

## **Crash recovery**

# Table of Contents

<b><u>Crash recovery</u></b> .....	<b>1</b>
<b><u>Preferences</u></b> .....	<b>2</b>
<b><u>Workbench</u></b> .....	<b>3</b>
<b><u>Appearance</u></b> .....	<b>5</b>
<b><u>Compare/Patch</u></b> .....	<b>6</b>
<u>General options</u> .....	8
<u>Text Compare options</u> .....	10
<b><u>Editors</u></b> .....	<b>10</b>
<b><u>Text Editor</u></b> .....	<b>10</b>
<u>Appearance options</u> .....	12
<u>Annotations options</u> .....	13
<b><u>External Tools</u></b> .....	<b>13</b>
<b><u>File Associations</u></b> .....	<b>13</b>
<u>File types list</u> .....	16
<u>Associated editors list</u> .....	17
<b><u>Fonts</u></b> .....	<b>17</b>
<b><u>Keys</u></b> .....	<b>17</b>
<u>Key strokes, Key sequences, and Key bindings</u> .....	17
<u>Configurations</u> .....	18
<u>Scopes</u> .....	18
<u>Platform and Locale</u> .....	22
<u>Customizing Key bindings</u> .....	22
<u>The Dynamic Nature of Key bindings</u> .....	25
<u>Conflict Resolution</u> .....	26
<u>Available Perspectives Options:</u> .....	28
<b><u>Label Decorations</u></b> .....	<b>29</b>
<b><u>Linked Resources</u></b> .....	<b>29</b>
<b><u>Local History</u></b> .....	<b>31</b>
<b><u>Perspectives</u></b> .....	<b>32</b>
<b><u>Search</u></b> .....	<b>33</b>

# Table of Contents

<b><u>Startup</u></b> .....	<b>35</b>
<b><u>Ant</u></b> .....	<b>35</b>
<b><u>Ant Editor</u></b> .....	<b>35</b>
<u>Appearance options</u> .....	38
<u>Annotations options</u> .....	42
<b><u>Ant Runtime</u></b> .....	<b>44</b>
<b><u>Build Order</u></b> .....	<b>46</b>
<b><u>Help</u></b> .....	<b>47</b>
<b><u>Help Server</u></b> .....	<b>49</b>
<b><u>Install/Update</u></b> .....	<b>50</b>
<b><u>Web-triggered Updates</u></b> .....	<b>51</b>
<b><u>Team</u></b> .....	<b>51</b>
<b><u>CVS</u></b> .....	<b>52</b>
<u>General CVS Preferences:</u> .....	52
<u>Console preferences:</u> .....	52
<u>Ext Connection Method preferences:</u> .....	53
<u>Label Decorations preferences:</u> .....	54
<u>Watch/Edit preferences:</u> .....	56
<b><u>Ignored Resources</u></b> .....	<b>57</b>
<b><u>File Content</u></b> .....	<b>58</b>
<b><u>CVS</u></b> .....	<b>58</b>
<b><u>CVS Repositories view</u></b> .....	<b>58</b>
<u>Titlebar</u> .....	58
<u>Go Home</u> .....	58
<u>Go Back</u> .....	58
<u>Go Forward</u> .....	58
<u>Refresh View</u> .....	59
<u>Collapse All</u> .....	59
<u>Drop-Down Menu</u> .....	59
<u>Context menu</u> .....	59
<u>New &gt; Repository Location</u> .....	59
<u>Configure Branches and Versions</u> .....	59
<u>Refresh Branches and Versions</u> .....	59
<u>Add to Branch List</u> .....	59

# Table of Contents

<b><u>CVS Repositories view</u></b>	
<u>Open</u> .....	59
<u>Show in Resource History</u> .....	59
<u>Compare</u> .....	60
<u>Compare With</u> .....	60
<u>Checkout as Project</u> .....	60
<u>Checkout as</u> .....	60
<u>Checkout Into</u> .....	60
<u>Checkout Out Module</u> .....	60
<u>Tag as Version</u> .....	61
<u>Tag with Existing</u> .....	61
<b><u>Synchronize view</u></b> .....	<b>62</b>
<u>Modes</u> .....	62
<u>Synchronization state</u> .....	63
<u>Structure compare pane</u> .....	64
<u>Text compare pane</u> .....	64
<b><u>CVS Resource History view</u></b> .....	<b>64</b>
<u>Revision</u> .....	64
<u>Tags</u> .....	64
<u>Date</u> .....	64
<u>Author</u> .....	66
<u>Comment</u> .....	67
<b><u>CVS Console view</u></b> .....	<b>68</b>
<b><u>New CVS Repository Location wizard</u></b> .....	<b>69</b>
<b><u>CVS repository location</u></b> .....	<b>69</b>
<b><u>Merge wizard</u></b> .....	<b>70</b>
<u>Start Point</u> .....	73
<u>End state</u> .....	73
<b><u>Ignoring resources from version control</u></b> .....	<b>73</b>
<u>Why ignore files when synchronizing?</u> .....	73
<u>Global ignore facility</u> .....	75
<u>CVS ignore facility</u> .....	76
<b><u>CVS Label Decorations</u></b> .....	<b>76</b>
<b><u>Workbench toolbar</u></b> .....	<b>76</b>
<u>New Wizard</u> .....	76
<u>Save The Open Editor Contents</u> .....	76
<u>Save To Another Location</u> .....	76
<u>Print</u> .....	76
<u>Search</u> .....	77

# Table of Contents

<b><u>Workbench toolbar</u></b>	
<u>External Tools</u> .....	77
<b><u>Shortcut bar</u></b> .....	<b>77</b>
<u>Open Perspective</u> .....	77
<u>Perspective Buttons</u> .....	78
<u>View Buttons</u> .....	78
<b><u>View toolbars</u></b> .....	<b>78</b>
<u>Close View</u> .....	79
<u>Title Bar</u> .....	79
<b><u>Builds</u></b> .....	<b>79</b>
<u>Auto-build vs. Manual Build</u> .....	80
<u>Full vs. Incremental Builds</u> .....	81
<b><u>Perspectives</u></b> .....	<b>82</b>
<b><u>Local history</u></b> .....	<b>82</b>
<b><u>Keys</u></b> .....	<b>82</b>
<u>Key strokes, Key sequences, and Key bindings</u> .....	82
<u>Configurations</u> .....	83
<u>Scopes</u> .....	83
<u>Platform and Locale</u> .....	87
<u>Customizing Key bindings</u> .....	87
<u>The Dynamic Nature of Key bindings</u> .....	90
<u>Conflict Resolution</u> .....	90
<b><u>Workbench window layout</u></b> .....	<b>90</b>
<u>Drop cursors</u> .....	91
<u>Fast views</u> .....	91
<u>Double-Click</u> .....	92
<u>Title bar context menu and amp; fast view toolbars</u> .....	92
<b><u>Editor area</u></b> .....	<b>92</b>
<u>Marker bar</u> .....	93
<u>Markers</u> .....	95
<u>Types of editors</u> .....	95
<b><u>Compare editor</u></b> .....	<b>97</b>
<u>Toolbar</u> .....	98
<b><u>Search view</u></b> .....	<b>99</b>
<b><u>Navigator view</u></b> .....	<b>99</b>

