three fonts:

Banner Font

Used in PDE editors, welcome pages and in the title area of many wizards. For instance the New Project wizard uses this font for the top title.

Header Font

Used as a section heading. For instance the Welcome page for the Eclipse Platform uses this font for the top title.

Text Font

Used in text editors.

To change these fonts:

1. On the main menu bar, click **Window > Preferences**.
2. Expand the **Workbench** category and select **Fonts**.
3. Select the font you want to change.
4. Click **Change**.
5. Use the dialog which opens to select a font.
6. Click **OK**.

Note: You can also click **Use System Font** to set the font to a reasonable value chosen by the operating system. For example, on Windows this will use the font selected in the Display Properties control panel.

Plug-ins that use other fonts may also provide preference entries to allow them to be customized. For example, the Java Development Tools provide a setting for controlling the font used by the Java editor (**Java > Editor > Appearance > Text font**).

In addition to the above, some text is always displayed in the system font. For example, the navigator tree always does this. To change the font used in these areas, you can use the configuration tools provided by the operating system (for example, the Display Properties control panel on Windows, or the .Xdefaults file in Motif).

Colors

To set the colors used by the Workbench to display error text and hyperlink text:

1. On the main menu bar, click **Window > Preferences**.
2. Expand the **Workbench** category and select **Appearance**.
3. Select the color you want to change.
4. Use the dialog which opens to select a color.
5. Click **OK**.

Running Eclipse

Plug-ins that use other colors may also provide preference entries to allow them to be customized. For example, the searching support provides a setting for controlling the color used to display inexact matches (**Workbench > Search > Foreground color for inexact matches**).

In general, the Workbench uses the colors that are chosen by the operating system. To change these colors you can use the configuration tools provided by the system (eg. the Display Properties control panel on Windows, or the .Xdefaults file in Motif).

■ Related concepts

[Fonts and Colors in Eclipse](#)

© Copyright IBM Corporation and others 2000, 2003.

Changing the placement of the tabs

You can change the placement of the tabs. The tabs for stacked views or editors can appear at the top or bottom of the area which contains them.

1. On the main menu bar, click **Window > Preferences**.
2. Expand the **Workbench** category and select **Appearance**.
3. Use the radio buttons in the **Editor tab positions** group or **View tab positions** group to control whether you want the tabs at the top or the bottom.
4. Click **Apply** and **OK**.
5. Close and reopen the Workbench to in order to see the changes.

■ Related concepts

[Views](#)

[Editors](#)

■ Related tasks

[Opening files for editing](#)

[Tiling editors](#)

© Copyright IBM Corporation and others 2000, 2003.

Controlling single and double click behavior

You can control how the Workbench responds to single and double clicks. To do this:

1. On the main menu bar, click **Window > Preferences**.
2. Select the **Workbench** category.
3. Select the behavior you want to change from the **Open mode** group.
4. Click **OK**.

The effect of these selections varies by view. For example, in the navigator:

Double click

Will cause a single click on a resource to select it, and a double click to open it in an editor.

Single click

Will cause a single click on a resource to both select it and immediately open an editor on it.

The check boxes under the **Single click** radio button further refine the single click behavior. Checking **Select on hover** will cause the resource to be selected if you hover over it with the mouse. Checking **Open when using arrow keys** will cause the resource to be opened if you use the arrow keys to navigate to it.

■ **Related reference**

[Workbench](#)

© Copyright IBM Corporation and others 2000, 2003.

Working with projects, folders and files

There are three different types of resources in the workbench: projects, folders, and files. Projects are the largest structural unit used by the Workbench. Projects contain folders and files, and they can be opened, closed, or built. Folders can contain other folders and files. The Workbench provides a number of mechanisms for working with projects, folders and files. See the related tasks section for more details.

Folders and files directly below projects can be linked to locations in the file system outside of the project's location. These special folders and files are called linked resources.

■ Related concepts

[Workbench](#)

[Resources](#)

[Resource hierarchies](#)

[Linked resources](#)

■ Related tasks

[Creating a project](#)

[Closing projects](#)

[Deleting projects](#)

[Creating a folder](#)

[Creating a file](#)

[Creating linked resources](#)

[Moving resources](#)

[Copying resources](#)

[Renaming resources](#)

[Deleting resources](#)

[Viewing resources properties](#)

© Copyright IBM Corporation and others 2000, 2003.

Creating a project

To create a project:

1. On the main menu bar, click **File > New Project**. The New Project wizard opens.
2. Select a category from the left column and then select the type of project to create from the right column. Click **Next**.
3. In the **Project name** field, type a name for your new project.
4. (Optional) The project that you create will map to a directory structure in the file system. The default file system location is displayed in the Location field. If you want to create the project and its contained resources in a different location, clear the **Use default location** check box and specify the new location.
5. If you want the new project to be dependent on one or more other projects, click **Next** and select the projects to be referenced.
6. Click **Finish**. The new project is listed in the Navigator view.

Tip: The **Window > Preferences > Perspectives > New project options** group allows you to specify the perspective behavior when a new project is created.

■ Related concepts

[Resources](#)

[Resource hierarchies](#)

■ Related tasks

[Creating a folder](#)

[Creating a file](#)

[Copying resources](#)

[Moving resources](#)

© Copyright IBM Corporation and others 2000, 2003.

Closing projects

When a project is closed, it can no longer be changed in the Workbench and its resources no longer appear in the Workbench, but they do still reside on the local file system. Closed projects require less memory. Also, since they are not examined during builds, closing a project can improve build time.

To close a project:

1. Select the project in the Navigator view.
2. Click **Close Project** on the pop-up menu.

To re-open the project:

1. Select the project in the Navigator view.
2. Click **Open Project** on the pop-up menu.

■ Related concepts

[Resources](#)

[Builds](#)

■ Related tasks

[Creating a project](#)

[Deleting projects](#)

© Copyright IBM Corporation and others 2000, 2003.

Deleting projects

To delete a project and remove its contents from the file system:

1. Select the project in the Navigator view.
2. Click **Delete** on the pop-up menu.
3. In the dialog which opens select **Also delete contents under**
4. Click **Yes**.

To delete a project from the workspace without removing its contents from the file system:

1. Select the project in the Navigator view.
2. Click **Delete** on the pop-up menu.
3. In the dialog which opens select **Do not delete contents**.
4. Click **Yes**.

■ Related concepts

[Resources](#)

■ Related tasks

[Creating a project](#)

[Closing projects](#)

© Copyright IBM Corporation and others 2000, 2003.

Creating a folder

To create a new folder:

1. In the Navigator view, right-click the project or folder where you want to create the new folder.
2. From the pop-up menu, select **New > Folder**.
3. Enter the name of the new folder and click **Finish**.

■ Related concepts

[Resources](#)

[Resource hierarchies](#)

■ Related tasks

[Creating a project](#)

[Creating a file](#)

[Creating linked resources](#)

[Copying resources](#)

[Moving resources](#)

© Copyright IBM Corporation and others 2000, 2003.

Creating a file

To create a file:

1. In the Navigator view, right-click the project or folder where you want to create the new file.
2. From the pop-up menu, select **New > File**.
3. Specify the name of the file, including the file extension (for example, newfile.txt).
4. Click **Finish**.

The file opens in the editor associated with its type.

■ Related concepts

[Resources](#)

[Resource hierarchies](#)

■ Related tasks

[Creating a project](#)

[Creating a folder](#)

[Creating linked resources](#)

[Copying resources](#)

[Moving resources](#)

© Copyright IBM Corporation and others 2000, 2003.

Creating linked resources

Folders and files that are direct children of projects can be linked to locations in the file system outside of the project's location. These special folders and files are called linked resources.

To create a linked folder:

1. In the Navigator view, right-click the project where you want to create the new folder.
2. From the pop-up menu, select **New > Folder**.
3. Specify the name of the folder as it will appear in the workbench. This name can be different from the name of the folder in the file system.
4. Click **Advanced**.
5. Check **Link to folder in the file system**.
6. Enter a file system path, or click **Browse** to select a folder in the file system.
7. Click **Finish**.

To create a linked file, follow the same steps as above, except choose **New > File** instead of **New > Folder** in the Navigator's context menu.

Linked resource locations can also be specified relative to a variable. This makes it easier to share projects containing linked resources with other team members, since it avoids hard-coded absolute file system paths that may vary from one machine to the next.

To define a linked resource relative to a path variable, do the following after step 5 above:

6. Click the **Variables** button.
7. In the resulting dialog, select an existing path variable or create a new one.
8. If the chosen variable defines the exact path of the linked resource, click **OK**. Otherwise, click **Extend** to specify a file or folder below the location described by the path variable, then click **OK**.
9. Click **Finish**.

Tip: The **Window > Preferences > Workbench > Linked Resources** preference page also allows you to define path variables.

Note that, once you create a linked resource you will not be able to change the link target path that you entered in step 6. or 8. above.

■ Related concepts

[Resources](#)

[Resource hierarchies](#)

[Linked resources](#)

■ Related tasks

[Creating a project](#)

[Creating a file](#)

[Creating a folder](#)

■ Related reference

[Linked resources](#)

New Folder wizard

New File wizard

© Copyright IBM Corporation and others 2000, 2003.

Moving resources

You can move resources from one Workbench location to another (for example, from one project to another project).

1. In the Navigator view, select the resources that you want to move.
2. From the pop-up menu, select **Move**.
3. In the Folder Selection window, select the project or folder where you want to move the resources to and click **OK**.

Tip: You can also move resources by dragging them from the original location to the new location in the Navigator view.

Moving a linked resource always moves the link and not the resource that it is linked to. E.g., when moving a linked folder the folder contents is not moved on the file system. Keep in mind that linked resources can only be moved to a project and not to a folder.

■ Related concepts

[Resources](#)

[Navigator view](#)

[Linked resources](#)

■ Related tasks

[Finding a resource quickly](#)

[Copying resources](#)

[Renaming resources](#)

[Deleting resources](#)

© Copyright IBM Corporation and others 2000, 2003.

Copying resources

You can copy resources from one Workbench location to another (for example, from one project to another project).

1. In the Navigator view, select the resources that you want to copy.
2. From the pop-up menu, select **Copy**.
3. In the Navigator view, select the project or folder where you want to copy the resources to.
4. From the pop-up menu, select **Paste**.

Tip: You can also copy resources by holding down the Ctrl key and dragging them from the original location to the new location in the Navigator view.

Copying a linked resource always copies the link and not the resource that it is linked to. E.g., when copying a linked folder the folder contents is not copied on the file system. Instead, a new linked folder is created linking to the same location on the file system. Keep in mind that linked resources can only be copied to a project and not to a folder.

■ Related concepts

[Resources](#)

[Navigator view](#)

[Linked resources](#)

■ Related tasks

[Creating a project](#)

[Creating a folder](#)

[Creating a file](#)

[Moving resources](#)

[Renaming resources](#)

[Deleting resources](#)

© Copyright IBM Corporation and others 2000, 2003.

Renaming resources

You can rename Workbench resources using the Rename command on the context menu in the Navigator view.

1. In the Navigator view, right-click the resource that you want to rename.
2. From the pop-up menu, select **Rename**.
3. Type the new name for the resource.
4. Hit the return key.

■ Related concepts

[Resources](#)

[Navigator view](#)

■ Related tasks

[Viewing resource properties](#)

[Moving resources](#)

© Copyright IBM Corporation and others 2000, 2003.

Deleting resources

To delete a resource from the Workbench:

1. In the Navigator view, select the resources that you want to delete. (Hold down the Ctrl key to select more than one resource.)
2. Press the Delete key.
3. Click **Yes** in the dialog which appears.

Tip: You can achieve the same result by selecting **Delete** from the pop-up menu.

Deleting resources from the workspace will also delete the corresponding files and/or folders from the local file system. The only exception is linked resources, which are not removed from the file system when you delete the link in the workspace. However, deleting child resources of linked folders *will* delete those resources from the local file system.

In most cases it is possible to restore deleted files from the local history. See the related tasks for more details.

■ Related concepts

[Resources](#)

[Navigator view](#)

[Linked resources](#)

■ Related tasks

[Deleting projects](#)

[Creating a project](#)

[Creating a folder](#)

[Creating a file](#)

[Creating linked resources](#)

[Restoring deleted resources from local history](#)

© Copyright IBM Corporation and others 2000, 2003.

Viewing resource properties

To display the various properties of a Workbench resource:

1. Right-click the resource in the Navigator view.
2. Select **Properties** from the pop-up menu.

The kinds of properties that are displayed depend on the specific resource you have selected and the features and plug-in that are installed in the Workbench.

Tip: When the resource is selected in the Navigator view you can also see basic properties for a resource by clicking **Window > Show View > Other > Basic > Properties**.

■ Related concepts

Resources

■ Related reference

Navigator view

■ Related tasks

Showing or hiding files in the Navigator view

© Copyright IBM Corporation and others 2000, 2003.

Navigating and finding resources

The Workbench provides a number of mechanisms for navigating and finding resources. See the Related tasks links for more details.

■ Related concepts

[Resources](#)

[Navigator view](#)

[Search view](#)

■ Related tasks

[Finding a resource quickly](#)

[Searching for files](#)

[Searching for text within a file](#)

[Sorting resources in the Navigator view](#)

[Showing or hiding files in the Navigator view](#)

[Narrowing the scope of the Navigator view](#)

[Linking the Navigator view to the active editor](#)

© Copyright IBM Corporation and others 2000, 2003.

Finding a resource quickly

To navigate to a particular resource in a view such as the navigator view:

1. From **Navigate** menu, select **Go To > Resource**.
2. In the window that opens, start typing the name of the resource in the **Pattern** field. As you type the file name, the system offers a list of possible matches, based on what you have entered so far.
3. Select the resource that you want from the **Matching Resources** list, and click **OK**.

The view displays the selected resource.

If you want to open a particular resource in an editor rather than select it in a view, you can use the **Navigate > Open Resource** action. This action presents the same dialog as the **Go To > Resource** action, but immediately opens the matching resource for editing.

Tip: If the **Open Resource** and **Go To > Resource** actions don't appear in your perspective, you can add them by selecting **Window > Customize Perspective > Other > Resource Navigation**. By default, these actions only appear in certain perspectives. The **Go To > Resource** action is only enabled when a view containing resources has focus.

You can also do a contextual search for character strings contained within files in the Workbench. See the links to related tasks below.

■ Related concepts

[Resources](#)

[Resource hierarchies](#)

[Navigator view](#)

■ Related tasks

[Searching for text within a file](#)

[Narrowing the scope of the Navigator view](#)

[Sorting resources in the Navigator view](#)

[Showing or hiding files in the Navigator view](#)

© Copyright IBM Corporation and others 2000, 2003.

Searching for files

The **Go To > Resource** action in the **Navigate** menu allows you to quickly find a resource in the Workbench by searching resource *names*.

You can also do more complex searches for files in the Workbench. For example, to find all files that end with .xml:

1. On the main toolbar, click the **Search** button .
2. Type .xml into the **File name patterns** field and leave the **Containing text** field empty. (You can use the pull-down list to select .xml if it had been previously entered.)
3. Finish entering your search options, for example to scope the search to specified working sets, and click **Search**.
4. The Search view displays the results of your search. Right-click on any item in the Search view to open a pop-up menu that allows you to remove items from the list, copy search results to the clipboard, or rerun the search. To open one of the listed files, double-click it or select **Go to File** from its pop-up menu.

If you close the Search view, you can return to it later by selecting **Window > Show View > Other > Basic > Search**.

■ Related concepts

[Search view](#)

[Working sets](#)

■ Related tasks

[Searching for text within a file](#)

■ Related reference

[Search view](#)

[File search](#)

© Copyright IBM Corporation and others 2000, 2003.

Searching for text within a file

The **Go To** selection in the pop-up menu for the Navigator view allows you to quickly find a resource in the Workbench by searching resource *names*.

You can also do contextual searches for information that is contained *inside* files in the Workbench. To find all files that contain a particular string of characters:

1. On the main toolbar, click the **Search** button .
2. Type your search string in the **Containing Text** field, or use the pull-down list to select a previously entered search expression.
Use \ as an escape character for search strings that contain the special characters *, ?, or \, (for example: d:\\directory\\filename.ext.
)
3. Finish entering your search options, (for example, to scope the search to specified file types), and click **Search**.
4. The Search view displays the results of your search. Right-click on any item in the Search view to open a pop-up menu that allows you to remove items from the list, copy search results to the clipboard, or rerun the search. To open one of the listed files, double-click it or select **Go to File** from its pop-up menu.

If you close the Search view, you can return to it later by selecting **Window > Show View > Other > Basic > Search**.

■ Related concepts

[Search view](#)

■ Related tasks

[Searching for files](#)

■ Related reference

[Search view](#)

[File search](#)

© Copyright IBM Corporation and others 2000, 2003.

Sorting resources in the Navigator view

To sort Workbench resources in the Navigator view by name or by file type:

1. On the toolbar for the Navigator view, click the **Menu** button ▼ to open the drop-down menu of display options.
2. Select **Sort**.
3. Select the desired sort option.

■ Related concepts

[Resources](#)

[Resource hierarchies](#)

[Navigator view](#)

■ Related tasks

[Finding a resource quickly](#)

[Narrowing the scope of the Navigator view](#)

[Showing or hiding files in the Navigator view](#)

© Copyright IBM Corporation and others 2000, 2003.

Showing or hiding files in the Navigator view

You can choose to hide system files or generated class files in the Navigator view. (System files are those that have only a file extension but no file name, for example .classpath.)

1. On the toolbar for the Navigator view, click the **Menu** button ▼ to open the drop-down menu of display options.
2. Select **Filter**.
3. In the dialog box that opens, select the checkboxes for the types of files that you want to hide.

In addition, you can restrict the displayed files to a working set.

1. On the toolbar for the Navigator view, click the **Menu** button ▼ to open the drop-down menu of display options.
2. Choose **Select Working Set...**
3. Select an existing working set from the list or create a new one by selecting **New...**

■ Related concepts

[Resources](#)

[Resource hierarchies](#)

[Navigator view](#)

[Working sets](#)

■ Related tasks

[Viewing resource properties](#)

[Finding a resource quickly](#)

[Narrowing the scope of the Navigator view](#)

[Sorting resources in the Navigator view](#)

© Copyright IBM Corporation and others 2000, 2003.

Narrowing the scope of the Navigator view

By default, the Navigator view shows all resources in your Workbench. You can focus on a subset of resources by temporarily "going into" a project or folder and hiding all other resources.

1. In the Navigator view, right-click the project or folder that you want to focus on.
2. From the pop-up menu, select **Go Into**.

The Navigator now shows only the contents of the selected project or folder. The title of the Navigator shows the name of the resource you are currently looking at. You can use the **Back**, **Forward**, and **Up** buttons on the Navigator view's toolbar to change the scope.

■ Related concepts

[Navigator view](#)

■ Related tasks

[Showing or hiding files in the Navigator view](#)

© Copyright IBM Corporation and others 2000, 2003.