# Table of Contents

<b><u>Toolbar</u></b> .....	<b>99</b>
<u>Back</u> .....	99
<u>Forward</u> .....	99
<u>Up</u> .....	99
<u>Collapse All</u> .....	100
<u>Link with Editor</u> .....	100
<b><u>Menus</u></b> .....	<b>100</b>
<u>Select Working Set</u> .....	100
<u>Deselect Working Set</u> .....	100
<u>Edit Active Working Set</u> .....	100
<u>Sort</u> .....	101
<u>Filters</u> .....	102
<u>Link with Editor</u> .....	102
<b><u>Context menu</u></b> .....	<b>102</b>
<u>New</u> .....	102
<u>Go Into</u> .....	102
<u>Open</u> .....	102
<u>Open With</u> .....	102
<u>Copy</u> .....	102
<u>Paste</u> .....	102
<u>Delete</u> .....	102
<u>Move</u> .....	103
<u>Rename</u> .....	103
<u>Import</u> .....	103
<u>Export</u> .....	103
<u>Add Bookmark</u> .....	103
<u>Refresh</u> .....	103
<u>Close Project</u> .....	103
<u>Open Project</u> .....	103
<u>Team</u> .....	104
<u>Compare With</u> .....	104
<u>Replace With</u> .....	105
<u>Properties</u> .....	105
<b><u>Bookmarks view</u></b> .....	<b>105</b>
<u>Toolbar</u> .....	106
<u>Menus</u> .....	107
<b><u>Properties view</u></b> .....	<b>108</b>
<b><u>Outline view</u></b> .....	<b>109</b>
<b><u>Tasks view</u></b> .....	<b>109</b>
<u>Toolbar</u> .....	110
<u>Menus</u> .....	110

# Table of Contents

<b><u>CVS Repositories view</u></b> .....	<b>110</b>
<u>Titlebar</u> .....	110
<u>Go Home</u> .....	110
<u>Go Back</u> .....	110
<u>Go Forward</u> .....	110
<u>Refresh View</u> .....	111
<u>Collapse All</u> .....	111
<u>Drop-Down Menu</u> .....	111
<u>Context menu</u> .....	111
<u>New &gt; Repository Location</u> .....	111
<u>Configure Branches and Versions</u> .....	111
<u>Refresh Branches and Versions</u> .....	111
<u>Add to Branch List</u> .....	111
<u>Open</u> .....	111
<u>Show in Resource History</u> .....	111
<u>Compare</u> .....	112
<u>Compare With</u> .....	112
<u>Checkout as Project</u> .....	112
<u>Checkout as</u> .....	112
<u>Checkout Into</u> .....	112
<u>Checkout Out Module</u> .....	112
<u>Tag as Version</u> .....	113
<u>Tag with Existing</u> .....	113
<b><u>CVS Resource History view</u></b> .....	<b>113</b>
<u>Revision</u> .....	113
<u>Tags</u> .....	113
<u>Date</u> .....	113
<u>Author</u> .....	115
<u>Comment</u> .....	115
<b><u>Synchronize view</u></b> .....	<b>116</b>
<u>Modes</u> .....	116
<u>Synchronization state</u> .....	117
<u>Structure compare pane</u> .....	118
<u>Text compare pane</u> .....	119
<b><u>Ant editor</u></b> .....	<b>119</b>
<b><u>Ant view</u></b> .....	<b>119</b>
<u>Titlebar</u> .....	119
<u>Add Buildfile</u> .....	119
<u>Add Buildfiles with Search</u> .....	120
<u>Run the Default Target of the Selected Buildfile</u> .....	120
<u>Refresh View</u> .....	120
<u>Collapse All</u> .....	120
<u>Remove Selected Buildfile</u> .....	120
<u>Remove All Buildfiles</u> .....	120

# Table of Contents

<b><u>Ant view</u></b>	
<u>Context menu</u> .....	120
<u>Properties...</u> .....	121
<u>Open With &gt;</u> .....	121
<b><u>Help view</u></b> .....	<b>121</b>
<u>Table of Contents (Bookshelf)</u> .....	121
<u>Contents Tab</u> .....	121
<u>Search Results Tab</u> .....	122
<u>Links Tab</u> .....	122
<u>Bookmarks Tab</u> .....	122
<u>Search</u> .....	122
<u>Search scope</u> .....	122
<u>Go Back</u> .....	122
<u>Go Forward</u> .....	122
<u>Synchronize Navigation</u> .....	122
<u>Bookmark Document</u> .....	123
<u>Print Page</u> .....	124
<b><u>Update manager views</u></b> .....	<b>124</b>
<b><u>Install Configuration view</u></b> .....	<b>124</b>
<u>Titlebar</u> .....	124
<u>Go Home</u> .....	124
<u>Go Back</u> .....	124
<u>Go Into</u> .....	124
<u>Show Disabled Features</u> .....	125
<u>Context Menu – common</u> .....	125
<u>Go Home</u> .....	125
<u>Go Back</u> .....	125
<u>Go Into</u> .....	125
<u>Show Details</u> .....	125
<u>Context Menu – features</u> .....	125
<u>Show Status</u> .....	125
<u>Context Menu – histories</u> .....	125
<u>Save</u> .....	125
<u>Remove</u> .....	126
<u>Restore</u> .....	127
<u>Properties</u> .....	127
<b><u>Feature Updates view</u></b> .....	<b>127</b>
<u>Titlebar</u> .....	127
<u>Refresh</u> .....	128
<u>Drop Down Menu</u> .....	128
<u>Context Menu – common</u> .....	128
<u>Refresh</u> .....	128
<u>New, Site Bookmark</u> .....	128
<u>New, Folder</u> .....	128