Linking the Navigator view to the active editor

When you have multiple files open for editing, you can configure the Navigator view to automatically bring an open file to the foreground (make its editor session the active editor) every time you select that open file in the Navigator view. There are two ways to set this

1. From the Navigator menu, select **Link With Editor**.
2. Click on the Link With Editor icon  on the Navigator bar.

■ Related concepts

[Editors](#)

■ Related tasks

[Opening files for editing](#)

[Associating editors with file types](#)

© Copyright IBM Corporation and others 2000, 2003.

Bookmarks, tasks and other markers

Markers are objects that may be associated with Workbench resources. There are many uses of markers in the Workbench, including providing support for bookmarking resources or locations within resources, tracking ongoing tasks, or displaying error messages. See the related tasks section for more details.

■ Related concepts

[Bookmarks](#)

[Tasks view](#)

[Markers](#)

■ Related tasks

[Creating a bookmark within a file](#)

[Creating a bookmark for an entire file](#)

[Deleting a bookmark](#)

[Adding line items in the Tasks view](#)

[Associating a task with a resource](#)

[Deleting tasks](#)

[Filtering the Task view](#)

[Automatically fixing problems](#)

© Copyright IBM Corporation and others 2000, 2003.

Creating a bookmark within a file

The Workbench allows you to create bookmarks in files that you edit so that you can quickly reopen those files from the Bookmarks view.

1. With the file open in an editor, right-click in the gray border at the left of the editor area, next to the line of code or text that you want to bookmark.
2. Select **Add Bookmark** from the pop-up menu.
3. Notice that an icon for the bookmark now appears in the left border of the editor area. A line is also added to the Bookmarks view (**Window > Show View > Bookmarks.**)

You can reopen the file for editing at any time by double-clicking the bookmark in the Bookmarks view.

■ Related concepts

[Bookmarks](#)

[Tasks view](#)

■ Related tasks

[Creating a bookmark for an entire file](#)

[Deleting a Bookmark](#)

[Adding line items in the Tasks view](#)

[Associating a task with a resource](#)

© Copyright IBM Corporation and others 2000, 2003.

Creating a bookmark for an entire file

You can bookmark individual files in the Workbench, in order to open them quickly from the Bookmarks view later.

1. In the Navigator view, right-click the file that you want to add to your list of bookmarks.
2. Select **Add Bookmark** from the file's pop-up menu.

At any time, You can open the bookmarked file for editing by double-clicking the bookmark in the Bookmarks view (**Window > Show View > Bookmarks**).

You can also create bookmarks for specific lines of text or source code or within a file. See the list of related topics below.

■ Related concepts

[Bookmarks](#)

[Tasks view](#)

■ Related tasks

[Creating a bookmark within a file](#)

[Deleting a Bookmark](#)

[Adding line items in the Tasks view](#)

[Associating a task with a resource](#)

© Copyright IBM Corporation and others 2000, 2003.

Deleting a bookmark

To delete any bookmark:

1. Open the Bookmarks view.
2. Right-click the bookmark that you want to delete and select **Delete** from the pop-up menu.

If you have added bookmarks for specific lines within a source file, you can also delete them while editing the file by right-clicking the bookmark icon in the editor area.

■ Related concepts

[Bookmarks](#)

■ Related tasks

[Creating a bookmark for an entire file](#)

[Creating a bookmark within a file](#)

© Copyright IBM Corporation and others 2000, 2003.

Adding line items in the Tasks view

The Tasks view contains line items for system-generated problems, warnings, and errors. You can add your own entries to the table to build a list of to-do items, or tasks.

1. On the toolbar in the Tasks view, click the **New Task** button . The **New Task** dialog will open.
2. Type a brief description for the task in the **Description** field. The new task is assigned default priority and completed values. These values may also be modified within the **New Task** dialog.
3. Press **OK**.

■ Related concepts

[Tasks view](#)

[Bookmarks](#)

■ Related tasks

[Deleting tasks](#)

[Associating a task with a resource](#)

[Filtering the Tasks view](#)

[Creating a bookmark for an entire file](#)

[Creating a bookmark within a file](#)

© Copyright IBM Corporation and others 2000, 2003.

Associating a task with a resource

You can associate tasks with an editable resource, for instance to remind yourself to update a line of source code later.

1. In the Navigator view, double-click the resource with which you wish to associate the new task. The resource opens in the editor area.
2. Right-click in the gray border at the left of the editor area, beside the line of text or source code against which you want to log the new task.
3. On the pop-up menu, select **Add Task**.
4. When prompted, enter a brief description of the task.

A new task icon appears in the border of the editor area, to the left of the line where you added the task. When you move the mouse pointer over the marker, the description of the task is displayed as a tooltip. The task is also added to the Tasks view. You can delete a task either by right-clicking its icon in the editor area and selecting **Remove Task**, or by pressing the Delete key in the Tasks view.

■ Related concepts

[Tasks view](#)

[Bookmarks](#)

■ Related tasks

[Adding line items in the Tasks view](#)

[Deleting tasks](#)

[Filtering the Tasks view](#)

[Creating a bookmark for an entire file](#)

[Creating a bookmark within a file](#)

© Copyright IBM Corporation and others 2000, 2003.

Deleting tasks

You can delete associated tasks from the gray border at the left of the editor area.

1. In the marker bar in the editor area, locate the task marker that you want to delete.
2. From the marker's pop-up menu, select **Remove Task**.
3. The task marker disappears and the task is removed from the Tasks view.

You can also delete one or more line items from the Tasks view by pressing the Delete key or by clicking the **Delete** button on the Tasks view toolbar, or from the pop-up menu.

Tip: To delete all completed tasks, right-click in the Tasks view and select **Delete Completed Tasks** from the pop-up menu.

■ Related concepts

[Tasks view](#)

© Copyright IBM Corporation and others 2000, 2003.

Filtering the Tasks view

You can filter the tasks that are displayed in the Tasks view. For example, you might wish to see only problems that have been logged by the Workbench, or tasks that you have logged as reminders to yourself. You can filter items according to which resource or group of resources they are associated with, by text string within the Description field, by problem severity, by task priority, or by task status.

1. On the toolbar of the Tasks view, click the **Filter** button .
2. Select the radio buttons and checkboxes that correspond to your filtering objectives.
3. Press **OK**.

■ Related concepts

[Tasks view](#)

■ Related tasks

[Adding line items in the Tasks view](#)

[Associating a task with a resource](#)

[Deleting tasks](#)

© Copyright IBM Corporation and others 2000, 2003.

Automatically fixing problems

Problems displayed in the task view may offer a **Quick Fix** menu option in the context menu. Selecting this menu option will present one or more possible fixes that can be automatically applied for you.

Certain Java and team problems may be resolved using the Quick Fix feature. For example a missing import statement in a Java file will result in a Quick Fix suggestion to add the import statement or change the type name to a name that is already imported. The CVS plug-in will offer a quick fix item to add a newly created file to version control.

1. In the Tasks view, right-click the task you want to Quick Fix.
2. Select **Quick Fix** from the pop-up menu.
3. Choose from one of the suggested fixes.

■ Related concepts

[Markers](#)

[Tasks view](#)

■ Related reference

[Tasks view](#)

© Copyright IBM Corporation and others 2000, 2003.

Comparing resources

When a comparison is performed, comparison editors appear in the editor area. The differences between files are highlighted in the comparison editors, allowing you to browse and copy changes between the compared resources.

To compare resources:

1. Select one or more resources in the Navigator view.
2. From the resource's pop-up menu, select **Compare With**.

A tool for viewing differences is opened in the editor area.

Here are some of the comparisons that you can perform.

Compare With > Latest from Repository

compares the selected resource with the version of the same resource that is currently committed to the active branch.

Compare With > Branch or Version

compares the selected resource with a version that has been committed to the repository or to the latest version in a particular branch. This requires you to choose a version of the resource or a branch from a list.

Compare With > Patch

Displays the result of applying a patch to a resource.

Compare With > Each Other

compares two or three selected resources with each other

Compare With > Revision

compares the selected resource with a version of the same resource that was committed to the active branch. This requires you to choose a revision of the resource from a list.

Compare With > Local History

compares the selected resource to one that is in the local history, which is maintained when you save changes. (*Tip:* to configure the size and depth of your local history, look under **Window > Preferences > Workbench > Local History**.)

Tip: You can customize the behavior of the comparison editor by selecting **Window > Preferences > Workbench > Compare/Patch** from the main menu bar. You can select to "lock scroll" the pane contents, as well as setting options for showing conflicts with other team members and changing the default font.

■ Related concepts

[Synchronizing with a CVS repository](#)

[Three way comparisons](#)

[Local history](#)

■ Related tasks

[Synchronizing with the repository](#)

[Merging changes in the Compare editor](#)

[Resolving conflicts](#)

[Setting preferences for comparing files](#)

[Comparing resources with repository versions](#)

[Understanding the comparison](#)

[Working with patches](#)

[Tiling editors](#)

■ **Related reference**

[Compare editor](#)

[CVS Synchronize view](#)

© Copyright IBM Corporation and others 2000, 2003.

Setting preferences for comparing files

When you select to compare or synchronize two or more resources in the Workbench, one or more comparison editors usually open. To customize how these editors behave:

1. On the main menu bar, select **Window > Preferences**. The Preferences window opens
2. Expand the **Workbench** category on the left and select **Compare**.
3. Set your preferences and click **OK**.

You can configure the following options on the **General** tab.

Option	Description	Default
Open structure compare automatically	Makes an additional information area visible which shows differences in the underlying structure of the resources being compared. This information may not be available for all comparisons.	On
Show additional compare information in the status line	Causes the status line to display additional context information about the comparison.	Off
Ignore white space	Causes the comparison to ignore differences which are whitespace characters (spaces, tabs, etc.). Also causes differences in line terminators (LF versus CRLF) to be ignored.	Off
Automatically save dirty options before patching	This option controls whether any unsaved changes are automatically saved before a patch is applied.	Off

You can configure the following options in the **Text Compare** tab.

Option	Description	Default
Synchronize scrolling between panes in compare viewers	The two compare viewers will "lock scroll" along with one another in order to keep identical and corresponding portions of the code in each pane side by side.	On
Initially show ancestor pane	Sometimes you may want to compare two versions of a resource with the previous version from which they were both derived. This is called their <i>common ancestor</i> , and it appears in its own comparison pane during a three way comparison.	Off
Show pseudo conflicts	Displays conflicts that occur when two developers make the same change, for example when both add or remove the same line of code.	Off

Running Eclipse

Connect ranges with single line	Controls whether differing ranges are visually connected by a single line or a range delimited by two lines.	On
---------------------------------	--	----

■ Related concepts

Three way comparisons

■ Related tasks

Comparing resources

Synchronizing with the repository

Merging changes in the Compare editor

■ Related reference

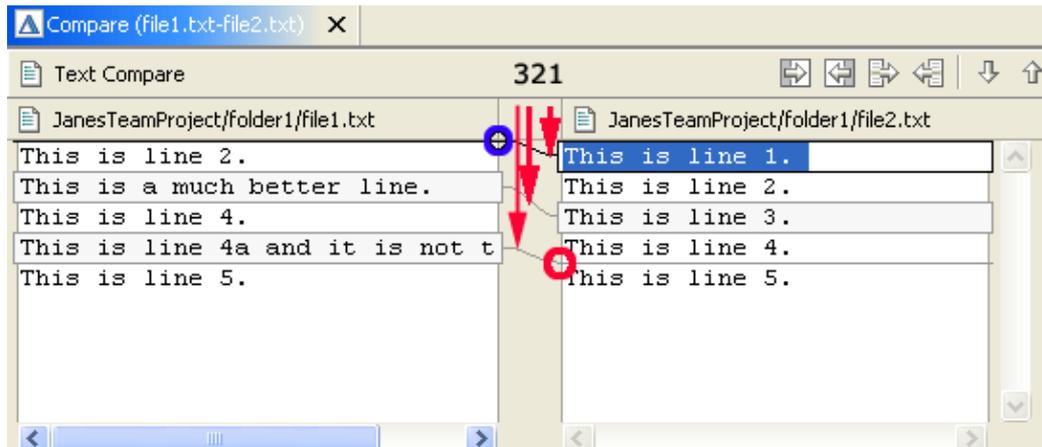
Compare editor

© Copyright IBM Corporation and others 2000, 2003.

Understanding the comparison

If you compare two files (file1.txt and file2.txt), the following results are shown in the compare editor. The left side shows the contents of file1.txt and the right side shows the contents of file2.txt. The lines connecting the left and right panes indicate the differences between the files.

You can double click on the editor tab to maximize the editor.



Looking at the numbered changes in the above image:

- Starting with the top line (in the the left pane) we can see that the difference bar (in the area of the blue circle) indicates something is missing from the very top of the left file. If we follow the difference band (see #1) to the right file we can see that it contains "This is line 1" .
- The next line "This is line 2." is white indicating it matches the right file.
- Moving onto the next line (colored in gray) we can see that the left file and right file have different contents for this line (see #2).
- The next line ("This is line 4") is once again in white, so we can skip it, since the contents are the same in both.
- The next line exists in the left file but since it is gray we follow its difference bar to the right (see #3) and notice that the right file does not contain the line (see red circle).

Hint: On the right-hand side of the comparison, to the right of the scrollbar, there is a column which shows a graphical representation of all differences between the resources. You can click on any of the segments displayed there to quickly scroll to that difference.

Related concepts

[Synchronizing with a CVS repository](#)

[Three way comparisons](#)

[Local history](#)

Related tasks

[Comparing resources](#)

[Synchronizing with the repository](#)

[Merging changes in the Compare editor](#)

[Resolving conflicts](#)

[Setting preferences for comparing files](#)

[Comparing resources with repository versions](#)

[Tiling editors](#)

■ **Related reference**

[Compare editor](#)

[Synchronize view](#)

© Copyright IBM Corporation and others 2000, 2003.

Merging changes in the compare editor

The toolbar buttons in the compare editor allows you to merge changes from the left file to the right file and vice versa. There are four types of merges you can perform:

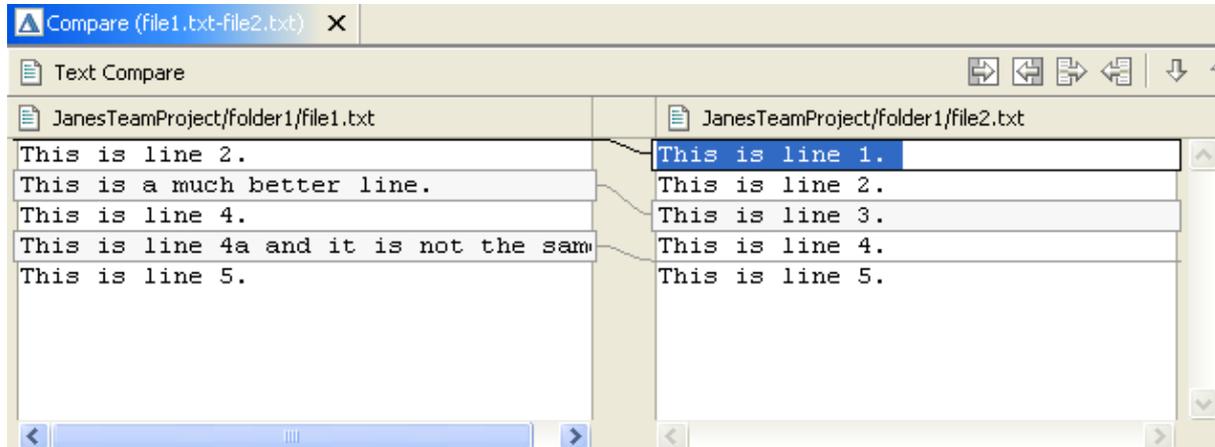
- Copy All from Left to Right
- Copy All from Right to Left
- Copy Current Change from Left to Right
- Copy Current Change from Right to Left

The **Copy All from Left to Right** and **Copy All from Right to Left** actions completely replace the contents of one resource with the other resource.

To merge a single change:

1. Select the highlighted difference that you want to merge.
2. Depending on what you want to do, click either the **Copy Current Change from Right to Left** or the **Copy Current Change from Left to Right** toolbar button. The selected text is copied from one file to the other.
3. Right click to get the resource's pop-up menu, and select **Save**.

The image below gives an example of two files (file1.txt and file2.txt) compared.



● Related concepts

[Three way comparisons](#)

[Synchronizing with a CVS repository](#)

● Related tasks

[Comparing resources](#)

[Understanding the comparison](#)

[Synchronizing with the repository](#)

[Resolving conflicts](#)

[Setting preferences for comparing files](#)

■ Related reference

[Compare editor](#)

[Synchronize view](#)

© Copyright IBM Corporation and others 2000, 2003.

Working with local history

A local edit history of a file is maintained when you create or modify a file. Each time you edit and save the file, a copy is saved so that you can replace the current file with a previous edit or even restore a deleted file. You can also compare the contents of all the local edits. Each edit in the local history is uniquely represented by the date and time the file was saved. See the Related tasks links for more details.

■ Related concepts

[Local history](#)

■ Related tasks

[Comparing resources with the local history](#)

[Replacing a resource with local history](#)

[Restoring deleted resources from local history](#)

[Setting local history preferences](#)

© Copyright IBM Corporation and others 2000, 2003.

Comparing resources with the local history

To compare a Workbench resource with a state in the local history:

1. In the Navigator view, select the resource that you want to compare with a local history state.
2. From the resource's pop-up menu, select **Compare With > Local History**. The Compare with Local History page opens.
3. Select a state in the **Local History** list. The Text Compare editor opens.
4. Click the **Select Next Change** and **Select Previous Change** buttons to browse the changes made between the state in the local history and the Workbench resource.
5. Click **OK** when you are finished.

■ Related concepts

[Local history](#)

[Versions](#)

■ Related tasks

[Replacing a resource with local history](#)

[Restoring deleted resources from local history](#)

[Setting local history preferences](#)

[Comparing resources](#)

© Copyright IBM Corporation and others 2000, 2003.

Replacing a resource with local history

To replace a Workbench resource with a state in the local history:

1. In the Navigator view, select the resource that you want to replace with a local history state.
2. From the resources pop-up menu, select **Replace with > Local History**. The Replace from Local History page opens.
3. Select a state from the **Local History** list. The Text Compare editor opens.
4. Click the **Select Next Change** and **Select Previous Change** buttons to browse through the changes made between states in the local history and the Workbench resource.
5. Select the state you want to replace, and click **Replace**.

Tip: You can configure your Workbench preferences to specify how many days to keep files, or how many entries per file you want to keep, or the maximum file size for files to be kept. Select **Window > Preferences > Workbench > Local History**.

■ Related concepts

[Local history](#)

[Resources](#)

■ Related tasks

[Comparing resources with the local history](#)

[Restoring deleted resources from the local history](#)

[Setting local history preferences](#)

© Copyright IBM Corporation and others 2000, 2003.