# Table of Contents

<b><u>Feature Updates view</u></b>	
<u>New Search</u> .....	128
<u>Show Details</u> .....	128
<u>Context Menu – custom items</u> .....	130
<u>Delete</u> .....	133
<b><u>Preview</u></b> .....	<b>133</b>
<u>Titlebar</u> .....	133
<u>Go Home</u> .....	133
<u>Go Back</u> .....	135
<u>Go Forward</u> .....	135
<b><u>External Preview</u></b> .....	<b>135</b>
<u>Titlebar</u> .....	135
<u>Go Back</u> .....	136
<u>Go Forward</u> .....	136
<u>Go</u> .....	136
<u>Stop</u> .....	137
<u>Refresh</u> .....	137
<b><u>New Project wizard</u></b> .....	<b>138</b>
<u>Create a New Project Resource Page</u> .....	139
<u>Select Referenced Projects page</u> .....	140
<u>New Project perspective options</u> .....	141
<b><u>New Folder wizard</u></b> .....	<b>142</b>
<u>Advanced</u> .....	143
<b><u>New File wizard</u></b> .....	<b>144</b>
<u>Advanced</u> .....	145
<b><u>New CVS Repository Location wizard</u></b> .....	<b>146</b>
<b><u>CVS repository location</u></b> .....	<b>146</b>
<b><u>Merge wizard</u></b> .....	<b>147</b>
<u>Start Point</u> .....	150
<u>End state</u> .....	150
<b><u>Import wizard</u></b> .....	<b>152</b>
<u>File System</u> .....	153
<u>ZIP File</u> .....	153
<u>Existing Project</u> .....	155
<u>Team Project Set</u> .....	155
<b><u>Export wizard</u></b> .....	<b>157</b>
<u>File System</u> .....	158
<u>ZIP File</u> .....	160

# Table of Contents

<b><u>Export wizard</u></b>	
<u>Jar File</u> .....	162
<u>Team Project Set</u> .....	162
<b><u>Workbench User Guide</u></b> .....	<b>162</b>
<u>Getting started</u> .....	162
<u>Concepts</u> .....	162
<u>Tasks</u> .....	163
<u>Reference</u> .....	163
<b><u>File search</u></b> .....	<b>163</b>
<u>Containing text</u> .....	163
<u>Wildcards</u> .....	164
<u>File name patterns</u> .....	164
<u>Wildcards</u> .....	164
<u>Case sensitive</u> .....	165
<u>Scope</u> .....	165
<b><u>Help search</u></b> .....	<b>165</b>
<u>Search Expression</u> .....	165
<u>Advanced Search options</u> .....	167
<u>Scope</u> .....	167
<b><u>File menu</u></b> .....	<b>167</b>
<u>New</u> .....	167
<u>Close</u> .....	167
<u>Close All</u> .....	167
<u>Save</u> .....	167
<u>Save As</u> .....	167
<u>Save All</u> .....	167
<u>Revert</u> .....	167
<u>Move</u> .....	167
<u>Rename</u> .....	168
<u>Refresh</u> .....	168
<u>Print</u> .....	168
<u>Import</u> .....	168
<u>Export</u> .....	168
<u>Properties</u> .....	168
<u>Recent file list</u> .....	169
<u>Exit</u> .....	169
<b><u>Edit menu</u></b> .....	<b>169</b>
<u>Undo</u> .....	169
<u>Redo</u> .....	169
<u>Cut</u> .....	169
<u>Copy</u> .....	169
<u>Paste</u> .....	169
<u>Delete</u> .....	169

# Table of Contents

## **Edit menu**

<u>Select All</u> .....	170
<u>Find/Replace</u> .....	170
<u>Find Next</u> .....	170
<u>Find Previous</u> .....	170
<u>Incremental Find Next</u> .....	170
<u>Incremental Find Previous</u> .....	170
<u>Add Bookmark</u> .....	170
<u>Add Task</u> .....	170
<u>Convert Line Delimiters</u> .....	171
<u>Encoding</u> .....	171

## **Navigate menu**.....**171**

<u>Go Into</u> .....	171
<u>Go To</u> .....	171
<u>Open Resource</u> .....	171
<u>Show In</u> .....	171
<u>Next</u> .....	172
<u>Previous</u> .....	172
<u>Go to Line</u> .....	172
<u>Back</u> .....	173
<u>Forward</u> .....	173

## **Project menu**.....**173**

<u>Open Project</u> .....	173
<u>Close Project</u> .....	173
<u>Build Project</u> .....	173
<u>Rebuild Project</u> .....	173
<u>Build All</u> .....	173
<u>Rebuild All</u> .....	174
<u>Properties</u> .....	174

## **Window menu**.....**174**

<u>New Window</u> .....	174
<u>Open Perspective</u> .....	175
<u>Show View</u> .....	175
<u>Show/Hide Editors</u> .....	175
<u>Lock the Toolbars</u> .....	176
<u>Customize Perspective</u> .....	177
<u>Save Perspective As</u> .....	177
<u>Reset Perspective</u> .....	177
<u>Close Perspective</u> .....	177
<u>Close All Perspectives</u> .....	178
<u>Keyboard Shortcuts</u> .....	178
<u>Switch to Editor</u> .....	180
<u>Preferences</u> .....	180

# Table of Contents

<b><u>Help menu</u></b> .....	<b>180</b>
<u>Welcome</u> .....	180
<u>Tips and Tricks</u> .....	180
<u>Help Contents</u> .....	180
<u>Software Updates</u> .....	181
<u>About</u> .....	182
<b><u>Navigator view icons</u></b> .....	<b>183</b>
<b><u>Editor area marker bar</u></b> .....	<b>184</b>
<b><u>Tasks view</u></b> .....	<b>185</b>
<b><u>Toolbar buttons</u></b> .....	<b>185</b>
<b><u>External Tools and Ant icons</u></b> .....	<b>185</b>
<u>Objects</u> .....	185
<u>Launch configurations</u> .....	title
<u>Ant View</u> .....	title

# Crash recovery

The Workbench periodically saves a snapshot in order to reduce the risk of losing data due to crashes.

- Saved data is never lost as it is written to disk immediately (upon save).
- Unsaved data in open editors may be lost, depending on the editor implementation.
- Bookmarks and tasks might be lost.
- If a crash occurs during CVS synchronization, the Workbench may be out of sync. You can check by performing the synchronize operation again.
- Previously-created projects are never lost.

Developers of plug-ins can also choose to participate in this lifecycle and save the state of their plug-ins.

© Copyright IBM Corporation and others 2000, 2003.