Restoring deleted resources from local history

To restore a deleted Workbench resource with a state from the local history:

1. In the Navigator view, select the folder or project into which you want to restore a local history state.
2. From the resource's pop-up menu, select **Restore from Local History...** The Restore From Local History dialog opens showing all files that were previously contained in the selected folder or project and all of their sub-folders.
3. Check the files that you want to restore
4. If you don't want to restore just the last state of a file you can select any other state of the file from the **Local History** list on the right hand side of the dialog. The bottom pane of the dialog shows the contents of the state.
5. If you are done with all files click **Restore**.

Tip: You can configure your Workbench preferences to specify how many days to keep files, or how many entries per file you want to keep, or the maximum file size for files to be kept. Select **Window > Preferences > Workbench > Local History**.

■ Related concepts

[Local history Resources](#)

■ Related tasks

[Comparing resources with the local history](#)
[Replacing resources with the local history](#)
[Setting local history preferences](#)

© Copyright IBM Corporation and others 2000, 2003.

Setting local history preferences

To indicate the level of local history that should be kept for each resource in the Workbench:

1. Click **Window > Preferences**.
2. Select the **Workbench > Local History** category in the left pane. The Local History Preferences page opens.
3. In the **Days to keep files** field, type the number of days that you want to keep records for any one Workbench resource. For example, if you type 7, then a history of saved states from the last seven days will be kept.
4. In the **Entries per file** field, type the number of states to keep for any one Workbench resource. Note that when you exceed the number of entries per file, the oldest changes are discarded to make room for the newer changes.
5. In the **Maximum file size (MB)** field, type the maximum file size (in MB) of a resource for which a local history should be kept. If the size of the resource exceeds the maximum amount of file size allocated, no local history is kept for that resource.
6. Click **OK** to set your preferences and close the Local History Preferences page.

■ Related concepts

[Local history](#)

[Versions](#)

[Resources](#)

■ Related tasks

[Comparing resources with the local history](#)

[Replacing a resource with local history](#)

[Restoring deleted resources from local history](#)

© Copyright IBM Corporation and others 2000, 2003.

Importing

You can import files into the Workbench three ways:

- By using the import wizard
-  By dragging files or folders from the file system to the Navigator view
-  By copying files or folders from the file system and pasting them into the Navigator view

See the related tasks section for more details.

■ Related concepts

[Resources](#)

■ Related tasks

[Importing resources from the file system](#)

[Importing resources from a ZIP File](#)

[Exporting](#)

© Copyright IBM Corporation and others 2000, 2003.

Importing existing projects

You can use the Import Wizard to copy a project that exists in a different workspace, or one that previously existed in a workspace, into the Workbench.

1. From the main menu bar, select **File > Import**. The Import wizard opens.
2. Select **Existing Project into Workspace** and click **Next**.
3. Click the **Browse** button on the next page of the wizard to select the project which you would like to import.
4. Click **Finish** to start the import.

■ Related reference

[Import wizard](#)

■ Related tasks

[Importing resources from the file system](#)

[Importing resources from a ZIP File](#)

[Exporting](#)

© Copyright IBM Corporation and others 2000, 2003.

Importing resources from the file system

You can use the Import Wizard to copy files from a file system directory into the Workbench.

1. From the main menu bar, select **File > Import**. The Import wizard opens.
2. Select **File System** and click **Next**.
3. Click the **Browse** button on the next page of the wizard to select the directories from which you would like to add the resources.
4. In the import selection panes, use the following methods to select exactly the resources you want to add:
 - ◆ Expand the hierarchies in the left pane and select or clear the checkboxes that represent the folders in the selected directory. Then in the right pane, select or clear checkboxes for individual files.
 - ◆ Click **Filter Types** to filter the current selection for files of a specific type.
 - ◆ Click **Select All** to select all resources in the directory, then go through and deselect the ones that you do not want to add.
 - ◆ Click **Deselect All** to deselect all resources in the directory, then go through and choose individual resources to add.
5. Specify the Workbench project or folder that will be the import destination.
6. When you have finished specifying your import options, click **Finish**.

Tip: You can also import folders and files by dragging them from the file system and dropping them in the Navigator view, or by copying and pasting.

■ Related concepts

[Resources](#)

■ Related reference

[Import wizard](#)

■ Related tasks

[Importing existing projects](#)

[Importing resources from a ZIP File](#)

[Exporting](#)

© Copyright IBM Corporation and others 2000, 2003.

Importing resources from a ZIP file

You can use the Import wizard to extract files from a ZIP file into the Workbench.

1. From the main menu bar, select **File > Import**. The Import wizard opens.
2. Select **ZIP file** and click **Next**.
3. Click the **Browse** button on the next page of the wizard, to select the ZIP files that contain the files you want to extract and import into the Workbench.
4. In the import selection panes, use the following methods to select exactly the resources you want to add:
 - ◆ Expand the hierarchies in the left pane and select or clear the checkboxes that represent the folders in the selected directory. Then in the right pane, select or clear checkboxes for individual files.
 - ◆ Click **Filter Types** to filter the current selection for files of a specific type.
 - ◆ Click **Select All** to select all resources in the directory, then go through and deselect the ones that you do not want to add.
 - ◆ Click **Deselect All** to deselect all resources in the directory, then go through and choose individual resources to add.
5. Specify the Workbench project or folder that will be the import destination.
6. When you have finished specifying your import options, click **Finish**.

■ Related concepts

[Resources](#)

■ Related reference

[Import wizard](#)

■ Related tasks

[Importing existing projects](#)

[Importing resources from the file system](#)

[Exporting](#)

© Copyright IBM Corporation and others 2000, 2003.

Exporting

You can export files from the Workbench in three ways:

- By using the Export wizard.
-  By dragging files or folders from the Navigator view to the file system.
-  By copying files or folders from Navigator view and pasting them into the file system.

See the related tasks section for more details.

■ Related concepts

[Resources](#)

■ Related tasks

[Exporting resources to the file system](#)

[Exporting resources to a ZIP File](#)

[Importing](#)

© Copyright IBM Corporation and others 2000, 2003.

Exporting resources to the file system

You can use the Export wizard to export resources from the Workbench to the file system.

1. In the Navigator view, select the resources that you want to export.
2. From the main menu bar, select **File > Export**. The Export wizard opens.
3. Select **File System** and click **Next**.
4. By default, the resources that you selected will be exported, along with all their children. Optionally, use the checkboxes in the left and right panes to select the set of resources to export, and use push buttons such as **Select Types** to filter the types of files that you want to export.
5. Click the **Browse** button on the next page of the wizard, to select the directory you would like to export the resources to.
6. Specify the directory in the file system that will be the export destination.
7. Click **Finish**.

Tip: You can also export folders and files by dragging them from the Navigator view to the file system and dropping them in the file system, or by copy and paste.

■ Related concepts

[Resources](#)

■ Related tasks

[Importing](#)

[Exporting resources to a ZIP File](#)

© Copyright IBM Corporation and others 2000, 2003.

Exporting resources to a ZIP file

You can use the Export wizard to export resources from the Workbench to a ZIP file in the file system.

1. In the Navigator view, select the resources that you want to export.
2. From the main menu bar, select **File > Export**. The Export wizard opens.
3. Select **ZIP file** and click **Next**.
4. By default, the resources that you selected will be exported along with their children. Optionally, use the checkboxes in the left and right panes to select the set of resources to export, and use push buttons such as **Select Types** to filter the types of files that you want to export.
5. Specify the path and name of the ZIP file into which you want to export the selected resources.
6. Click **Finish**.

■ Related concepts

[Resources](#)

■ Related tasks

[Exporting resources to the file system](#)

[Importing](#)

© Copyright IBM Corporation and others 2000, 2003.

Building resources

There are a number of ways that resources in the Workbench can be built. The scope of a build can be one or more selected projects or over the entire workspace.

Builds can be full,(meaning that all resources within the scope of the build are considered); or incremental (meaning that only resources that have changed since the last build are considered).

Builds can be done automatically (each time resources are modified), or manually, using a menu item. See the Related tasks links for more details.

■ Related concepts

[Builds](#)

■ Related tasks

[Performing builds manually](#)

[Performing builds automatically](#)

[Saving resources automatically before a manual build](#)

© Copyright IBM Corporation and others 2000, 2003.

Performing builds automatically

To indicate that you want the Workbench to perform incremental builds whenever resources are saved:

1. Click **Window > Preferences**.
2. Select the **Workbench** category in the left pane. The Workbench Preferences page opens.
3. Select the **Perform build automatically on resource modification** check box.
4. Click **OK** to close the Preferences page. The Workbench will automatically perform incremental builds of resources modified since the last build. Whenever a resource is modified, another incremental build occurs.

■ Related concepts

[Builds](#)

■ Related tasks

[Performing builds manually](#)

[Changing build order](#)

© Copyright IBM Corporation and others 2000, 2003.

Performing builds manually

By default, builds are performed automatically when you save resources. If you need more control over when builds occur, you can disable automatic building and manually invoke builds. This is sometimes desirable in cases where you know building should wait until you finish a large set of changes. The disadvantage of manual building is that tasks generated to indicate build errors quickly become out of date until you build. In addition, it is very important that you remember to manually build before relying on build output (for example, before running your Java program).

Note: Some of the menu items described below are only available when the automatic build preference is disabled in the Workbench preferences (select **Window > Preferences > Workbench**).

To build only selected projects:

1. In the Navigator view of the Resource perspective, select one or more projects.
2. Right-click and select the option **Rebuild Project** from the pop-up menu.

To build all projects in the workspace click **Project > Rebuild All**. Note that, this will rebuild all resources, even the ones which have not been modified.

■ Related concepts

[Builds](#)

■ Related tasks

[Saving resources automatically before a manual build](#)

[Changing build order](#)

[Performing builds automatically](#)

© Copyright IBM Corporation and others 2000, 2003.

Saving resources automatically before a manual build

To automatically save all modified resources in the Workbench before a manual build is done:

1. Click **Window > Preferences**.
2. Select the **Workbench** category in the left pane to open the Workbench Preferences page.
3. Select the **Save all modified resources automatically prior to manual build** check box.
4. Click **OK** to close the Preferences page.

■ Related concepts

[Builds](#)

■ Related tasks

[Building resources](#)

[Performing builds manually](#)

[Changing build order](#)

© Copyright IBM Corporation and others 2000, 2003.

Changing build order

By default, the Workbench computes the build order by interpreting project references as prerequisite relationships. Alternatively, you can explicitly define the order in which projects are built.

The Build Order preference page allows you to disable the **Use default build order** option so that you can access the projects list and manipulate the order of it.

To define the order in which the Workbench performs builds projects:

1. Click **Window > Preferences** to open the Preferences dialog.
2. Select the **Build Order** category in the left pane. The Build Order preference page opens, listing the projects for the build.
3. Ensure that the **Use default build order** check box is cleared. If it is selected, the buttons on this page are disabled, and builds are performed in the order of the project list.
4. (Optional) Click **Add Project** and **Remove Project** buttons to add and remove projects from the list.
5. Select one or more projects in the list and click **Up** or **Down** to set the preferred project build order.
6. Click **OK** to close the Preferences dialog.

■ Related concepts

[Builds](#)

■ Related tasks

[Performing builds manually](#)

[Performing builds automatically](#)

© Copyright IBM Corporation and others 2000, 2003.

Running external tools

The Workbench provides a mechanism for running tools that are not part of it. To configure an external tool:

1. Click **Run > External Tools > Configure**.
2. Click the **New** button.
3. Enter a name for your external tool (for example, My External Tool).
4. Click the **Browse File System** button.
5. Find the tool you want to run (for example, on Windows it is usually a file with the extension .exe or .bat).
6. (Optional) In the **Tool Arguments** field enter the necessary arguments for the tool.
7. (Optional) In the **Working directory** field enter the working directory for the tool.
8. Click **OK**.
9. To run the tool, click **Run > External Tools > My External Tool**.

It is also possible to set up and run an external tool to build a project.

1. Select the desired project.
2. From its pop-up menu choose **Properties**.
3. Click **External Tools Builders** from the list, and configure the tool as described above.

■ Related concepts

[External tools](#)

■ Related tasks

[Running Ant buildfiles](#)

© Copyright IBM Corporation and others 2000, 2003.

Running Ant buildfiles

To run an Ant buildfile in the Workbench:

1. In the Navigator view, select an XML file.
2. From the file's pop-up menu, select **Run Ant...** The launch configuration dialog opens.
3. Select one or more targets from the **Targets** tab. The order in which you select the items is the order in which they will run. The order is displayed in the **Target execution order** box at the bottom of the tab. You can change the order of the targets by clicking the **Order...** button.
4. (Optional) Configure options on the other tabs. For example, on the **Main** tab, type any required arguments in the **Arguments** field.
5. Click **Run**.

The Ant buildfile will run on the selected targets. Unless you disabled the **Capture output** option on the **Main** tab, the console displays any applicable execution results as the buildfile runs.

These steps create a persisted launch configuration. The newly created configuration will appear in the launch history under **Run > External Tools** and will be available in the launch configuration dialog which is opened by clicking **Run > External Tools > External Tools...**

■ Related concepts

[Ant support](#)

[Builds](#)

[External tools](#)

■ Related tasks

[Running external tools](#)

[Modifying the Ant classpath](#)

[Using a different version of Ant](#)

[Adding new Ant tasks and types](#)

© Copyright IBM Corporation and others 2000, 2003.

Modifying the Ant classpath

When using an optional or custom task it is usually necessary to add extra libraries to the classpath. The Ant classpath can be modified globally or per launch configuration.

To modify the Ant classpath globally:

1. Click **Window > Preferences**.
2. Expand **Ant** and select **Runtime**. If you are not on it already, click the **Classpath** tab.
3. To add a JAR file to the classpath, click **Add Jar** and select the JAR file.
4. To add a folder to the classpath, click **Add Folder** and select the folder.
5. To remove an item from the classpath, select it and click **Remove**.
6. To restore the classpath to the default, click **Restore Defaults**.

To modify the Ant classpath for a launch configuration:

1. Click **Run > External Tools > External Tools...**
2. Select the Ant configuration whose classpath you wish to modify
3. Select the **Classpath** tab
4. Uncheck the option to use the global classpath as specified in the preferences
5. To add a JAR file to the classpath, click **Add Jar** and select the JAR file.
6. To add a folder to the classpath, click **Add Folder** and select the folder.
7. To remove an item from the classpath, select it and click **Remove**.

■ Related concepts

[Ant support](#)

[External tools](#)

■ Related tasks

[Running Ant buildfiles](#)

[Running external tools](#)

© Copyright IBM Corporation and others 2000, 2003.

Using a different version of Ant

The Eclipse platform provides Ant 1.5.2 as a plug-in library. When running an Ant buildfile in the Workbench, version 1.5.2 is used by default. It is possible to use different versions, although they are not supported. There are at least two ways of using a different version of Ant:

Changing the Ant runtime classpath:

When Ant runs a buildfile, it looks for the necessary classes on the Ant classpath. The Ant classpath consists of the plug-ins contributing new tasks, types or libraries, plus the classpath defined in the Ant runtime classpath preferences. To access the preferences, click **Window > Preferences > Ant > Runtime**. The two JARs related to Ant 1.5.2 are `ant.jar` and `optional.jar`. Remove these JARs and add the ones from the desired Ant version.

Note that explicitly adding the Xerces JARs to the runtime Ant classpath is no longer required and will cause problems. The Xerces classes are loaded from the `org.apache.xerces` plug-in provided with Eclipse. For most Ant distributions, the Xerces JARs cannot even be in the same physical location as the `ant.jar` and `optional.jar`. This results from the Ant JARs containing manifest files which contain classpath entries pointing to the Xerces JARs.

After you change the Ant classpath, all future Ant builds will use the updated version instead of the default. To restore the Ant classpath to its original state, **Restore Defaults** button on the preference page.

Using Ant as an external tool:

When changing the Ant classpath is not an option, or if you just want to test a newer version or beta version of Ant, using it as an external tool can be a better solution. Usually when it is running in the Workbench, the Ant buildfile itself is considered to be a external tool, but this is not the only way. To install a binary distribution of Ant as an external tool (**Note:** These steps are for Windows, but similar methods can be used for other operating systems):

- a. Download and install the binary version of Ant from <http://ant.apache.org>.
- b. Click **Run > External Tools > External Tools...**
- c. Click **Program**
- d. Click **New**.
- e. Enter a name for your external tool (for example, External Ant).
- f. For the **Location** field, click **Browse File System**.
- g. Find and select a file called `ant.bat` (it should be in the `bin` folder of your Ant installation).
- h. In the **Arguments** field enter the arguments for your buildfile that would normally enter for running the buildfile outside of the Workbench.
- i. In the **Working Directory** field enter the directory of your buildfile.
- j. Click **Run** to execute the buildfile.

When you run Ant as an external tool, none of the tasks or types contributed by Eclipse will work. Also, the Ant classpath preference has no effect in the buildfile execution.

■ Related concepts

[Ant Support](#)
[External tools](#)

■ Related tasks

[Running Ant buildfiles](#)

Running external tools

Modifying the Ant classpath

Adding new Ant tasks and types

© Copyright IBM Corporation and others 2000, 2003.

Adding new Ant tasks and types

Add new Ant tasks and types through the Ant preferences page. These tasks and types will be available for buildfiles running in the Workbench without, having to use taskdef or typedef in the script declaration (For more on taskdef or typedef see the Ant documentation in <http://ant.apache.org>).

To add a new task or type:

1. Click **Window > Preferences**
2. Expand **Ant** and select **Runtime**.
3. Click the **Tasks** tab or the **Types** tab.
4. Click **Add Task** or **Add Type**.
5. Provide a name and class for the task or type.
6. Select the library where the task or type is declared. If the library is not present on the list, you must add it to the Ant classpath (see the Related task link below).

■ Related concepts

[Ant Support](#)

[Builds](#)

[External tools](#)

■ Related tasks

[Running external tools](#)

[Modifying the Ant classpath](#)

[Using a different version of Ant](#)

© Copyright IBM Corporation and others 2000, 2003.

Using the help system

The Workbench provides a number of ways to provide help information, including context sensitive help and extensive online documentations. See the Related tasks links for more details.

■ Related concepts

[Online help system](#)

■ Related tasks

[Accessing context sensitive help](#)

[Accessing and navigating online help](#)

[Searching online help](#)

[Setting help fonts and colors for accessibility](#)

[Changing the web browser used by the help system](#)

© Copyright IBM Corporation and others 2000, 2003.

Accessing context-sensitive help

To access the context-sensitive help for a given widget:

1. Select the widget by putting focus on it (see the table below).
2. Press F1.

This displays an *infopop* with the description of the selected widget, and usually, a list of links to related information. You can dismiss the infopop by clicking outside it, or by pressing **Esc**.

If you want more information than what is shown in the infopop, click a link to the related information. This opens the Help browser to the selected topic and closes the infopop. The list of related links stays in the left hand frame, so you do not have to press F1 again if you are interested in other topics related to the same widget.