# Preferences

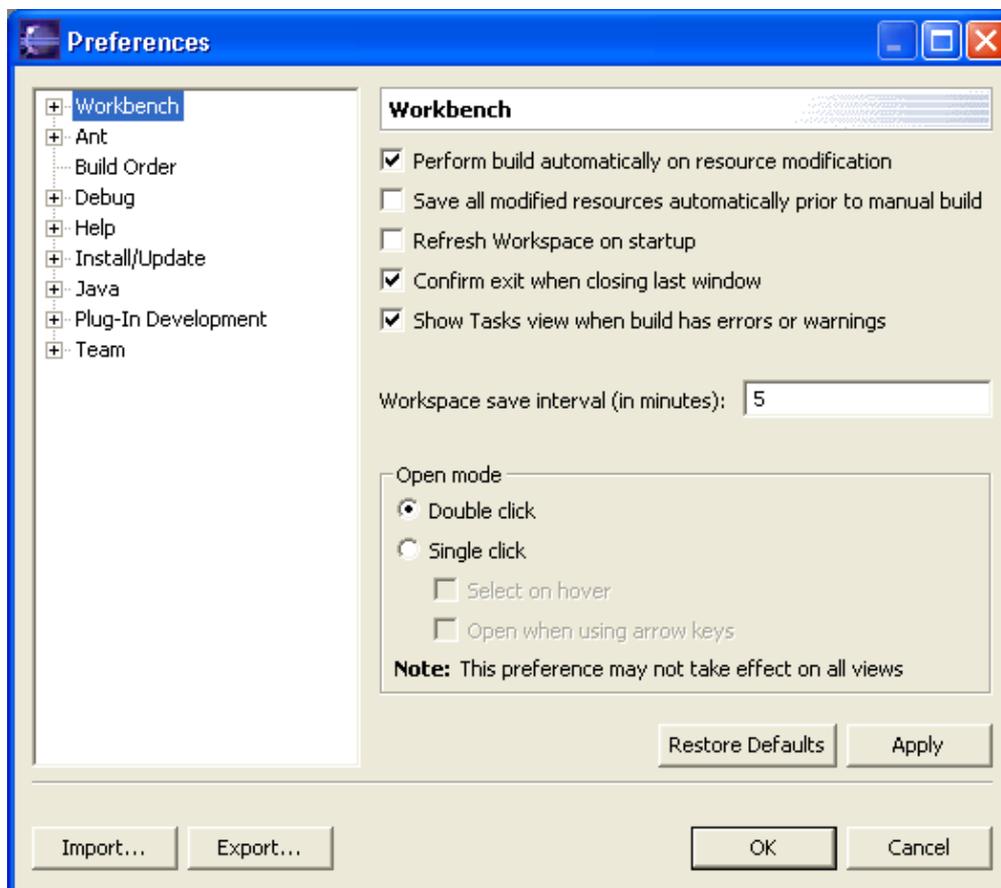
The Preferences dialog is the dialog used to set user preferences. It can be found from the main workbench **Window** menu under **Window > Preferences**. Preference pages contributed by plug-ins will be found in this dialog.

Most of the functionality of the preferences are defined by their individual pages but the dialog provides two general functions:

**Export:** Export will write any changes to the default preferences to a user-specified file.

**Import:** Import will apply the preferences in a user-specified file.

Here is what the preferences dialog looks like:



## ■ Related reference

[Workbench window layout](#)

# Workbench

The term *Workbench* refers to the desktop development environment.

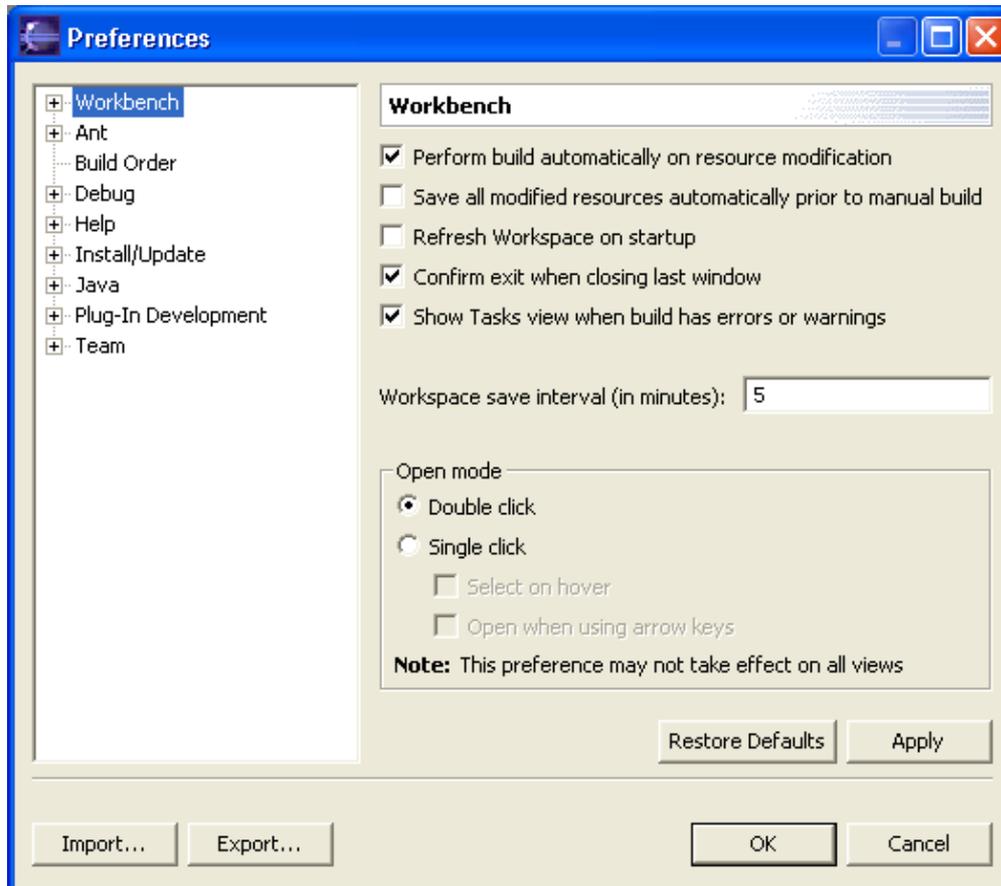
Each Workbench window contains one or more perspectives. Perspectives contain views and editors and control what appears in certain menus and tool bars. More than one Workbench window can exist on the desktop at any given time.

The following preferences can be changed on the Workbench page.

Option	Description	Default
Perform build automatically on resource modification	If this option is turned on, then the Workbench will perform an automatic build whenever a modified resource is saved.	On
Save all modified resources automatically prior to manual build	If this option is turned on, then whenever you perform a manual build (from the menu bar, select <b>Project &gt; Rebuild All</b> ), the Workbench will automatically save all resources that have been modified since the last build was performed.	Off
Refresh Workspace on startup	If this option is turned on then the workspace resources will be synchronized with their corresponding resources in the file system when the platform is started.  Note that this can potentially be a lengthy operation depending on the number of resources you have in your workspace.	Off
Confirm exit when closing last window	If this option is turned on, then when you close the last workbench window you will be prompted to confirm that you want to exit. You will be able to disable this option in the prompt.	On
Show Tasks view when build has errors or warnings	If this option is turned on, then when a build results in new errors or warnings the Tasks view will be activated.	On
Workspace save interval (in minutes)	This number indicates how often the state of the workspace is automatically saved to disk.	5
Open mode...	You can select one of the following methods for opening resources: <ul style="list-style-type: none"> <li>• Double click – Single clicking on a resource will select it and double clicking on it will open it in an editor.</li> <li>• Single click (Select on hover) – Hovering the mouse cursor over the resource will select it and clicking on it once will open it in an editor.</li> <li>• Single click (Open when using arrow keys) – Selecting a resource with the arrow keys will open it in an editor.</li> </ul> <p>Note: Depending on which view has focus, selecting and opening a resource may have different behavior.</p>	Double click

## Crash recovery

Here is what the Workbench preferences page looks like:



### ● Related reference

[Workbench Window Layout](#)

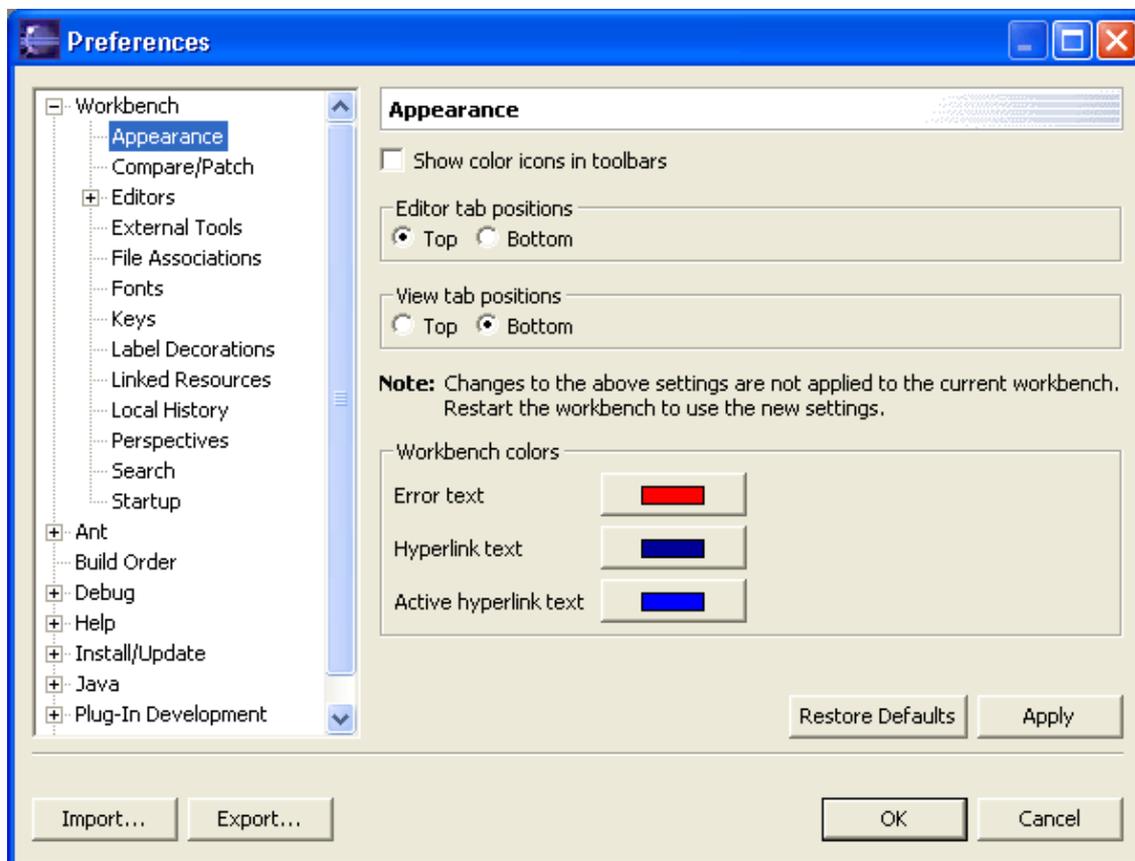
© Copyright IBM Corporation and others 2000, 2003.

# Appearance

The following preferences can be changed on the Appearance page.

Option	Description	Default
Tab positions – Editors	Specify either top or bottom to indicate where you want tabs for stacked editors to appear.	Top
Tab positions – Views	Specify either top or bottom to indicate where you want tabs for stacked views to appear.	Bottom
Workbench colors	Use these options to change the colors for the text shown in Workbench. <ul style="list-style-type: none"> <li>• Error text (<i>red</i>)</li> <li>• Hyperlink text (<i>navy</i>)</li> <li>• Active Hyperlink text (<i>blue</i>)</li> </ul>	Default values are shown in <i>italics</i> .

Here is what the Appearance preferences page looks like:



© Copyright IBM Corporation and others 2000, 2003.

# Compare/Patch

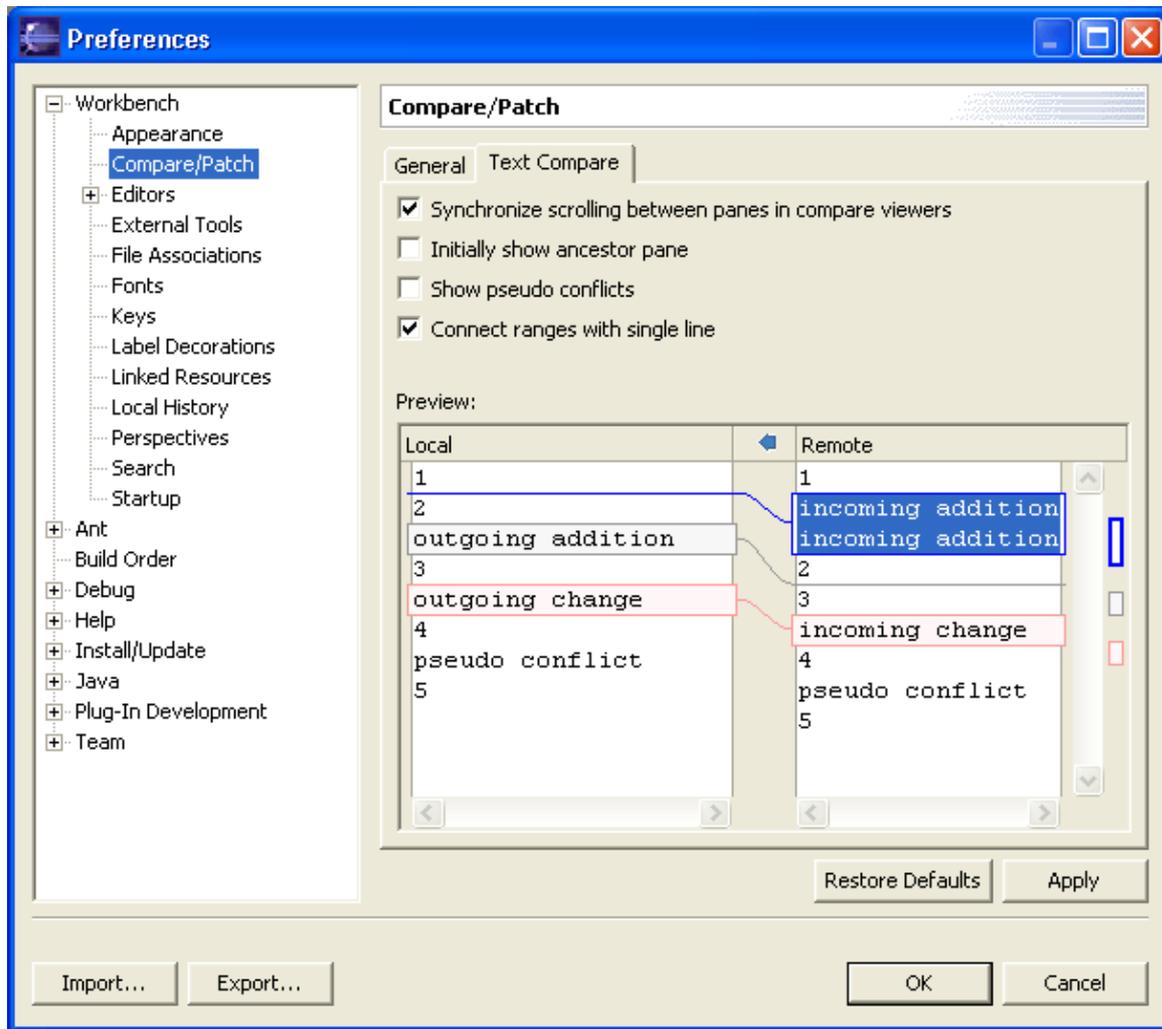
The following preferences can be changed on the Compare/Patch page.