To access context sensitive help for a widget, you must put focus on it and then press F1. The following table shows how to put focus on some of the different kinds of widgets.

Widget type	How to select it
Menu item (for example <i>Save</i> in the <i>File</i> menu)	Let your mouse pointer rest over it so that it is highlighted, but don't click on it.
Field	Put the cursor in the field.
Button (for example Cancel button on a Dialog)	Tab until the button is in focus.
List	Click an item in the list.
View or pane	Click the title bar of the View or pane.
Check box	Click the check box (which changes its state) or use the Tab key to bring it into focus.

Note: Context-sensitive help via F1 is unavailable from toolbar buttons. Instead, let your mouse pointer hover over a toolbar button to view tooltip help for buttons.

■ Related tasks

[Accessing and navigating online help](#)

© Copyright IBM Corporation and others 2000, 2003.

Accessing and navigating online help

The help browser lets you browse, search, and print online documentation for the product. To open the help browser, select **Help > Help Contents** in the Workbench. This opens the help browser to the bookshelf, which shows the major sets of documentation available.

To navigate online help:

1. Select the desired link on the bookshelf.
2. Expand the topic tree to find the information you are looking for. To view a topic, click the link in the topic tree.
3. Most topics provide a list of links to related topics at the bottom. Follow these links to learn more.
4. Use the **Go Forward** and **Go Back** buttons. These behave the same way back and forward buttons work in an Internet browser, taking you to topics you have already looked at.
5. To synchronize the navigation frame with the current topic, click the **Synchronize** button . This is helpful if you have followed several links to related topics in several files, and want to see where the current topic fits into the navigation path.

As an alternative to browsing the information this way, use the search engine. Type a query into the **Search** field at the top of the browser and click **Go**.

To see context sensitive help from the Workbench put focus on a particular widget and press F1. For more details, refer to the list of related topics below.

Maximizing help frames

To increase the space available for viewing the help content, the frame that displays content can be maximized. To maximize the frame, double click its toolbar. To return the frame to its original size, double click the tool bar again. Similarly, you can maximize the frame showing navigation, by double clicking its toolbar.

Printing online help

To print a topic from the online help:

1. Select the topic in the navigation frame.
2. Click the **Print** button  in the Help toolbar.
3. Select the desired printer settings, and click **Print**.

■ Related concepts

[Online help system](#)

■ Related tasks

[Searching online help](#)

[Accessing context sensitive help](#)

Searching online help

The help system includes a powerful text search engine that runs simple or complex queries on the documentation to help you find the information you are looking for.

To search the online help:

1. In the **Search** field at the top of the Help browser, type the term or terms for which you want to search.
2. Click **Go** or press Enter. The result set will be shown in the Search Results view of the Help browser.
3. To view the content of a topic in the result set, select it. Hits within the selected topic are highlighted.

Tip: You can also search the documentation from inside the Workbench by selecting **Search > Help**. Type in search terms, optionally select the books you want to search, and click **Search**. The results will be shown in the Search view. Double-click a result to open the help browser to that topic.

Remember the following search expression rules:

- Unless otherwise stated, there is an implied AND between all search terms. In other words, topics that contain all the search terms will be returned. For example:

```
Java project
```

returns topics that contain the word *Java* and the word *project*, but does not return topics that contain only one of these words.

- Use OR before optional terms . For example:

```
applet OR application
```

returns topics that contain the word *applet* or the word *application* (or both).

- Use NOT before terms you want to exclude from search results. For example:

```
servlet NOT ejb
```

returns topics that contain the word *servlet* and do not contain the word *ejb*. **Note:** NOT only works as a binary operator (that is, "NOT servlet" is not a valid expression).

- Use ? for a single-character wildcard and * for a multi-character wildcard. For example:

```
par?
```

returns topics that contain *part* or *park*, but not *participate*. On the other hand:

```
par*
```

returns topics that contain *part*, *park*, *participate*, *pardon*, and so on.

- Use double quotation marks around terms you want treated as a phrase. For example:

```
"creating projects"
```

returns topics that contain the entire phrase *creating projects*, and not *creating* or *project* on its own.

- Punctuation acts as term delimiters. For example:

Running Eclipse

plugin.xml

returns hits on topics that contain *plugin.xml*, *plugin*, and *xml*, which is likely broader than you want. If you want to find just those topics containing *plugin.xml*, use double quotes, as in:

"plugin.xml"

- The search engine ignores character case. For example:

Workbench

returns topics that contain 'workbench', 'Workbench', 'WorkBench', and 'WORKBENCH'.

- The following stop words are common English words which will be ignored (not searched for) if they appear in the search expression: a, and, are, as, at, be, but, by, in, into, is, it, no, not, of, on, or, s, such, t, that, the, their, then, there, these, they, to, was, will, with.
- The search engine does "fuzzy" searches and word stemming. If you enter *create*, it will return hits on topics that contain *creates*, *creating*, *creator*, and so on. To prevent search engine from stemming terms, enclose them in double quotes.

Refining the search results

If the result set is very large, the information you are looking for might not appear in the top 10 or 15 results. You can then refine the search to reduce the number of results.

To refine a search:

1. Click the **Search scope** link.
2. Click **New** button to define a new search scope.
3. In the window that opened, **select** the topics to which you want to narrow the search.
4. Give the selected list a name and click **OK**.
5. Click **OK**, to activate the new search scope.
6. Click **Go** again. The results will be shown in the Search Results view in the Help browser.

Search index generation

The first time you search the online help, the help system might initiate an index-generation process. This process builds the indexes for the search engine to use. It may take several minutes, depending on the amount of documentation.

Each time you add or modify the documentation set (for example, when you install a new feature or update an existing one), the index will be updated to reflect the new information set.

■ Related tasks

[Accessing and navigating online help](#)

© Copyright IBM Corporation and others 2000, 2003.

Setting help fonts and colors for accessibility

The help browser uses your operating system's settings for the font colors, styles, and sizes. Users with visual impairments may wish to change some of these settings to increase the readability of the documentation.

WIN In addition, on Windows platforms using Microsoft Internet Explorer, the help browser uses a component of Internet Explorer to display documentation, so changes you make to its display settings also affect the help display. To change the help browser's font and color settings:

1. Open Microsoft Internet Explorer.
2. Select **Tools > Internet Options**.
3. On the **General** page, click the **Colors, Fonts, or Accessibility** button.
4. Set the formatting options you desire.
5. Optionally, you can specify a cascading style sheet (CSS) to apply to the content.
6. Click **OK** and exit Internet Explorer.
7. Restart the Workbench. Open the Help perspective and browse the documentation to see the changes.

Note: The help system also uses the Icon font setting on the Display Properties **Appearance** tab.

For more information about creating a CSS, consult a CSS reference. The W3 Consortium (www.w3.org) has an extensive collection of information about CSS and links to valuable resources.

■ Related tasks

[Accessing and navigating online help](#)

© Copyright IBM Corporation and others 2000, 2003.

Changing the web browser used by the help system

The help system uses an adapter to display online documentation in a web browser you already have installed on your system. If you have more than one supported adapter and browser, you can select which adapter you want the help system to use. To change the web browser adapter setting:

1. Select **Window > Preferences** to open the Preferences notebook.
2. Select **Help**.
3. In the **Current web browser adapter** box, select the desired adapter. If only one adapter appears, it cannot be changed.

If your system does not have any of the supported browser adapters installed, you may need to select **Custom browser** in the Preferences, and specify a path to the browser program that launches your browser. If the program is available on your system path, you may specify only the program name. If the program does not accept URL as its last parameter, you may use *%1* string to denote the position of the URL parameter in the command.

© Copyright IBM Corporation and others 2000, 2003.

Working in the team environment

The Workbench provides tools to manage, share and synchronize resources. The CVS standard is supported by default. See the Related tasks links below for more details.

■ Related concepts

[Team programming with CVS](#)

■ Related tasks

[Working with a CVS repository](#)

[Working with projects shared with CVS](#)

[Synchronizing with the repository](#)

© Copyright IBM Corporation and others 2000, 2003.

Working with a CVS repository

In the CVS team programming environment, team members do all of their work in their own Workbenches, isolated from others. Eventually they will want to share their work. They do this via a CVS repository. The Workbench provides a number of mechanisms which support working with CVS repositories. See the Related tasks links for more details.

■ Related concepts

[Team programming with CVS
CVS Repositories](#)

■ Related tasks

[Creating a CVS repository location](#)
[Discarding a CVS repository location](#)
[Refreshing the CVS Repositories view](#)
[Discovering branch and version tags](#)

© Copyright IBM Corporation and others 2000, 2003.

Creating a CVS repository location

Prerequisite: A CVS server must already be configured on the host machine to create a valid repository location in the Workbench.

To create a new repository location:

1. Open the CVS Repositories view by selecting **Window > Show View > Other...** on the main menu bar and then selecting **CVS > CVS Repositories** from the Show View dialog. Alternatively, the CVS Repositories view is also shown in the CVS Repository Exploring perspective.
2. From the context menu of the CVS Repositories View, select **New > Repository Location**. The Add CVS Repository wizard opens.
3. Enter the information required to identify and connect to the repository location:
 - a. In the **Host** field, type the address of the host. (For example: `mymachine.com`).
 - b. In the **Repository path** field, type the path to the repository on the host (for example `/home/repo`, `d:/repo`.)
 - c. In the **User** field, type the user name under which you want to connect to the repository.
 - d. In the **Password** field, type the password for the above user name.
 - e. From the **Connection Type** list, select the authentication protocol of the CVS server. There are three connection methods that come with the Eclipse CVS client:
 - ◇ **pserver** – a CVS specific connection method.
 - ◇ **extssh** – an SSH 1.0 client included with Eclipse
 - ◇ **ext** – the CVS ext connection method which uses an external tool such as SSH to connect to the repository. The tool used by ext is configured in the **Team > CVS > EXT Connection Method** preference page.
 - f. If the host uses a custom port, enable **Use Port** and enter the port number.
4. (Optional) Select **Validate Connection on Finish** if you want to authenticate the specified user to the specified host when you close this wizard. (If you do not select this option, the user name will be authenticated later, when you try to access the contents of the repository.)
5. Click **Finish**. The repository location is created.

■ Related concepts

[Team programming with CVS](#)

[CVS Repositories](#)

■ Related tasks

[Discarding a CVS repository location](#)

[Refreshing the CVS repositories view](#)

[Checking out a project from a CVS repository](#)

[Discovering branch and version tags](#)

[Changing the properties of a CVS Repository Location](#)

■ Related reference

[CVS](#)

[CVS Repositories view](#)

Discarding a CVS repository location

Note: You cannot discard a location if you have projects in the Workbench that are shared with that location.

To discard a CVS repository location:

1. Switch to the CVS Repository Exploring perspective or add the CVS Repositories view to the current perspective.
2. In the CVS Repositories view, select the repository location that you want to delete from the Workbench.
3. Select **Discard Location** from the pop-up menu or press the DELETE key.

■ Related concepts

[Team programming with CVS](#)

[CVS Repositories](#)

■ Related tasks

[Creating a CVS repository location](#)

■ Related reference

[CVS](#)

[CVS Repositories view](#)

© Copyright IBM Corporation and others 2000, 2003.

Refreshing the CVS Repositories view

There are two ways to refresh the CVS Repositories view in order to see the current connection status of all repositories:

- Click the **Refresh View** button on the view's toolbar
- Press F5 on the keyboard

Note: Refreshing the CVS Repositories view may change the resources displayed by the view but does not change the branch and version tags known to the view. Refreshing the tags is discussed in [Discovering branch and version tags](#)

■ Related concepts

[Team programming with CVS](#)
[CVS Repositories](#)

■ Related tasks

[Creating a CVS repository location](#)
[Discarding a CVS repository location](#)
[Discovering branch and version tags](#)

■ Related reference

[CVS](#)
[CVS Repositories view](#)

© Copyright IBM Corporation and others 2000, 2003.

Discovering branch and version tags

Prerequisite: Before you can discover the branch and version tags for a repository, you must have a valid repository location in the CVS Repositories view.

To discover existing branch tags for multiple folders in a single repository location:

1. Switch to the CVS Repository Exploring perspective or add the CVS Repositories view to the current perspective.
2. In the CVS Repositories view, select a repository location.
3. From the pop-up menu for the CVS Repositories view, select **Refresh Branches**. The Refresh Branches dialog will open.
4. In the folder list, select the folders whose tags you would like to refresh. Alternatively, at the bottom of the dialog, you can choose a working set and all remote folders whose names match the name of a project in the working set will become checked.
5. Click **Finish** to discover the tags for the checked projects. These tags will be added to the repositories view.

Note: This operation may be long running especially if the number of selected folders is large.

Tip: If the above operation does not result in the discovery of any tags, it is probable that the folder in the repository does not have a .project file in it. The .project file is the default file used by the tag discovery process. The file can be changed for individual projects using *Configure Branches and Versions* (see below).

To discover existing branch and version tags for a single project folder:

1. Switch to the CVS Repository Exploring perspective or add the CVS Repositories view to the current perspective.
2. In the CVS Repositories view, expand the repository location.
3. Expand **HEAD** and select the root folder whose tags are to be discovered.
4. From the pop-up menu for the CVS Repositories view, select **Configure Branches and Versions**. The Tag Configuration dialog will open.
5. In the top pane on the left (titled *Browse files for tags*), expand one or more projects to find a file that is known to have tags that should be added to the CVS Repositories view.
6. Select a file in the left top pane. If the file selected has tags defined on it, they will be displayed in the right top pane (titled *New tags found in selected files*).
7. Check or uncheck the tags in the right pane as desired.
8. Click **Add Checked Tags** to add the checked tags to the list of remembered tags.
9. Click **OK** to add the remembered tags to the selected repository location or project.

Note: You can also set the auto-refresh files by selecting a file in the files pane and clicking *Add Selected Files* in the lower right hand portion of the dialog.

■ Related concepts

[Team programming with CVS](#)

[CVS Repositories](#)

[Branches](#)

[Versions](#)

[Synchronizing with a CVS repository](#)

■ Related tasks

[Creating a CVS repository location](#)

[Checking out a project from a CVS repository](#)

[Sharing a new project using CVS](#)

© Copyright IBM Corporation and others 2000, 2003.

Changing the properties of a CVS repository location

You can change certain properties of an existing repository location. The properties that can be changed are the connection method, user name and password.

To change the properties of a CVS repository location:

1. Switch to the CVS Repository Exploring perspective or add the CVS Repositories view to the current perspective.
2. In the CVS Repositories view, select the repository location whose properties you want to change.
3. Select **Properties** from the pop-up menu. The properties dialog will open.
4. Change one or more of the connection method, user name and password.
5. Click **OK** to change the properties.

Note: The properties will be changed for all projects shared with the selected repository location.

■ Related concepts

[Team programming with CVS](#)

[CVS Repositories](#)

■ Related tasks

[Creating a CVS repository location](#)

[Changing the sharing of a Project](#)

■ Related reference

[CVS](#)

[CVS Repositories view](#)

© Copyright IBM Corporation and others 2000, 2003.

Setting the content type of a file extension

CVS repositories distinguish between files that contain ASCII data and those that do not. For ASCII files, additional helpful functionality is available to the user. This includes:

- Proper end of line conversion between the client and the server. This ensures that a Windows user and a UNIX user can work on the same file without introducing incompatibilities.
- Auto merging of conflicts. If a file contains both incoming and outgoing changes but none on the same line as another, then the file may be automatically merged when updating.

Certain file types invariably contain either ASCII data or binary data. For example, *.txt files usually contain ASCII data, and *.exe files usually contain binary data. Eclipse comes with a pre-defined set of file types, which you may add to or modify.

To set the content type associated with a file extension:

1. From the main menu bar, select **Window > Preferences** to open the Workbench Preferences dialog.
2. In the dialog, select the **Team > File Content** preference page. This page displays a list of file extensions and content type (ASCII or Binary) for file extensions whose content type is known.
3. To add a file extension, click the **Add** button and enter the file extension in the text prompt that appears. Once **OK** is clicked, the extension will be added to the list with a content type of ASCII.
4. To change the content type for an existing file extension, select the file extension entry and click the **Change** button. This will toggle the type from ASCII to Binary or vice versa.
5. To remove a file extension, select the file extension entry and click the **Remove** button.

■ Related concepts

[Team programming with CVS](#)

■ Related reference

[CVS](#)

© Copyright IBM Corporation and others 2000, 2003.

Working with projects shared with CVS

The Workbench provides a number of mechanisms which support working with projects shared via CVS. See the Related tasks links for more details.

■ Related concepts

[Team programming with CVS](#)

[CVS Repositories](#)

[Branches](#)

[Synchronizing with a CVS repository](#)

■ Related tasks

[Sharing a new project using CVS](#)

[Project checked out with another CVS tool](#)

[Checking out a project from a CVS repository](#)

[Checking out a module from a CVS repository](#)

[Disconnecting a project from CVS](#)

[Setting the CVS keyword substitution mode](#)

[Enabling the CVS resource decorations](#)

© Copyright IBM Corporation and others 2000, 2003.

Enabling the CVS resource decorations

When enabled, CVS will decorate resources in Workbench views with icon and label decorations that indicate the CVS state of the resource.

To enable the CVS decorations:

1. From the main menu bar, select **Window > Preferences** to open the Workbench Preferences dialog.
2. In the dialog, select the **Workbench > Label Decorations** preference page. This page allows the enabling and disabling of all decorations defined by different plug-ins in the Workbench.
3. Click on the CVS box to enable the CVS decorations.
4. Click **OK**.

Note: In some cases, the decorations may not appear immediately due to the fact that they are expensive to compute. In those cases, they are calculated in the background.

To configure the CVS decorations:

1. From the main menu bar, select **Window > Preferences** to open the Workbench Preferences dialog.
2. In the dialog, select the **Team > CVS > Label Decorations** preference page. This page allows the configuration of how decorations appear on CVS projects, folders and files.
3. To configure the text label decorations click the **Text** tab and then:
 - a. Pick the resource type (file, folder, project) for which you want to change the decoration.
 - b. Click in the decoration string for that resource type at the location where the new decoration element should go.
 - c. Click the **Add Variables** button for that resource type.
 - d. Choose the CVS information variable to add. For instance, for files you could select the *added_flag* which will be inserted in the decoration where you placed the cursor. The string *{added_flag}* will be added to the decoration string which will decorate newly added resources using the added resource label (which can be set in the *Label Decoration for Added* field).
4. Repeat the above for any other decorations to be added.
5. Decorations can be removed by deleting the decoration variable name (i.e. *{added_flag}*) from the decoration string. In a similar manner, spacing can be added.
6. To configure icon decorations, click the **Icon** tab simply enable/disable the icon checkboxes.
7. Click **OK** or **Apply** for the new decorations to take effect.