Option	Description	Default
Open structure compare automatically	This option controls whether a structure compare is automatically performed whenever a content compare is done. Turn this option off if you don't want to see the structural differences.	On
Show additional compare information in the status line	If this option is on, additional information about a change is shown in the status line. Turn this option on if you are interested in additional information about a change.	Off
Ignore white space	This option controls whether or not whitespace change are shown in the compare viewer. Turn this option on if you want to see changes in whitespace.	Off
Automatically save dirty options before patching	This option controls whether any unsaved changes are automatically saved before a patch is applied. Turn this option on if you want to save changes automatically.	Off

Option	Description	Default
Synchronize scrolling between panes in compare viewers	The two comparison 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. Turn this option off if you do not want the compare viewers to lock scroll.	On
Initially show ancestor pane	Sometimes you 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 compare. Turn this option on if you want the ancestor pane to always appear at the start of a comparison.	Off
Show pseudo conflicts	Displays pseudo conflicts, which occur when two developers make the same change (e.g., both add or remove the exact same line of code or comment). Turn this option on if you want pseudo conflicts to appear in compare browsers.	Off
Connect ranges with single line	Controls whether differing ranges are visually connected by a single line or a range delimited by two lines.	On

Here is what the Compare preference page looks like:

## Crash recovery



### ■ Related reference

[Compare Editor](#)

[CVS Synchronization View](#)

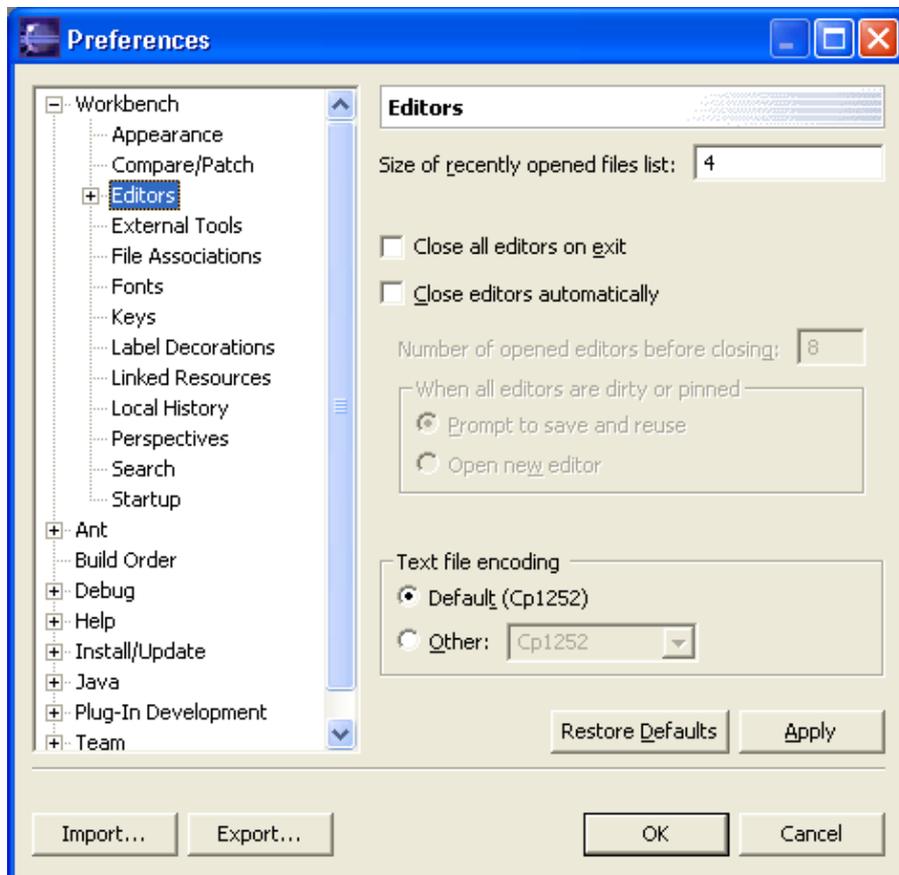
© Copyright IBM Corporation and others 2000, 2003.

# Editors

The following preferences can be changed on the Editors page.

Option	Description	Default
Size of recently opened files list	For each file that is opened in an editor, it is stored in a list of recently used files. This option controls the number of files which is displayed in this list in the File menu.	4
Close all editors on exit	This option is used to specify whether or not to close all editors when the Workbench is exited. Enabling this option will speed up start up as reducing the amount of work required to start Eclipse.	Off
Close editors automatically	This option is used to specify whether or not to re-use editors in the Workbench. If turned on, then you may specify the number of editors to use before they are recycled (Default is 8). You can also specify if a prompt dialog should be opened or if a new editor should be opened when all editors are dirty. Once it is turned on, the Pin Editor action is added to the toolbar and editor tab menu. Pinned editors are not recycled.	Off
Text File Encoding	Use this option to specify the encoding to use when saving text files in editors.	Default (CP1252)

Here is what the Editors preference page looks like:



■ Related reference

[Editor Area](#)

© Copyright IBM Corporation and others 2000, 2003.

# Text Editor

The following preferences can be changed on the Text Editor page.