Note: Under the **General** tab is an option to disable the deep dirty decoration of folders. This will make decorator determination more efficient but will make it harder to find dirty resources in the Navigator and other views. Under the **Synchronize View** tab, the Synchronize view can be configured to show the synchronization state of a resource as part of the text label displayed in the Structure Compare pane.

■ Related concepts

[Team programming with CVS](#)

[Synchronizing with a CVS repository](#)

■ Related tasks

[Checking out a project from a CVS repository](#)

[Sharing a new project using CVS](#)

■ Related reference

[CVS](#)

[CVS Label Decorations](#)

© Copyright IBM Corporation and others 2000, 2003.

Sharing a new project using CVS

There are several scenarios that can occur when sharing a project using CVS. The most common scenario is sharing a new project using CVS when the project does not already exist remotely.

To share a new project using CVS:

1. In the Navigator view, select the project to be shared.
2. Select **Team > Share Project...** from the project's pop-up menu. The Share Project wizard opens.
3. From the list of available repository providers, choose **CVS** and click **Next** to go to the next page. (Note: If there is only one repository provider available, the first page will be skipped and next page will be displayed automatically.)
4. Select the target repository from the list of known repositories or, if the target repository is not in this list, choose to create a new repository location and click **Next**.
5. If entering a new repository location, enter the repository information and click **Next** when completed. (Note: this page is the same format as the [Creating a CVS repository location](#) wizard.)
6. (Optional) Either choose to use the name of the local project as the remote project name or enter another name.
7. Click **Finish** to share the project with the repository. The project folder will be created remotely and the Synchronize view will open and allow the committing of the project's resources. (Note: If the project already exists remotely, the Synchronize view will show conflicts on any files that exist both locally and remotely.)

Consequences for Linked Resources

As mentioned in [Linked resources](#), different providers may handle linked resources differently. For Team CVS, linked resources are allowed but ignored. Specifically, projects which contain linked resources can be shared with CVS, but the linked resources themselves cannot be version controlled.

■ Related concepts

[Team programming with CVS](#)

[Branches](#)

[Synchronizing with a CVS repository](#)

■ Related tasks

[Project checked out with another CVS tool](#)

[Checking out a project from a CVS repository](#)

[Replacing resources in the Workbench](#)

■ Related reference

[CVS](#)

© Copyright IBM Corporation and others 2000, 2003.

Project checked out with another CVS tool

It is possible to use a CVS project checked out using another CVS tool into Eclipse. A common concern for those migrating to Eclipse is that they already have their CVS project checked out on their local machine. You *don't* have to check out the contents again to create a project in Eclipse. Although the project already contains the CVS sharing information in the CVS folders, it still needs to be shared within Eclipse in order to enable the CVS Team menu operations.

To share an CVS project that was already checked out::

1. In the Navigator view, select the project to be shared.
2. Select **Team > Share Project...** from the project's pop-up menu. The Share Project wizard opens.
3. From the list of available repository providers, choose **CVS** and click **Next** to go to the next page.
Note: If there is only one repository provider available, the first page will be skipped and next page will be displayed automatically.
4. The next page displays the sharing information stored in the CVS folder of the project. Click **Finish** to share the project and enable the CVS Team menu operations.

■ Related concepts

[Team programming with CVS](#)

[Synchronizing with a CVS repository](#)

■ Related tasks

[Sharing a new project using CVS](#)

■ Related reference

[CVS](#)

© Copyright IBM Corporation and others 2000, 2003.

Checking out a project from a CVS repository

To check out a project from a CVS repository to the Workbench:

1. Switch to the CVS Repository Exploring perspective or add the CVS Repositories view to the current perspective.
2. In the CVS Repositories view, expand the repository location.
3. Expand **HEAD** and select the folders that you want to add as projects to the Workbench. If you are looking for a folder in a particular version:
 - a. Expand the **Versions** category and find the folder name that you want to add to the Workbench.
 - b. Expand the folder to reveal its versions.If you are looking for the latest folder in a branch:
 - a. Expand the **Branches** category.
 - b. Expand the branch that holds the folder that you want to add to the Workbench.
4. From the pop-up menu for the selected folders, select one of the following:
 - a. **Check Out as Project** to check out each of the selected folders as a project in the local workspace with the same name as the folder in the repository.
 - b. **Check Out As...** to check out the selected folders into a custom configured project in the local workspace. *Note:* When multiple folders are selected, this operations only allows the specification of a custom parent location for the new projects.
5. If **Check Out As...** was chosen on a single project, one of two possible dialogs is displayed depending on whether the folder in the repository contains a .project file or not.
 - a. If there is a .project file, the dialog will accept a custom project name and location.
 - b. Otherwise, the dialog will be the New Project wizard which allows full customization of the project (e.g. Java project).

Tip: Any folder, including non-root folders, can be checked out from a CVS repository.

■ Related concepts

[Team programming with CVS](#)
[CVS Repositories](#)

■ Related tasks

[Checking out a module from a CVS repository](#)
[Checking out a folder into an existing project](#)
[Creating a CVS repository location](#)
[Replacing resources in the Workbench](#)
[Discovering branch and version tags](#)
[Sharing a new project using CVS](#)
[Synchronizing with the repository](#)

■ Related reference

[CVS](#)
[CVS Repositories view](#)

Checking out a module from a CVS repository

Modules can be defined in the CVSROOT/modules file of a CVS repository. To check out such a module from a CVS repository to the Workbench:

1. Switch to the CVS Repository Exploring perspective or add the CVS Repositories view to the current perspective.
2. In the CVS Repositories view, expand the repository location.
3. Expand **HEAD** and select the modules that you want to add to the Workbench. *Note:* Modules appear as root folders of the repository but are associated with the  icon.
4. From the pop-up menu select **Check Out Module**. This will checkout the module into one or more projects in the local workspace, depending on how the module is defined in the repository.

■ Related concepts

[Team programming with CVS
CVS Repositories](#)

■ Related tasks

[Checking out a project from a CVS repository](#)
[Checking out a folder into an existing project](#)
[Creating a CVS repository location](#)
[Replacing resources in the Workbench](#)
[Discovering branch and version tags](#)
[Sharing a new project using CVS](#)
[Synchronizing with the repository](#)

■ Related reference

[CVS](#)
[CVS Repositories view](#)

© Copyright IBM Corporation and others 2000, 2003.

Checking out a folder into an existing project

To check out a folder from a CVS repository into an existing project in the Workbench:

1. Switch to the CVS Repository Exploring perspective or add the CVS Repositories view to the current perspective.
2. In the CVS Repositories view, expand the repository location.
3. Expand **HEAD** and select the folder that you want to check out. If you are looking for a folder in a particular version:
 - a. Expand the **Versions** category and find the folder name that you want to check out.
 - b. Expand the folder to reveal its versions.If you are looking for the latest folder in a branch:

- a. Expand the **Branches** category.
 - b. Expand the branch that holds the folder that you want to check out.
4. Select **Check Out Into...** from the pop-up menu for the selected folder. The *Check Out Into* dialog will open.
 5. In the *Check Out Into* dialog, you can provide the following information:
 1. **Target folder name:** The name of a local folder in which the contents of the remote folder will be placed. *Note:* If the local folder exists, its contents will be deleted and replaced by the contents of the remote folder.
 2. **Parent of target folder:** The existing local folder into which the target folder will be created.
 3. **Checkout sub-folders:** The option to indicate whether the sub-folders of the remote folder should be checked out as children of the target folder.

Note: Only folders within non-shared projects or projects shared with the same CVS repository as the selected remote folder are valid targets for the *Checkout Into* operation. Also, if the target project of the operation is an unshared project, the project will be connected to the CVS repository (i.e. the project will become a shared CVS project) but any pre-existing content will be ignored.

■ Related concepts

[Team programming with CVS](#)
[CVS Repositories](#)

■ Related tasks

[Checking out a project from a CVS repository](#)
[Checking out a module from a CVS repository](#)
[Creating a CVS repository location](#)
[Replacing resources in the Workbench](#)
[Discovering branch and version tags](#)
[Sharing a new project using CVS](#)
[Synchronizing with the repository](#)

■ Related reference

[CVS](#)
[CVS Repositories view](#)

Disconnecting a project from CVS

Disconnect a project from CVS to disable the CVS operations that can be performed on the project and its resources and optionally to remove the CVS information (stored in the CVS folders) associated with the project.

To disconnect a project from CVS:

1. In the Navigator view, select the project to be disconnected.
2. Select **Team > Disconnect** from the project's pop-up menu. The Confirm Disconnect from CVS dialog opens.
3. In the dialog, choose one of:
 - a. **Delete the CVS meta information** – disables the CVS team menu operations and removes the CVS folders and their contents from the file system.
 - b. **Do not delete the CVS meta information** – disables the CVS team menu operations but leaves the CVS meta information.
4. Click **Yes** to disconnect the project.

■ Related concepts

[Team programming with CVS](#)

[Branches](#)

[Synchronizing with a CVS repository](#)

■ Related tasks

[Checking out a project from the CVS repository](#)

[Replacing resources in the Workbench](#)

[Branching](#)

[Synchronizing with the repository](#)

[Updating](#)

[Resolving conflicts](#)

[Merging from a branch](#)

■ Related reference

[CVS](#)

© Copyright IBM Corporation and others 2000, 2003.

Setting the CVS keyword substitution mode

CVS uses the keyword substitution mode of a file to differentiate binary files from ASCII files and to indicate what type of keyword substitution is to take place when files are committed and checked out.

To set the CVS keyword substitution mode:

1. In the Navigator view, select the files or containing folders for which a change in keyword substitution mode is desired. **Note:** Ensure that any new files that are to be committed are added to CVS version control as the keyword substitution mode can only be set for files that are already under CVS control.
2. From the pop-up menu, select **Team > Change ASCII/Binary Property**. The Set Keyword Substitution Mode wizard will open.
3. On the first page of the wizard, select the mode to be applied to the selected files:
 - ◆ **Automatic** – uses the file type to determine whether selected resources should use binary or the default ASCII mode (the default ASCII mode is set on the Team > CVS preference page).
 - ◆ **Binary** – sets all selected files to binary. Files that are binary are not altered in any way by CVS.
 - ◆ **ASCII without keyword substitution** – sets all the selected files to ASCII but CVS does not perform any keyword substitution. The line endings of ASCII files may be altered by CVS to align with a platform's line-terminator convention.
 - ◆ **ASCII with keyword substitution** – sets all the selected files to ASCII and CVS will perform the selected type of keyword substitution.

Click **Next** to go to the next page.

4. On the **Include versioned files** page of the wizard, indicate whether files already shared in the repository are to be affected. If this setting is disabled, only files that have been added to CVS control but not committed will be affected. This page will only appear if there are files already shared with the repository. Click **Next** to go to the next page.
5. On the **Include files with outgoing changes** page of the wizard, indicate whether files that have outgoing changes are to be affected. This page will only appear if there are files with outgoing changes. Click **Next** to go to the next page.
6. On the **Enter Commit Comment** page of the wizard, enter the commit comment to be associated with any file commits. Files will need to be committed if changing the file type from Binary to ASCII results in a change in the file content due to line terminator adjustments. Click **Next** to go to the next page.
7. On the **Summary** page of the wizard, inspect the proposed keyword substitution mode settings to ensure they are correct. Click **Finish** to apply the changes.

Tip: On the Summary page, the assigned mode for a file can be changed by clicking on the proposed mode in the table and selecting the desired mode from the drop down list that becomes available.

■ Related concepts

[Team programming with CVS](#)

[Branches](#)

[Synchronizing with a CVS repository](#)

■ Related tasks

[Creating a CVS repository location](#)

[Checking out a project from the CVS repository](#)

[Replacing resources in the Workbench](#)

[Branching](#)

[Synchronizing with the repository](#)

[Updating](#)

[Resolving conflicts](#)

[Merging from a branch](#)

■ **Related reference**

[CVS](#)

[File Content](#)

© Copyright IBM Corporation and others 2000, 2003.

Filtering in the CVS Resource History View

Over time, the revision history for a file can grow drastically. As a result, it may be difficult to find the revision or revisions that you are looking for. Using the filtering mechanism of the CVS Resource History view can help.

To set a filter on the CVS Resource History view:

1. Select the **Filter History** action from the local toolbar of the CVS Resource History view.
2. Enter any author, comment, or date range that you would like the view to display.
3. Click **OK**.

The view will now display only those revisions that match the criteria you entered.

Tip: You can remove a date range filter by selecting the first item ("----") in the date and month fields.

Tip: You can choose whether the filter should match any or all of the criteria by selecting the appropriate radio button in the dialog.

■ Related concepts

[Team programming with CVS](#)

■ Related reference

[CVS](#)

[CVS Resource History view](#)

© Copyright IBM Corporation and others 2000, 2003.

Changing the sharing of a project

If a project is shared with a CVS repository location, you can change the sharing information so it is shared with a different repository location. The new location must map to the same repository but can have a different connection method and/or user name.

To change the sharing of a project:

1. In the Navigator view, select the project whose sharing information is to be changed.
2. Select **Properties** from the pop-up menu. The properties dialog will open.
3. Select **CVS** to see the CVS properties page for the project.
4. Click **Change Sharing...** A repository selection dialog opens containing only the compatible repository locations.
5. Select the desired location and click **OK** to change the sharing for the project.

■ Related concepts

[Team programming with CVS](#)

[CVS Repositories](#)

■ Related tasks

[Creating a CVS repository location](#)

[Changing the properties of a CVS Repository Location](#)

■ Related reference

[CVS](#)

[CVS Repositories view](#)

© Copyright IBM Corporation and others 2000, 2003.

Synchronizing with the repository

Method 1: Using the context menu

To synchronize resources in the Workbench with those in the repository:

1. In the Navigator view, select the resources that you want to synchronize.
2. Right click and select **Team > Synchronize with Repository**. The Synchronize view opens.

Method 2: Using the synchronize action

1. From the CVS Repository Exploring perspective select the  **Synchronize CVS projects** button.
2. Select to either include all projects that are shared with CVS or specify a working set. Then select **OK**.
3. The Synchronize view will open.

Note: The synchronize action is not enabled by default in other non-CVS specific perspectives. You can enable the action to appear in your current perspective by selecting **Window > Customize Perspective**. Then enable the CVS item via **Other > CVS**.

From within the synchronize view

1. Use the toolbar buttons to switch modes for this view. There are three modes:
 - ◆ Incoming mode – shows incoming changes only (resources in the repository that differ from what is in the Workbench).
 - ◆ Outgoing mode – shows outgoing changes only (resources modified in the Workbench).
 - ◆ Incoming/Outgoing mode – shows both incoming and outgoing changes.

Important: It is preferable to update resources in the Workbench first, resolve any conflicts that exist by merging, then commit Workbench resources to the repository.
2. It is possible that someone has committed a new revision of your file since you started working on it. This will result in a *conflict*, and care must be taken to resolve this. To see only conflicts, click the **Show Only Conflicts** button on the view's toolbar.

The resources displayed in the Structure Compare area will depend on which mode is used in the Synchronize view. In the Text Compare area, the Workbench resource is on the left side, and the repository resource is on the right side.

■ Related concepts

[Team programming with CVS](#)

[Branches](#)

[Synchronizing with a CVS repository](#)

[Three way comparisons](#)

■ Related tasks

[Committing](#)

[Updating](#)

[Resolving conflicts](#)

[Comparing resources](#)

[Merging changes in the Compare editor](#)

[Merging from a branch](#)

[Version control life cycle: adding and ignoring resources](#)

[Replacing resources in the Workbench](#)

■ Related reference

[CVS](#)

[Synchronize view](#)

© Copyright IBM Corporation and others 2000, 2003.

Updating

While you are working on a project in the Workbench, other members of your team may be committing changes to the copy of the project in the repository. To get these changes, you may "update" your Workbench to match the state of the branch. The changes you will see will be specific to the branch that your Workbench project is configured to share. You control when you choose to update.

The update command can be issued from two places: the **Team > Update** menu, or the **Synchronize** view. In order to understand the difference between these two commands, it is important to know about the three different kinds of incoming changes.

- A *non-conflicting* change occurs when a file has been changed remotely but has not been modified locally.
- An *automergable conflicting* change occurs when an ASCII file has been changed both remotely and locally (i.e. has non-committed local changes) but the changes are on different lines.
- A *non-automergable conflicting* change occurs when one or more of the same lines of an ASCII file or when a binary file has been changed both remotely and locally (binary files are never automergable).

When you select **Team > Update**, the contents of the local resources will be updated with incoming changes of all of the above three types. For non-conflicting and automergable conflicts, there is no additional action required (for automergable conflicts, the changed local resource is moved to a file prefixed with ".#" just in case the automerge wasn't what the user wanted). However, for non-automergable conflicts, the conflicts are either merged into the local resource using special CVS specific markup text (for ASCII files) or the changed local resource is moved to a file prefixed with ".#" (for binary files). This matches the CVS command line behavior but can be problematic when combined with the Eclipse auto-build mechanism. Also, it is often desirable to know what incoming changes there are before updating any local resources. These issues are addressed by the Synchronize view.

To open the Synchronize view in incoming mode:

1. In the Navigator view, select the resources which you want to update.
2. From the pop-up menu for the selected resources, select **Team > Synchronize with Repository**. The Synchronize view will open.
3. On the toolbar of the Synchronize View, click the **incoming mode** button to filter out any modified Workbench resources (outgoing changes) that you may have.

In incoming mode, you will see changes that have been committed to the branch since you last updated. The view will indicate the type of each incoming change (non-conflict, automergable conflict or non-automergable conflict). There are two update commands (available from the context menu of any resource in the view) to deal with the different types of conflicts: **Update from Repository** and **Override and Update**. When you select the **Update from Repository** command in the Synchronize view, only non-conflicting changes are processed, leaving any files that have automergable or non-automergable conflicts in the view (any files that have been successfully processed are removed from the view). The **Override and Update** command operates on the two types of conflicts. After selecting this command, you will be prompted before a merge is attempted and asked if you want to auto merge the contents or overwrite them with the repository file. If you select to auto merge then only automergable conflicts will be processed and the incoming changes will be automerged with the local changes. Otherwise all conflicts will be processed and the local resources will be replaced with the remote contents. This "replace" behavior is rarely what is desired. An alternative is described later.

Running Eclipse

To update non-conflicting and automergable files:

1. The Structure Compare pane at the top of the Synchronize view contains the hierarchy of resources with incoming changes.
2. Select the non-conflicting files and choose **Update from Repository** from the pop-up menu. This will update the selected resources and remove them from the view.
3. Select the automergable conflicts and choose **Override and Update** from the pop-up menu. Select to only update automergable resources and click OK when prompted. This will update the selected resources and remove them from the view.

If your local Workbench contains any outgoing changes that are not automergable with incoming changes from the branch, then, instead of performing an **Override and Update**, you can merge the differences into your Workbench manually, as follows:

1. In the Structure Compare pane, if there is a conflict in the resource list (represented by red arrows), select it.
2. In the Text Compare area of the Synchronize view, local Workbench data is represented on the left, and repository branch data is represented on the right. Examine the differences between the two.
3. Use the text compare area to merge any changes. You can copy changes from the repository revision of the file to the Workbench copy of the file and save the merged Workbench file (using the pop-up menu in the left pane).
4. Once you are completed merging the remote changes into a local file, choose **Mark as Merged** from the pop-up menu. This will mark the local file as having been updated and allow your changes to be committed.

Note: The repository contents are not changed when you update. When you accept incoming changes, these changes are applied to your Workbench. The repository is only changed when you commit your outgoing changes.

Tip: In the Structure Compare pane, selecting an ancestor of a set of incoming changes will perform the operation on all the appropriate children. For instance, selecting the top-most folder and choosing **Update from Repository** will process all non-conflicting incoming changes and leave all other incoming changes unprocessed.

Warning: The behavior of the **Override and Update** command described above only applies to the incoming mode of the Synchronize view. In the **incoming/outgoing mode** of the view, the behavior for incoming changes and conflicts is the same but the command will revert outgoing changes to whatever the repository contents are. Exercise great caution if using this command in incoming/outgoing mode.

■ Related concepts

[Team programming with CVS](#)

[Synchronizing with a CVS repository](#)

■ Related tasks

[Committing](#)

[Resolving conflicts](#)

[Comparing resources](#)

[Version control life cycle: adding and ignoring resources](#)

■ Related reference

CVS

Synchronize view

© Copyright IBM Corporation and others 2000, 2003.

Committing

You can commit Workbench resources that you have modified to the repository so that other team members can see your work. Only those changes committed on that branch will be visible to others working on that branch. The commit command can be issued from two places: the **Team > Commit** menu, or the **Synchronize** view.

To commit changes using **Team > Commit**:

1. In the Navigator view, select the resources that you want to commit.
2. Right-click on the resources and select **Team > Commit** from the pop-up menu.
3. In the Commit Comment dialog box, provide a comment for your changes (for example, Fixed the spelling mistakes).

If there are conflicting changes on any files that are committed in the above fashion, the operation will fail. If this occurs, you must either perform an update or use the Synchronize view to resolve the conflicts. It is considered a more ideal workflow to always update before committing in order to ensure that you have the latest state of the repository before committing more changes.

If one or more of the resources being committed are new and not yet added to CVS control, you will be prompted to add these resources before the commit is performed. You can choose to add all, some or none of the new resources to CVS control before performing the commit. Any resources that are not under CVS control will not be committed. Committing from the Synchronize view also prompts if there are new resources.

To commit changes in the Synchronize view:

1. In the Navigator view, select the resources that you want to commit.
2. Right-click to open the pop-up menu and select **Team > Synchronize with Repository**. The Synchronize view will open.
3. On the toolbar of the Synchronize view, select the **outgoing mode** button to show any modified Workbench resources (outgoing changes) that you may have.
4. If there are conflicts (red arrows), resolve them. Use the text compare area to merge resources with conflicts. You can copy changes from the repository revision of the file to the Workbench revision of the file and save the merged Workbench resource. Once all the conflicts in the Structure Compare area have been resolved, you are ready to commit.
5. In the Structure Compare pane, right-click the top of the hierarchy that you want to commit, and select **Commit** from the pop-up menu.
6. In the Commit Comment dialog box, provide a comment for your changes (for example, Fixed the spelling mistakes).

Tip: You can commit files that are in conflict by performing an **Override and Commit**. This will commit the Workbench copy of the resource into the repository and thus remove any of the incoming changes.

Warning: The behavior of the **Override and Commit** command described above only applies to the outgoing mode of the Synchronize view. In the **incoming/outgoing mode** of the view, the behavior for outgoing changes and conflicts is the same but the command will revert incoming changes to whatever the local Workbench contents are. Exercise great caution if using this command in incoming/outgoing mode.

■ Related concepts

[Team programming with CVS](#)

[CVS Repositories](#)

[Branches](#)

[Synchronizing with a CVS repository](#)

■ Related tasks

[Updating](#)

[Resolving conflicts](#)

[Comparing resources](#)

[Version control life cycle: adding and ignoring resources](#)

■ Related reference

[CVS](#)

[Synchronize view](#)

© Copyright IBM Corporation and others 2000, 2003.

Version control life cycle: adding and ignoring resources

When committing resources, often there are resources that you do not want to store in the repository. For example, external editors might create temporary files in your project, compilation of .java files might create .class files, or some build operations might result in binary files. When put together, these generated files may be quite large. They may also be regenerated whenever a build is performed, resulting in many outgoing changes. Typically these are not files that one wants to persist in the repository or to share with other members of a team.

Team CVS has two related tasks that allow you to control which files are stored in the repository: adding a resource to version control, and ignoring a resource.

Adding a file to version control

Team CVS does not automatically add files to version control. Rather, it's your choice to explicitly add files to version control. This is accomplished by selecting the **Team > Add to Version Control** menu.

When performed on a file, it will add that file to version control. The result is that the CVS repository immediately creates an entry so that it can start maintaining history state for that file. This occurs even before you commit the file to the repository.

When adding a folder or project, the action will recursively descend into sub-folders, adding those files it finds to version control, provided the files have not been explicitly ignored.

Tip: When committing files, if the selection either directly or recursively contains files which have not been added to version control, you will be prompted whether or not you want them added. This is a convenience function to help ensure that you do not miss committing new resources which were unintentionally never added.

How may I ignore thee, let me count the ways

There are several facilities that allow you to specify which resources should be excluded from version control:

1. There is a global preference which you can use for ignoring files and directories that match a certain filename pattern. For example, if you create a global ignore for `/bin`, and any resource that matches "bin" in any directory in the workspace will be ignored for version control. This preference can be found in **Window > Preferences > Team > Ignored Resources**.
2. Any resource marked as *derived* will be automatically ignored for version management by Team CVS. Some builders, such as the Java builder, mark all of its build output (e.g. .class files) as derived.
3. CVS supports the creation of a special `.cvsignore` file whose contents describe which files or folders to ignore for version management. The `.cvsignore` file only applies the pattern to resources in the *same* directory as the `.cvsignore` file itself.

Tip: Once a resource is under version control, it cannot easily be subsequently ignored. This is why adding to version control is an explicit operation performed by you the user.

■ Related concepts

Team programming with CVS

■ **Related tasks**

Creating a global ignore pattern

Authoring the CVS .cvsignore file

Committing

Updating

© Copyright IBM Corporation and others 2000, 2003.

Creating a global ignore pattern

When synchronizing resources, often there are resources that you do not want to commit to the repository. The Workbench provides a global pattern matching facility for ignoring resources that can appear anywhere in your project hierarchy.

1. From the main menu bar, select **Window > Preferences**.
2. In the left pane of the Preferences window, expand the **Team** category and select **Ignored Resources**. This displays a list of resource name patterns against which resources will be screened, before they are considered as candidates for version control.
3. Click the **Add** push button and type the pattern that you want to match, for example `*.obj`.
4. Click **OK**.

All files with an extension that matches the pattern that you added will be excluded from version control.

The patterns that you define may contain the wildcard characters `*` and `?`. The asterisk represents any sequence of zero or more characters, the question mark represents a single character. For example, you can specify a pattern of `*~`, which would match any temporary files that end with a tilde (`~`). Any file or directory that matches any one of the patterns will be ignored when files are being considered for version control. You can temporarily disable ignoring the resource name pattern by deselecting it from the list. You do not have to remove the specified pattern from the list.

It is important to note that the path leading up to the resource name is not included in the matching. For example, for the file `"/path/to/file.txt"`, only the string `"file.txt"` is matched against the patterns. This facility is not intended for specifying fully-qualified path names but for specifying globally applicable patterns.

■ Related concepts

[Team programming with CVS](#)

■ Related tasks

[Version control life cycle: adding and ignoring resources](#)

[Overriding or removing resource ignore patterns](#)

[Synchronizing with the repository](#)

© Copyright IBM Corporation and others 2000, 2003.

Authoring the CVS .cvsignore file

When committing resources, often there are resources that you do not want to add to version control. One way you can do this is by using the CVS "ignore" facility, which reads the contents of the file *.cvsignore* to determine what to ignore.

The *.cvsignore* file can be added to any directory of a project. Many existing CVS projects already contain several of these files. This text file consists of a list of files, directories, or patterns.

For example, to add a *.cvsignore* file to ignore the entire *bin* directory of an existing project:

1. In the Navigator view, select a project that contains a */bin* directory. The *bin* directory will commonly contain the projects build output. These are files that are generated from the project's source files and are usually not version-controlled.
2. From the pop-up menu for the project, select **New > File**.
3. Type *.cvsignore* as the file name, then press **Finish**. The file will be created in your project's root directory. You should see it in the Navigator view.
4. Enter *bin* in the *.cvsignore* file and save it.
5. Select the project. From the context menu, select the **Team > Synchronize with Repository** menu item. Notice that the *bin* directory does not show as an outgoing change. It is ignored.

The *cvsignore* file consists of a list of files, directories, or patterns. In a similar way to the global ignore facility, the wildcards *** and *?* may be present in any entry in the *.cvsignore* file. Any file or sub-directory *in the current directory* that matches any one of the patterns will be ignored.

Tip: In the Team menu and in the Synchronize view resource context menu there is a menu item (**Add to .cvsignore**) for adding a file pattern to the appropriate *.cvsignore* file. This menu item is enabled for resources that are not yet under CVS version control.

■ Related concepts

[Team programming with CVS](#)

■ Related tasks

[Synchronizing with the repository](#)

[Version control life cycle: adding and ignoring resources](#)

[Creating a global ignore pattern](#)

■ Related reference

www.cvshome.org: Ignoring files via cvsignore

© Copyright IBM Corporation and others 2000, 2003.

Overriding or removing resource ignore patterns

After you have added global file–matching patterns to the list of files to be ignored for version management, you may wish to occasionally override the "ignore".

1. From the main menu bar, select **Window > Preferences > Team > Ignored Resources**.
2. To temporarily override one or more of the file–matching patterns, clear their checkboxes.
3. Click **OK**.

Files that match the patterns you have deselected will be considered for version control.

To permanently remove a pattern, select it and click **Remove**.

■ Related concepts

[Team programming with CVS](#)

■ Related tasks

[Version control life cycle: adding and ignoring resources](#)

[Creating a global ignore pattern](#)

[Synchronizing with the repository](#)

© Copyright IBM Corporation and others 2000, 2003.

Resolving conflicts

When updating or committing you may encounter conflicts. A conflict occurs when you have locally modified a resource for which a more recent revision is available in the branch in the repository. Specifically, the branch will contain a revision newer than the base revision of your resource. In this situation you can choose to do one of the following:

- You can take the change from the branch, discarding your local work. This could occur if you have made unintended changes locally, or if you realize that the revision in the repository is better than yours. Overwriting your local changes should be done with caution since you are in essence throwing work out.
- You can commit your change, subsuming the revision in the repository. This should be done with great caution since you are in essence throwing away someone else's work. In particular, the change you are overwriting may have other dependencies in the branch (for example there may be other incoming changes which depend on the conflict).
- You can merge your work and the repository resource, saving locally the merged resource. You may then later choose to commit this merged result.

Typically, you will want to take the third option, that is to merge, because of the loss of work issues with the other two choices.

Manually merging changes

The Synchronize View indicates those resources which are in conflict with the branch. For a given resource in conflict, typically you will want to merge your changes with changes in the branch's resource.

For example, let us assume that both you and another team member have modified the same html page. Selecting that resource in the Synchronize view will display a comparison of the local resource and the branch revision. By cycling through and merging the individual changes, you can decide for each change whether to accept the incoming change, reject it, or merge it with your local changes. When you are finished merging, you save your changes. This overwrites your local resource with the result of the merge. You can subsequently commit this merged resource.

Tip: When merging changes, it is often convenient to be able to distinguish which files you have completed merging. When you're done merging a file, you can pick **Mark as Merged** from the context menu. This will change the status of the file from being a conflict to being an outgoing change.

You can merge differences in the Synchronization view on two levels:

- **Whole Document** – In the Structure Compare editor, select the resource that you want to merge, so that the resource is displayed in the Text Compare editor. In the Text Compare editor, click the Copy whole document from right to left button to entirely replace the text in your local editor with the contents of the branch resource.
- **Current Change** – In the Text Compare editor, either use the Go to Next Difference and Go to Previous Difference buttons to navigate to the change that you want to merge, or simply click in either source pane somewhere in the desired change difference fragment. Click the Copy current change from right to left button as needed to overwrite the highlighted fragment either with the corresponding modification in the branch.

Auto merging changes

It is also possible to have your changes automatically merged for you. For any resource marked as ASCII, performing a **Team > Update** will automatically merge into your local resource differences with the branch resource. This works fine provided there are no lines with conflicting changes. If there are, CVS inserts special markup in the file to indicate those lines which could not be merged.

Updating from within the Synchronization view works a bit differently. In the case of a conflict, updating will prompt you to either merge in the changes or replace the entire file contents with those of the remote.

■ Related concepts

[Team programming with CVS](#)

[Synchronizing with a CVS repository](#)

[Three way comparisons](#)

■ Related tasks

[Synchronizing with the repository](#)

[Updating](#)

[Committing](#)

[Merging from a branch](#)

[Comparing resources](#)

[Merging changes in the Compare editor](#)

■ Related reference

[www.cvshome.org: Bringing a file up to date](http://www.cvshome.org)

[www.cvshome.org: Conflicts example](http://www.cvshome.org)

[Compare editor](#)

Branching

Creating a branch and releasing resources to that branch is useful in cases where you are not yet ready to put your changes in the main development flow. It is also useful for creating incremental patches to existing versions.

To create a branch:

1. From the context menu of the project, select **Team > Branch**.
2. Enter the Branch Name for the new branch in the text area.
3. If the project in your workspace is not already a version, enter the Version Name to create in the text area.
4. If you would like to start working in the branch immediately, make sure the checkbox is selected.
5. Click **OK**.

Note: The Version Name is important; you will need this when you want to merge your changes later on. It identifies the point at which the branch was created.

■ Related concepts

[Team programming with CVS](#)

[Branches](#)

[Synchronizing with a CVS repository](#)

■ Related tasks

[Synchronizing with the repository](#)

[Merging from a branch](#)

■ Related reference

[CVS](#)

© Copyright IBM Corporation and others 2000, 2003.

Merging from a branch

After creating and working in a CVS branch for some time, you may want to merge your changes from the branch into another branch, or into HEAD. To do this, you will need to know:

1. The name of the branch or version that contains your changes.
2. The version from which the branch was created. This is the version name that you supplied when branching.

To merge your changes:

1. Ensure that the destination is loaded into your workspace. For example, if you wish to merge your changes into HEAD, make sure the project is shared with HEAD in your workspace. To do this, select the project and choose **Replace With > Another Branch or Version** from the context menu. Then select the branch to replace with.
2. Select the project and choose **Team > Merge**.
3. Enter the start point of the merge. This is the version from which the branch was created. Click **Next**.
4. Enter the branch or version which contains the changes you want to merge. Click **Finish**.
5. A Merge editor will open, showing all differences between your workspace and the branch with the changes.
6. Load all of the desired changes into the workspace. This may be accomplished by either manually merging changes and then choosing **Save** from the text editor's context menu, or by choosing **Update from Repository, Override and Update**, or **Merge Changes Into Local Copy** from the tree's context menu.
7. After all desired changes are in the workspace, choose **Team > Synchronize with Repository**. You may then commit all the changes to the repository.

Merge actions

The actions in the merge editor complement the manual merge toolbar actions that are available in the bottom half of the merge editor.

<i>Update from Repository</i>	This action is enabled on files that contain incoming changes. Running this action will bring the changes into the file in the workspace.
<i>Override and Update</i>	This action is enabled on files with conflicting changes. Running this action will discard any local changes you have and replace the file with the remote contents.
<i>Merge changes Into Local Copy</i>	This action is enabled on files with conflicting changes. Running this action will perform the equivalent of the "cvs update" command. This will merge the local contents with the remote contents using the base as the ancestor. Changes are merged regardless of the conflicting state of the files and can result in conflict markers (e.g. lines that can't be automatically merged) in the local file.

■ Related concepts

[Team programming with CVS](#)

[Branches](#)

[Synchronizing with a CVS repository](#)

■ Related tasks

[Branching](#)

[Synchronizing with the repository](#)

[Updating](#)

[Committing](#)

■ Related reference

[CVS](#)

[Merge wizard](#)

© Copyright IBM Corporation and others 2000, 2003.

Viewing a file's revision history

Every time you commit a file to the CVS repository, a new revision is created. Each revision is identified by a number (for example, 1.1 or 1.11.3.5). Along with each revision is stored the author, the date, and a commit comment.

To view the resource history for a file:

1. Select the file in the Navigator view. From the context menu, choose **Team > Show in Resource History**.
2. The CVS Resource History view will open and show a history for the selected file.

Note: The CVS Resource History view also shows you all version and branch tags that are associated with the file.

Tip: You can drag and drop a file from the Navigator view into the CVS Resource History view instead of choosing the menu item.

Tip: By clicking on column headings in the CVS Resource History view you can change the sorting order of the items.

■ Related concepts

[Team programming with CVS](#)

[Branches](#)

[Synchronizing with a CVS repository](#)

■ Related tasks

[Synchronizing with the repository](#)

■ Related reference

[CVS](#)

[CVS Resource History view](#)

© Copyright IBM Corporation and others 2000, 2003.

Versioning

Resources can be versioned in order to capture a snapshot of their current state at one specific point in time. See the Related tasks links for more details.

■ Related concepts

[Versions](#)

[Branches](#)

■ Related tasks

[Creating a version of a project](#)

[Enabling the CVS resource decorations](#)

[Versioning projects in the repository](#)

© Copyright IBM Corporation and others 2000, 2003.

Creating a version of a project

Prerequisite: The project that you want to version must first be in the workspace.

Tip: It is often preferable to synchronize resources with the CVS repository before versioning. This will ensure that there are no outstanding incoming or outgoing changes that might be accidentally excluded from the version.

To make a version of a project:

1. In the Navigator view, select the project that you want to version.
2. From the project's context menu, select **Team > Tag as Version**.
3. Enter the name of the version you wish to create.
4. Click **OK** to close the Tag Resources dialog and create the version.

Tip: You can browse existing versions of your project by clicking on the **Details** button in the Tag Resources dialog. Clicking **Refresh from Repository** should populate the Existing Versions list. It is sometimes helpful to see existing versions before naming your new version.

■ Related concepts

[Versions](#)

[Synchronizing with a CVS repository](#)

[Branches](#)

[Local history](#)

■ Related tasks

[Sharing a new project using CVS](#)

[Versioning projects in the repository](#)

[Synchronizing with the repository](#)

[Enabling the CVS resource decorations](#)

■ Related reference

[CVS](#)

© Copyright IBM Corporation and others 2000, 2003.

Versioning projects in the repository

You can version a project in the repository without adding the project to the Workbench.