## Appearance options

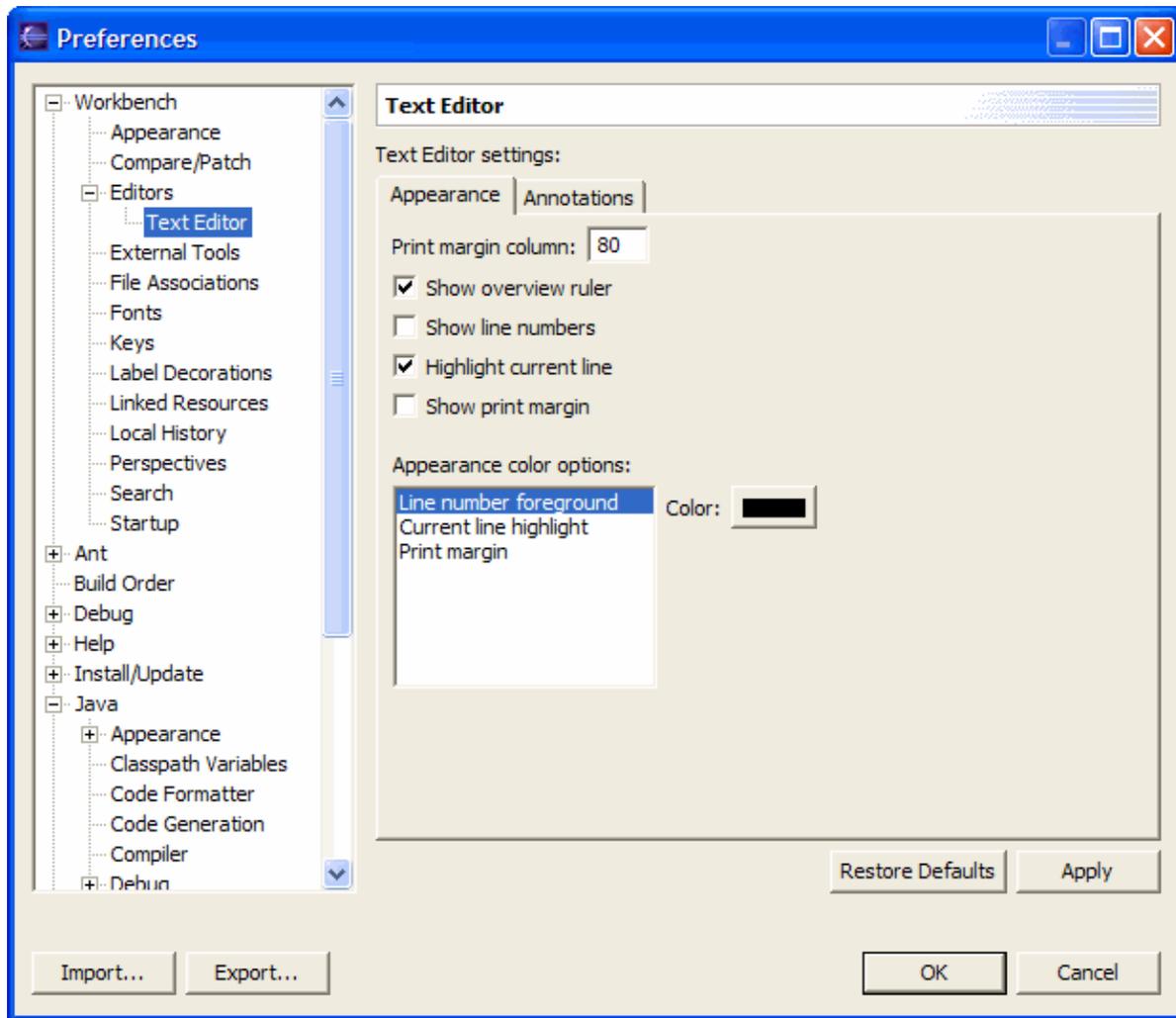
Option	Description	Default
Print margin column	This option allows to set the print margin for the text editor.	80
Show overview ruler	This option controls whether the overview ruler on the right side of the text editor is shown.	On
Show line numbers	This option controls whether or line numbers are shown in on the left side of the text editor.	Off
Highlight current line	This option controls whether or the current line is highlighted or not.	On
Show print margin	This option controls whether the print margin is visible or not.	Off
Appearance color options	This option controls various appearance colors.	

## Annotations options

Option	Description
Show in text	This option controls whether the selected annotation type is shown in the text. The corresponding text will be underlined with squiggles.
Show in overview ruler	This option controls whether the selected annotation type is shown in the overview ruler.
Color	This option controls the color for the selected annotation type.

Here is what the Text editor preference page looks like:

## Crash recovery



### ● Related reference

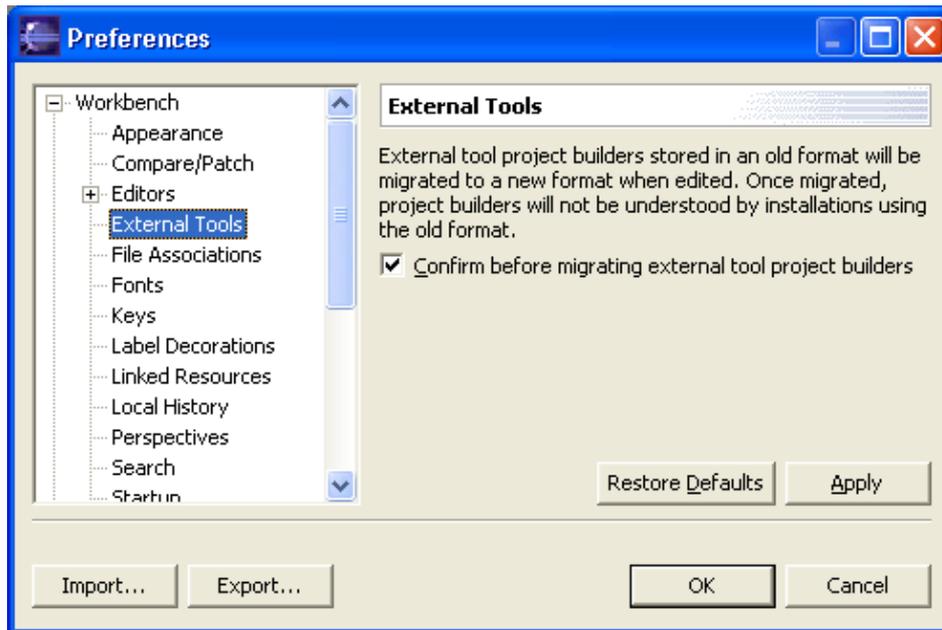
#### Editor

© Copyright IBM Corporation and others 2000, 2003.

# External Tools

The following preferences can be changed on the External Tools page.

You can configure whether or not you are prompted before external tool project builders are migrated.



## ● Related concepts

[Ant support](#)

## ● Related reference

[Ant preferences](#)

[Ant runtime preferences](#)

[Ant editor preferences](#)

© Copyright IBM Corporation and others 2000, 2003.