1. Switch to the CVS Repository Exploring perspective or add the CVS Repositories view to the current perspective.
2. In the CVS Repositories view, expand a repository location and find the project you wish to version.
3. Right-click on the project and select **Tag as Version**.
4. Enter the name of the version you wish to create.
5. Click **OK** to close the dialog and create the version.

Tip: You can browse existing versions of your project by clicking on the **Details** button in the Tag Resources dialog. Clicking **Refresh from Repository** should populate the Existing Versions list. It is sometimes helpful to see existing versions before naming your new version.

Tip: If the version tag you wish to apply already exists on some of the resources to be versioned, you can perform a **Tag with Existing** instead. This operation will move an existing tag to the selected resources and can be used to move both version and branch tags. If the tag you wish to apply is not show in the list, you can click **Refresh from Repository**. If this doesn't find the tag, you can click **Configure Tags** which opens a dialog that allows you to search for tags on specific files in the repository.

■ Related concepts

[Branches](#)

■ Related tasks

[Creating a version of a project](#)

[Sharing a new project using CVS](#)

[Viewing a file's revision history](#)

[Branching](#)

■ Related reference

[CVS](#)

© Copyright IBM Corporation and others 2000, 2003.

Comparing resources with repository versions

To compare Workbench resources with those in the repository:

1. Select a resource in the Navigator view.
2. From the resource's pop-up menu, select one of the following menu items:
 - ◆ **Compare With > Latest From <Branch Name> or <Version>**– Compare the Workbench resource with the latest resources currently committed to the branch that the local project is shared with or if the local project is checked out as a version then compare against the version. A compare editor will be opened. This editor will display a tree of all resource differences between the workspace resource and the latest resources in the repository.
 - ◆ **Compare With > Another Branch or Version** – Compare the Workbench resource with a specific version or branch that you select in the repository. A compare editor will be opened. The editor will display a tree of all resource differences between the workspace resource and the resources in the specified version or branch.
 - ◆ **Compare With > Revision** – (available only on a file) Compare the Workbench file with another revision of the file. A compare editor will be opened. The editor will display a table of all available revisions of the selected file. Selecting one of the revisions will show the differences between the workspace revision and the selected revision.

■ Related concepts

[Three way comparisons](#)

[Synchronizing with a CVS repository](#)

■ Related tasks

[Synchronizing with the repository](#)

[Setting preferences for comparing files](#)

■ Related reference

[CVS](#)

[Compare editor](#)

© Copyright IBM Corporation and others 2000, 2003.

Working with patches

Patches allow developers to share work without storing it in a repository. This is helpful when a developer wants to contribute to a project that is shared through a repository but does not have write access to the repository. In this situation, the developer can create a patch and either e-mail it to a developer who does have write access or attach it to a bug in the bug reporting system used by the project, depending on the process defined by the project. A developer that does have write access can then apply the patch to the project and commit the changes.

To create a patch from a CVS project:

1. Select the resource that contains the modifications to be included in the patch. Although this can be any folder, it is easiest to select the project itself because the patch must be applied to the same resource it is generated from. The patch should also be applied to the same file revisions that it is generated on so steps should be taken to ensure that the patch is applied to the same resource line-up (the easiest way to do this is to create the patch on top of a version).
2. From the popup menu, select **Team > Create Patch...** The Create Patch wizard will open.
3. Choose where the patch should be saved:
 - a. *Save to Clipboard* – this will place the patch on the clipboard so it can be pasted into a text editor such as an e-mail program.
 - b. *Save to File System* – this will place the patch in the specified file in the local file system
 - c. *Save in Workspace* – this will place the patch in the specified file inside one of the existing workbench projects.

For small patches it may be reasonable to transfer the patch using the clipboard but in most cases the local file system is the best option to use. Click **Next** to configure how the patch is generated.
4. Choose how to configure the patch:
 - a. *Recurse into sub-folders* – If disabled, only the direct children of the selection are included in the patch. Otherwise, all descendants are included.
 - b. *Include new files in patch* – If disabled, only files that are under CVS version control are included. Otherwise, files that have been newly created but not added or ignored will also be included.
 - c. *Diff output format* – Allows the choice of several common diff output formats. *Unified* is the format used by many patch application tools including Eclipse.
5. Click **Finish**.
6. Transfer the patch as appropriate for the project being patched.

To apply a patch:

1. Select the resource that the patch was generated on. This resource should contain the same file revisions as the line-up on which the patch was generated.
2. From the pop-up menu, select **Team>Apply Patch...** The Resource Patcher wizard will open.
3. Indicate where the patch is to be found:
 - a. *File* – the patch is in a file on the local file system. Either type in the full path to the file or use the **Browse...** button to find the file.
 - b. *Clipboard* – the patch is on the clipboard. **Warning:** It is safer to use a file based patch. Line endings may not be handled properly if the clipboard is used and the patch was generated on a different platform (i.e. Linux vs. Windows).

Click **Next** to see the effect of applying the patch.

Running Eclipse

4. The top pane of this page shows whether the patch could be successfully applied to files in your workspace. If you select a leaf item in the tree the bottom pane shows the part of the patch file (known as 'hunk' in patch terminology) in an easy to read side by side presentation. **Note:** The bottom pane **does not** show a preview of how resources in your workspace would look like after applying the hunk. It just shows the contents of the patch file.
 - a. A checked item indicates that a patch (or hunk) could be successfully applied to a workspace resource. You can exclude patches or individual hunks by unchecking them.
 - b. A red exclamation mark indicates that there is a problem with a patch or hunk. This happens if the patch is not well formed or the revision of one or more files that the patch were generated on do not match the revisions that the patch is being applied to. You find the reason for the failure in parenthesis.

In order to apply the full patch successfully you will have to eliminate the problems (red exclamation marks) and get checked items everywhere by tweaking the options on this wizard page (see 'Options' below).

5. If all is well, click **Finish** to apply the patch. The workspace will now contain outgoing changes for each file modified by the patch.

Options for applying a patch

For getting successful matches of your patch file you have the following options:

1. Go back to the first page of the Resource Patcher wizard and select the correct resource to which the patch should be applied.
2. If a common prefix of the path names stored in the patch file doesn't match the path names in your current workspace, you can 'Ignore leading path name segments'.
3. Use the 'Ignore whitespace' option to make the matching process independent from whitespace differences between the patch file and files in your workspace.
4. Adjust the 'Maximum fuzz factor' (patch terminology). This factor determines how far from its original line a hunk is allowed to match. The default is two. So if a hunk does not match at the line given in the patch file, the Resource Patcher tries to match the hunk 'fuzz' number of lines before or after the position.
5. Use the 'Reverse patch' option for patch files that already have been applied to your workspace. This option is also useful to undo or redo a patch.

■ Related concepts

[Team programming with CVS](#)

■ Related tasks

[Comparing resources](#)

■ Related reference

[CVS](#)

Replacing resources in the Workbench

To replace Workbench resources with versions in the repository:

1. Select a resource in the Navigator view.
2. From the resource's pop-up menu, select one of the following menu items:
 - ◆ **Replace With > Latest From <Branch Name>** or **<Version>** – replace the Workbench resource with the latest resources currently committed to the branch that the local project is shared with or if the local project is checked out as a version then replace with the same version.
 - ◆ **Replace With > Another Branch or Version** – replace the Workbench resource with a specific version or branch that you select in the repository. When you select this option, another window opens, so that you can browse through the branch and version tags in the repository and select the one that you want.
 - ◆ **Replace With > Revision** – (available only on a file) replace the Workbench file with another revision of the file. When you select this option, a dialog opens to allow you to select a revision from the list of available ones. A test window displays the comparison between the local revision and the selected remote revision.

■ Related concepts

[Team programming with CVS](#)

[Branches](#)

[Versions](#)

[Synchronizing with a CVS repository](#)

■ Related tasks

[Checking out a project from a CVS repository](#)

[Sharing a new project using CVS](#)

[Viewing a file's revision history](#)

[Branching](#)

[Synchronizing with the repository](#)

© Copyright IBM Corporation and others 2000, 2003.

Finding out who's working on what: watch/edit

CVS provides a notification scheme which allows you to know if someone is modifying a file that you care about. This facility is known as *watches*. By setting a *watch* on a file, you can have CVS notify you via email (or other) if someone else starts to *edit* this file.

There are two parts to CVS watches: *watch*, and *edit*. The first, *watch*, is how you specify which files you wish to be notified about. The second, *edit*, is how you inform the CVS server (and thus others) that you are about to modify a file.

Edit is useful on its own without ever setting up any watches and lots of people work this way. This is because when you edit a file, you will be told immediately if someone else is already editing that file. Since most people just want to know up front that they may have to merge their changes on commit, *edit* on its own is sufficient for most. Another advantage to using just *edit* is that it doesn't require any administrative changes to the server, where as *watch* does. All that *watches* gives above this is the email notification that some file you are watching is being modified.

For these reasons, *edit* is supported natively by Team CVS where as *watch* isn't.

Setting up Watches

As mentioned, you can't set watches in Team CVS. If you are interested in doing this, you should consult your cvs documentation. In brief though, this is what's involved:

1. First, you or your CVS administrator will need to modify the CVSROOT/notify file. Consult the CVS documentation on watches for details on how to configure this file.
2. Next, you will need to perform a command line "cvs watch add <filename>" for each file that you wish to watch. If <filename> is a directory name, then all files within that directory will be watched.

Setting up a Project for Watch/Edit

Watches and editing are optional in CVS. To use this facility, you must turn on this option in the **Team > CVS > Watch/Edit** preferences page. Select "Configure projects to use Watch/Edit on checkout", accept the preference dialog, and then checkout your project. All the files in the project will be checked out read-only. This tells the CVS client which files are being edited by you and which aren't (writeable files are being edited). If you've already checked out the project before you turned on this option, you can either check it out again or enable the "Use Watch/Edit for this project" option on the project's CVS properties page. Either of these operations will make the files in the project read-only.

Editing

Although typical CVS clients require you to perform an explicit edit, Team CVS automatically issues an edit as soon as you start to modify a file. This support is built deep into Eclipse, so typing in a text editor, performing Java refactoring, etc., will all issue a CVS edit for you. You can also perform an explicit edit via the **Team > Edit** context menu on a resource.

When an *edit* is issued, you will be informed immediately if someone is already editing that file. In addition, everyone who is *watching* that file will be notified by the CVS server via email etc. Since watches simply give you email notification, *edit* without ever setting up watch lists is still a useful (and popular) workflow.

Running Eclipse

If you prefer, you can turn off automatic issuing of edits. This means you will need to manually perform a **Team > Edit** for each file you are working on. To use this work mode, turn on the **Team > CVS > Watch/Edit** preference "Edit the file without informing the server".

Finally, you can see the list of editors of a file at any time by selecting **Team > Show Editors** from the context menu of that file.

Unediting

Just as you can tell CVS that you are editing a file, there also needs to be a way of telling CVS that you are no longer editing that file. This is referred to as *unedit*. This way, if someone checks the editors list for a file, they'll know if someone is still working on that file. This happens in one of two ways:

- When you commit an edited file, an unedit is automatically issued.
- If you discover that you want to back out of the changes to a file, you can explicit unedit the file. In addition to notifying the server, an explicit unedit will revert the file to its base (ie. the workspace will then contain the copy of the file before you started modifying it).

■ Related concepts

[Watch/Edit](#)

[Team programming with CVS](#)

■ Related reference

[CVS](#)

© Copyright IBM Corporation and others 2000, 2003.

Changing CVS team settings

You can customize the settings related to a number of CVS team views and operations.

1. From the main menu bar, select **Window > Preferences**.
2. On the left side of the Preferences window, select the **Team** category.
3. You will see a number of general **Team** options. These options can be used by any Eclipse integrated repository. In our case, they are all applicable to CVS.
4. You will also see a **CVS** category. Select it. You will see various CVS options. If you expand the **CVS** category, you will see additional CVS preference pages.

Here are some examples of settings that you can change in the **CVS** section:

- To filter out receiving empty directories when updating or synchronizing, select the **Prune empty directories** toggle.
- To automatically save the contents of modified editors prior to synchronize, commit, etc., choose **Auto-save**.

■ Related concepts

[Team programming with CVS](#)

■ Related tasks

[Synchronizing with the repository](#)

[Committing](#)

■ Related reference

[CVS Preference pages](#)

© Copyright IBM Corporation and others 2000, 2003.

Restoring deleted files from the repository

The CVS plug-in allows you to delete files from the CVS repository. If you delete a managed file and commit the deletion to the server, the file is deleted from the current branch or version. Although the file has been deleted from the current branch, the file and its previous revisions are actually still on the server. The CVS plug-in provides a tool to help restore a file to your workspace that has previously been deleted from the CVS repository:

1. Select a folder managed by CVS and from the context menu select **Team > Restore from Repository**.
2. The repository will be searched for deleted files and the list will be shown in a dialog.
3. Select a file in the left most pane to display the revisions of the file that are available in the repository.
4. Select a revision in the right-most pane to view the contents in the bottom text pane.
5. Check the revision that you would like to restore.
6. Click **Finish** once you have checked off a revision for each file you wish to restore.
7. After restoring the file if you want to actually restore the file to the current branch you have to add then commit the file back into the repository. If the filename does not change, revision history will be maintained for the restored file.

■ Related concepts

[Team programming with CVS](#)

■ Related tasks

[Synchronizing with the repository](#)

[Committing](#)

© Copyright IBM Corporation and others 2000, 2003.

Reverting a branch to a previous version

It is often useful to revert the contents of a branch to those of a specific version. For example, if your current branch contains changes that you no longer want to release you can revert all or a portion of a project to the contents of any version.

1. Checkout into your workspace the contents from the branch that you want to revert.
2. Select **Compare With > Another Branch or Version** on the resource(s) that you want to revert.
3. From the tag selection dialog box select the version to which you want to revert the branch.
4. When the compare editor opens, review the differences that are shown and ensure that they are what you expected.

Tip: If several files report as false differences (e.g. the files are shown but there are no actual changes in the files) select the  **Use File Contents** toolbar button to recompute the comparison using file contents instead of revision numbers.

5. Select the root folder in the compare view and from the context menu select **Copy All Changes From Right To Left**. After the operation is completed the folder or project you compared against will have exactly the same contents as the remote revision.
6. You can verify this by performing another comparison against the version. However this time, when the comparison completes select the  **Use File Contents** toolbar button to redo the comparison using file contents instead of revision numbers. *Note:* you may also have to select the **Ignore White Space** button too.
7. After the comparison using file contents completes the compare editor should be empty. Meaning that the files in your workspace are identical to the ones in the selected version.

Once your workspace contains the new contents, run your tests then commit the changes to the branch.

■ Related concepts

[Team programming with CVS](#)

■ Related tasks

[Synchronizing with the repository](#)

[Committing](#)

© Copyright IBM Corporation and others 2000, 2003.

Moving version tags

Warning: Although many people would prefer a CVS version to be frozen in time and not be modifiable, version and branch tags in CVS are mutable. As a result many are convinced that modifying a version is bad practice, but there are a couple of scenarios where it actually comes in handy. With that said, please be cautious when moving tags around.

Moving a tag on a single file

Let's say that you have just submitted your build by versioning your project as R1. But you soon after discover that there is a small change to a file that should be made and included in the build. Instead of having to re-version the project you can move the R1 version tag for the modified file.

1. Modify the file(s) Select the file that was modified after R1 was created and from the context menu select *Team > Show in Resource History*.
2. From within the Resource History view select the revision that should be marked with the R1 version.
3. From the context menu select *Tag with Existing...*
4. Select the R1 version from the dialog box and press OK.
5. The resource history view will be updated to confirm that the version had been moved.

Moving a tag from within the repositories view

Many projects use a well defined version name for their current stable lineup in HEAD. For example, by versioning HEAD with the STABLE tag the build scripts could simply checkout the STABLE version for builds. As the code evolves the STABLE tag is moved regularly to mark the most current stable lineup. The repositories view provides an action for moving an existing tag.

1. Open the Repositories view and select a resource.
2. From the context menu select **Tag with Existing**
3. A tag selection dialog will appear from which you can select the tag to move. If the tag you wish to apply is not show in the list, you can click **Refresh from Repository**. If this doesn't find the tag, you can click **Configure Tags** which opens a dialog that allows you to search for tags on specific files in the repository.
4. Then press **OK** and your tag will be moved. The operation will move an existing tag to the selected resources and can be used to move both version and branch tags.

■ Related concepts

[Team programming with CVS](#)

■ Related tasks

[Synchronizing with the repository](#)
[Committing](#)

© Copyright IBM Corporation and others 2000, 2003.

Running the CVS command-line client outside of Eclipse

Compatibility

Because the Eclipse CVS plug-in stores its meta information in a format that is compatible with the command-line CVS client you should be able to use a CVS command line client against Eclipse workspace files on disk. The metadata is stored in CVS/ sub-directories but you rarely see them within Eclipse. They are marked as private which causes them to be hidden from view. If you open a (non-Eclipse) file explorer you will see that these directories and their contents appear on the file system.

Don't forget to refresh!

Whenever you use external tools to modify workspace files, you must perform a **Refresh** from within Eclipse to make the workspace aware of the changes. If you get a *resource out of sync* error in Eclipse it is a sign that there are resources in Eclipse that have been modified outside of Eclipse. One solution is to perform a refresh (available from a resource's popup menu) on any resources or projects that were modified outside of Eclipse. There is also an *auto-refresh* plug-in available from [Platform Core](#) web site.

Caveats

1. Deleted folders

You may encounter unexpected behavior when using the command-line CVS client in conjunction with deleted folders. Eclipse's CVS support keeps track of deleted folders and their contents so that, on the next synchronization, the Synchronize view can properly report on the changes. This information is kept outside of the CVS meta folder structure. This is because in CVS you normally inform the repository of deletions prior to deleting them locally, which is a different workflow than we like to support in the Synchronization view. Thus it is recommended that you do not use the command-line CVS client while you have pending deletions to commit. In some circumstances it could cause the Synchronize view to display incorrect contents, although it will not cause any lost work.

2. CVS directories appear in the workbench

When you use the command-line CVS the CVS folders can sometimes appear in the Navigator View. There are some cases where CVS folders are not hidden from the UI as the user would expect. For instance, CVS folders will appear if a user imports a CVS project into Eclipse before the CVS plug-in is loaded. To avoid this, open the CVS Repositories view (thus loading the CVS plug-in) before importing CVS projects into Eclipse.

3. The 'extssh' connection method

The *extssh* connection method is unique to Eclipse CVS and doesn't exist in the command-line CVS client. If you are using *extssh* to connect to a SSH1 repository and would like to use the command line client for some CVS operations you have to create a new location in Eclipse that uses a compatible connection method:

1. Create a new repository location that uses a CVS command line client compatible connection method (e.g. *pserver* or *ext*). The repository path, host, and user should be identical to those in the *extssh* location.

Running Eclipse

2. Select the project that is shared with the *extssh* location and open the CVS properties page. This is the project that you wanted to run external CVS commands on.
3. Select the ***Change Sharing...*** button and select the new location that uses either the *ext* or *pserver*.
4. Next, you can run command line CVS operations and then return to Eclipse. *Don't forget to refresh!*

■ Related tasks

[Changing the properties of a CVS repository location](#)

[Creating a CVS repository location](#)

[Using projects checked out with another CVS tool](#)

© Copyright IBM Corporation and others 2000, 2003.

Updating features with the update manager

To check to see whether there are updates for a product's existing features (requires Internet access):

1. Click **Help > Software Updates > New Updates**. This will contact the Web sites associated with the product's features to discover what versions of those features are available. The potential upgrades are presented in an Available Updates page.
2. Select the feature versions that you wish to upgrade, and click **Next**.
3. Carefully review the license agreements for the upgraded features. If the terms of all these licenses are acceptable, check "I accept the terms in the license agreements." Do not proceed to download the features if the license terms are not acceptable.
4. Feature versions can be digitally signed by the company that provides them. This allows the user to verify more easily that the features and plug-ins that are about to be downloaded and installed are coming from a trusted supplier.

Warning: Because of the possibility of harmful or even malicious plug-ins, you should only download features from parties that you trust.

Click **Install** to allow the downloading and installing to proceed.

5. Once all the features and plug-ins have been downloaded successfully and their files installed into the product on the local computer, a new configuration that incorporates these features and plug-ins will be formulated. Click **Yes** when asked to exit and restart the Workbench for the changes to take effect.

■ Related concepts

[Features](#)

■ Related tasks

[Inspecting the current configuration](#)

[Installing new features with the update manager](#)

[Installing several features at once using group updates](#)

[Picking up pending changes](#)

[Searching for features](#)

[Saving a configuration](#)

[Restoring a saved configuration](#)

© Copyright IBM Corporation and others 2000, 2003.

Inspecting the current configuration

The About Eclipse Platform dialog (**Help > About Eclipse Platform**) shows most of the important information about the features and plug-in configured in the product. To find out further detailed information about the current configuration:

1. Open the main update manager by clicking **Help > Software Updates > Update Manager**. This opens the Install/Update perspective.
2. In the Install Configuration view, expand the Current Configuration item. The first tier of items are locations on the local computer where the files for features and plug-ins are stored. For a typical product, all features are installed in a single directory sub-tree. However, a feature that is installed as an extension of another product are usually located in a separate directory sub-tree.
3. Expand the location to see the feature versions installed there.
4. The nesting of the features indicates which child features are included in a given parent feature. Nested features act as a unit.
5. Select a feature version and click **Show Details** to open the Preview view which presents more detailed information about the selected feature.
6. Click **Show Disabled Features** icon (top bar of the Install Configuration view). Any feature versions that are disabled are now shown (you might not have any). Disabled feature versions are ones that are installed on the local computer but excluded from the current configuration. At most one version of any given feature can be present in a given configuration; usually the ones with the lower version numbers will be in a disabled state.
7. Select a feature version and click **Show Status** to find out whether the selected feature is configured exactly as packaged. The status will show when a feature's plug-ins are present at different versions, or disabled due to missing prerequisite plug-ins that are expected to be supplied by some other feature.

■ Related concepts

[Features](#)

■ Related tasks

[Installing new features with the update manager](#)

[Updating features with the update manager](#)

[Saving a configuration](#)

© Copyright IBM Corporation and others 2000, 2003.

Saving a configuration

For added safety when upgrading the product or installing new features, save a snapshot of the current configuration beforehand. This makes it easy to return to a known configuration if the new or updated features prove unsatisfactory, and creates a permanent record of configurations that you have been using. To save the current configuration:

1. Open the main update manager by clicking **Help > Software Updates > Update Manager**. This opens the Install/Update perspective.
2. In the Install Configuration view, expand the Configuration History item and select the most recent entry (the top one). Click **Save** on the context menu.
3. Expand the Saved Configurations item in the same view to see the entry for the newly saved configuration. Expanding the saved configuration shows the locations of the features.
4. To change the name of this saved configuration to something recognizable, select the saved configuration and click **Properties** on the context menu.
5. In the property dialog that opens, change the Configuration Name of the Preserved Configuration property to something more appropriate, like "Stable".
6. Click **OK** to have the name change take effect.

Unlike the entries in the Configuration History (a queue of limited length), saved configuration entries are retained indefinitely and go away only when explicitly removed (**Remove** on the saved configuration's context menu).

■ Related concepts

[Features](#)

■ Related tasks

[Inspecting the current configuration](#)

[Restoring a saved configuration](#)

© Copyright IBM Corporation and others 2000, 2003.

Restoring a saved configuration

A previously saved configuration (or any configuration recorded in the history) can be restored, thereby backing out of the results of an unsuccessful upgrade. Restoring a saved configuration does not delete the files for feature and plug-in versions from the local computer; it merely ignores them and acts as it did when they were not present. To restore a saved configuration to be the current configuration:

1. Open the main update manager by clicking **Help > Software Updates > Update Manager**. This opens the Install/Update perspective.
2. In the Install Configuration view, expand the Saved Configurations (or Configuration History) item and select the desired configuration. Click **Restore**.
3. Click **Yes** when asked to exit and restart the Workbench for the changes to take effect.
4. Expand the Current Configuration (and children) to confirm that the current configuration now matches the configuration saved earlier.

■ Related concepts

[Features](#)

■ Related tasks

[Inspecting the current configuration](#)

[Saving a configuration](#)

© Copyright IBM Corporation and others 2000, 2003.

Picking up pending changes

Whenever the Eclipse platform is started it detects and notes any changes that have been made to the installed base of files such as the appearance (or disappearance) of feature and plug-in versions. New or updated features are not automatically configured for an existing workspace; rather, you are told on start up that there are updates and given the opportunity to configure them. If you accept them, the new and updated feature versions become a part of the feature and plug-in configuration used for that workspace. If not accepted, the new and updated feature versions remain dormant (disabled) for that workspace, but available to be picked up at a later time.

To pick up pending changes:

1. When new changes are detected, Configuration Changes page will open on startup. Alternatively, click **Help > Software Updates > Pending Changes...** The Configuration Changes page opens showing a list of detected changes.
2. Select and expand detected change items to see the associated feature versions. You can accept the entire change items, or select individual features. Individual selection may be necessary in case when some of the new features have errors.
3. Check items that are to be included in the configuration. Unchecked items are ignored (but remain pending).
4. Click **Finish** to update the configuration, or **Cancel** to retain the current configuration (pending changes remain pending). Note that if a pending item has been partially selected (not all the features), pressing **Finish** will mark the item processed and it will be removed from the list of pending changes.
5. Click **Yes** when asked to exit and restart the workbench for the changes to take effect.

■ Related concepts

[Features](#)

■ Related tasks

[Updating features with the update manager](#)

[Restoring a saved configuration](#)

© Copyright IBM Corporation and others 2000, 2003.

Installing new features with the update manager

To locate and install a new feature into a product (ordinarily requires web access):

1. Open the main update manager by clicking **Help > Software Updates > Update Manager**. This opens the Install/Update perspective.
2. Create a bookmark for a web site where Eclipse features and plug-ins are published. In the Feature Updates view, select **New > Site Bookmark** from the context menu.
3. In the New Site Bookmark dialog that opens, give the site a name such as "CompanyA" and enter the URL such as "http://companyA.example.com/eclipseupdates".
4. Expand the newly created item in Feature Updates view to show the categories of feature versions available at that update site. This will contact the web site to discover what features are available.
5. Expand one of the top-level categories and select a feature. Information about that feature appears in the Preview view.
6. After selecting the feature to be added, click the **Install** button within the Preview view.
7. The Feature Install wizard confirms the details of the feature you are about to install. Click **Next**.
8. Carefully review the license agreements for the upgraded features. If the terms of all these licenses are acceptable, check "I accept the terms in the license agreements". Do not proceed to download the features if the license terms are not acceptable.
9. If a feature selected for install include optional features, a page will show up allowing you to select whether you want them installed or not. Optional features typically carry functionality that is not essential for proper functioning of the main feature.
10. The Install Location page controls where the new feature's files are to be installed on the local computer. Select the directory into which the product is installed and hit **Next**. (If the product is installed in a directory to which you do not have write access, you should contact your system administrator and get them to install this feature so that it will be generally available. The other option is to click **Add** and point to a directory to which you do have write access; however, features installed there are only available when using the current workspace.)
11. Feature versions can be digitally signed by the company that provides them. This allows you to verify more easily that the features and plug-ins that are about to be downloaded and installed are coming from a trusted supplier.
Warning: Because of the possibility of harmful or even malicious plug-ins, you should only download features from parties that you trust.

Click **Install** to allow the downloading and installing to proceed.

12. Once the new feature and plug-ins have been downloaded successfully and their files installed into the product on the local computer, a new configuration that incorporates these features and plug-ins will be formulated. Click **Yes** when asked to exit and restart the workbench for the changes to take effect. To add other new features at the same time before restarting, click **No** and repeat.

■ Related concepts

[Features](#)

■ Related tasks

[Inspecting the current configuration](#)

[Updating features with the update manager](#)

[Installing several features at once using group updates](#)

[Searching for features](#)

[Saving a configuration](#)

Restoring a saved configuration

© Copyright IBM Corporation and others 2000, 2003.

Installing several features at once using group updates

To install several features at once using update manager (ordinarily requires web access):

1. Open the main update manager by clicking **Help > Software Updates > Update Manager**. This opens the Install/Update perspective.
2. Create a bookmark for a web site where Eclipse features and plug-ins are published. In the Feature Updates view, select **New > Site Bookmark** from the context menu.
3. In the New Site Bookmark dialog that opens, give the site a name such as "CompanyA" and enter the URL such as "http://companyA.example.com/eclipseupdates".
4. Expand the newly created item in Feature Updates view to show the categories of feature versions available at that update site. This will contact the web site to discover what features are available.
5. Expand one of the top-level categories and select a feature. Information about that feature appears in the Preview view.
6. After selecting the feature to be added, locate section in the Preview view with a title **Group Updates** and check the "Install by adding to the Selected Updates" checkbox.
7. Repeat for every feature you want to install. At any point, you can inspect the selection by pressing "Selected Updates" hyperlink to the right of the checkbox, or by bringing the Selected Updates view to the top (by default, it is stacked with the Install Configuration view).
8. When all the features that are to be installed are selected, switch to the Selected Updates view, bring up the pop-up menu and select **Process All** item. If you change your mind about any feature, you can remove it from the list prior to processing.
9. The Feature Install wizard confirms the details of the features you are about to install. Note that there may be conflicts between features you have selected. You will be informed about them and asked to deselect the ones causing the conflict. Click **Next**.
10. Carefully review the license agreements for the upgraded features. If the terms of all these licenses are acceptable, check "I accept the terms in the license agreements". Do not proceed to download the features if the license terms are not acceptable.
11. If one or more features selected for install include optional features, a page will show up allowing you to select whether you want them installed or not. Optional features typically carry functionality that is not essential for proper functioning of the main feature.
12. The Install Location page controls where the new feature's files are to be installed on the local computer. Select the directory into which the product is installed and hit **Next**. (If the product is installed in a directory to which you do not have write access, you should contact your system administrator and get them to install this feature so that it will be generally available. The other option is to click **Add** and point to a directory to which you do have write access; however, features installed there are only available when using the current workspace.)
13. Feature versions can be digitally signed by the company that provides them. This allows you to verify more easily that the features and plug-ins that are about to be downloaded and installed are coming from a trusted supplier.
Warning: Because of the possibility of harmful or even malicious plug-ins, you should only download features from parties that you trust.

Click **Install** to allow the downloading and installing to proceed.

14. Once the new feature and plug-ins have been downloaded successfully and their files installed into the product on the local computer, a new configuration that incorporates these features and plug-ins will be formulated. Click **Yes** when asked to exit and restart the workbench for the changes to take effect. To add other new features at the same time before restarting, click **No** and repeat.

■ **Related concepts**

Features

■ **Related tasks**

Inspecting the current configuration

Updating features with the update manager

Installing new features with the update manager

Searching for features

Saving a configuration

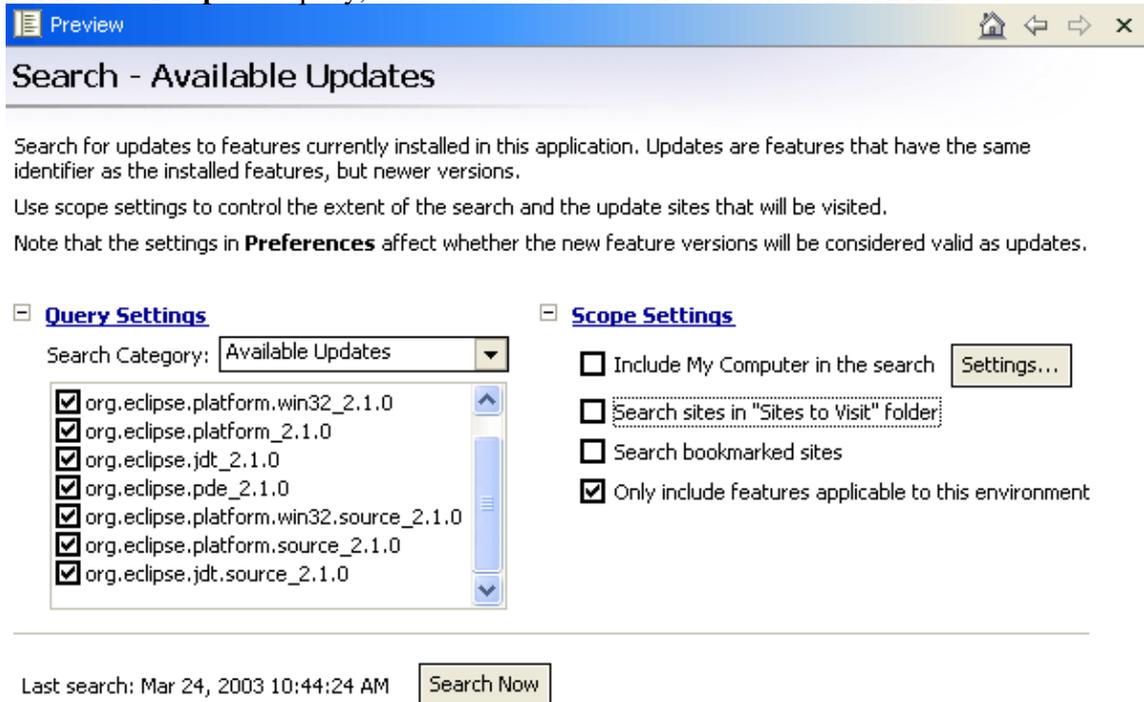
Restoring a saved configuration

© Copyright IBM Corporation and others 2000, 2003.

Searching for features

To search for features by running queries (ordinarily requires web access):

1. Open the main update manager by clicking **Help > Software Updates > Update Manager**. This opens the update manager perspective.
2. In the Feature Updates view, select the built-in **Available Updates** query, or create a new query object using the view's context menu **New > Search**.
3. If creating a new query, in the New Search dialog that opens, give the query a name, select the query category, and optionally select a folder that is to contain the query. The selected query category determines what query settings can be specified. The following categories are supported:
 - ◆ **Available Updates** allows search for available updates to selected features.
 - ◆ **Regular Expression** allows search for features using expression matching.
 - ◆ **Plug-ins in Features** allows search for features containing specified plug-ins.
4. Select the query in the Feature Updates view.
5. The Preview shows the query details. Expand the **Query Settings**.
6. For **Available Updates** query, select one or more installed feature to search for.



7. For **Regular Expression** query, specify the values you want to search for.

Running Eclipse

The screenshot shows the 'Search - Regular Search' dialog box in Eclipse. The title bar includes a 'Preview' icon and standard window controls. The main title is 'Search - Regular Search'. Below the title, there is a brief instruction: 'Search for features that contain desired text or regular expression. Use the query settings to configure search options.' The dialog is divided into two main sections: 'Query Settings' and 'Scope Settings'. Under 'Query Settings', there is a 'Search Category' dropdown set to 'Regular Expressions', an 'Expression' text field containing 'test', a 'Case sensitive' checkbox (unchecked), and a 'Look for expression in:' section with three checked options: 'Feature name', 'Feature provider', and 'Feature description'. Under 'Scope Settings', there are four checkboxes: 'Include My Computer in the search' (unchecked), 'Search sites in "Sites to Visit" folder' (unchecked), 'Search bookmarked sites' (unchecked), and 'Only include features applicable to this environment' (checked). A 'Settings...' button is located next to the first checkbox. At the bottom, there is a 'Last search:' timestamp 'Mar 24, 2003 10:44:24 AM' and a 'Search Now' button.

8. For **Plug-ins in features** query, use the Add or Delete buttons to specify the plug-ins to search for. When adding plug-ins to the list enter the plug-in identifier and version into the Add dialog.

The screenshot shows the 'Search - Plugins Search' dialog box in Eclipse. The title bar includes a 'Preview' icon and standard window controls. The main title is 'Search - Plugins Search'. Below the title, there is a brief instruction: 'Look for plug-ins that are packaged in the features. This search category is useful when required plug-ins of the feature you wanted to install were not found in this application.' The dialog is divided into two main sections: 'Query Settings' and 'Scope Settings'. Under 'Query Settings', there is a 'Search Category' dropdown set to 'Plug-ins in Features', a text area containing 'org.eclipse.test_1.0.0', and two buttons: 'Add...' and 'Delete'. Under 'Scope Settings', there are four checkboxes: 'Include My Computer in the search' (unchecked), 'Search sites in "Sites to Visit" folder' (unchecked), 'Search bookmarked sites' (unchecked), and 'Only include features applicable to this environment' (checked). A 'Settings...' button is located next to the first checkbox. At the bottom, there is a 'Last search:' timestamp 'Mar 24, 2003 10:44:24 AM' and a 'Search Now' button.

9. Expand the **Scope Settings** and select the targets to search.
- ◆ **Include My Computer in search** will cause the selected local roots to be searched for local update sites containing features
 - ◆ **Search sites in "Sites to Visit" folder** will cause the preconfigured sites to be searched
 - ◆ **Search bookmarked sites** will cause bookmarked sites to be searched
 - ◆ **Only include features applicable to this environment** will only consider features that match

Running Eclipse

the current runtime environment to be included in the results

- ◆ Note, that Available Updates queries always search the update sites that are specified as part of the feature definitions
10. Click **Search Now** to trigger the search. The search runs asynchronously, so the workbench UI remain responsive. Search results are accumulated in the Preview view, as well as being shown as children of the query object in the Feature Updates view.
 11. To install one of the located features, select the feature in the search result list, or the Feature Updates view, and click the Install or Update button.

■ Related concepts

[Features](#)

■ Related tasks

[Installing new features with the update manager](#)

[Updating features with the update manager](#)

© Copyright IBM Corporation and others 2000, 2003.