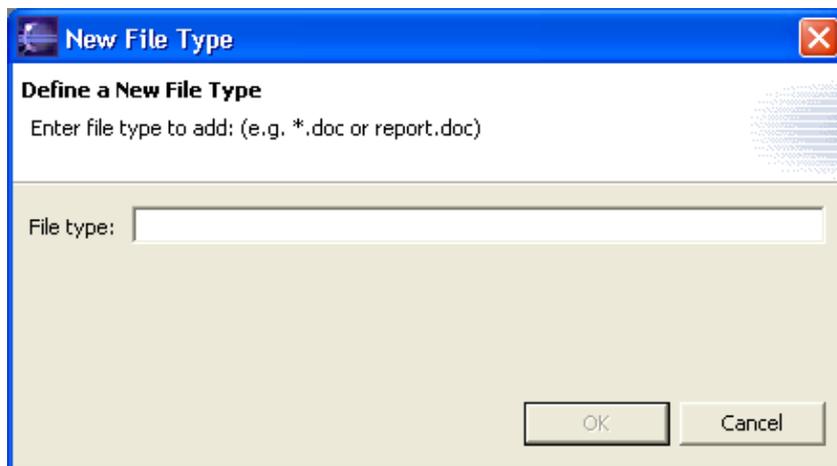
# File Associations

On the File Associations preference page, you can add or remove file types recognized by the Workbench. You can also associate editors with file types in the file types list.

## File types list

- **Add:** Adds a new file or file type (extension) to the predefined list. In the resulting New File Type dialog, type the name of a file or a file extension. If you are adding a file extension, you must type either a dot or a "\*" before the file type (e.g., ".xml" or "\*.xml" as opposed to simply "xml").
- **Remove:** Removes the selected file type from the list

Dialog to create a new file type:

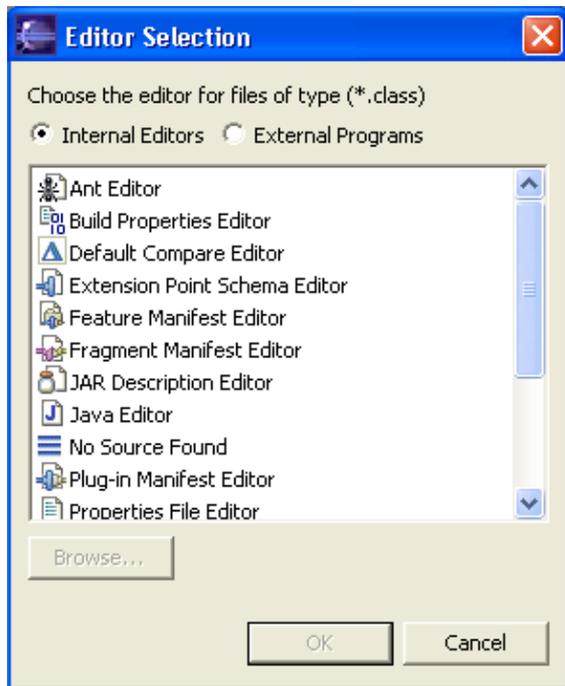


## Associated editors list

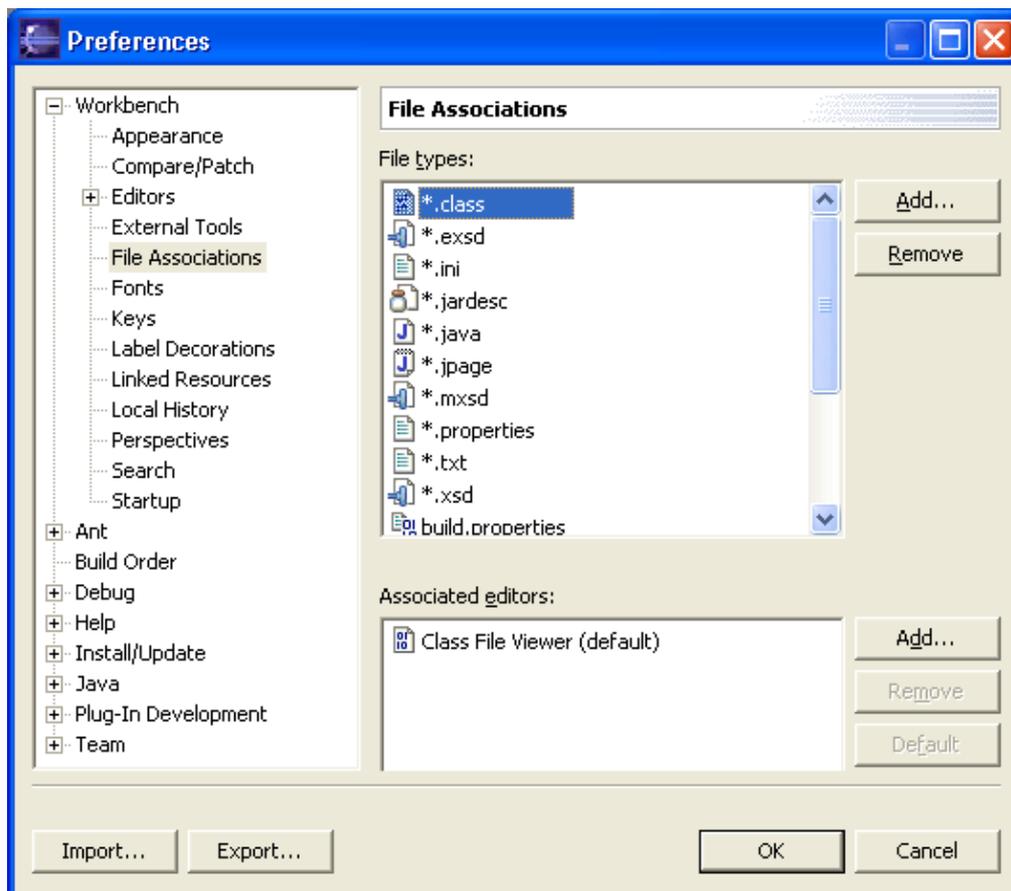
- **Add:** Adds a new editor to the list of editors associated with the file type selected above. In the resulting Editor Selection dialog, you can choose an editor to launch either inside the Workbench (internal) or outside the Workbench (external); click **Browse** to locate an editor yourself if the editor you want is not displayed in the list.
- **Remove:** Removes the association between an editor and the file type selected above.
- **Default:** Sets the selected editor as the default editor for the file type selected above. The editor moves to the top of the Associated Editors list to indicate that it is the default editor for that file type.

Dialog to create a new file association:

## Crash recovery



Here is what the File Associations preference page looks like:



## Crash recovery

© Copyright IBM Corporation and others 2000, 2003.

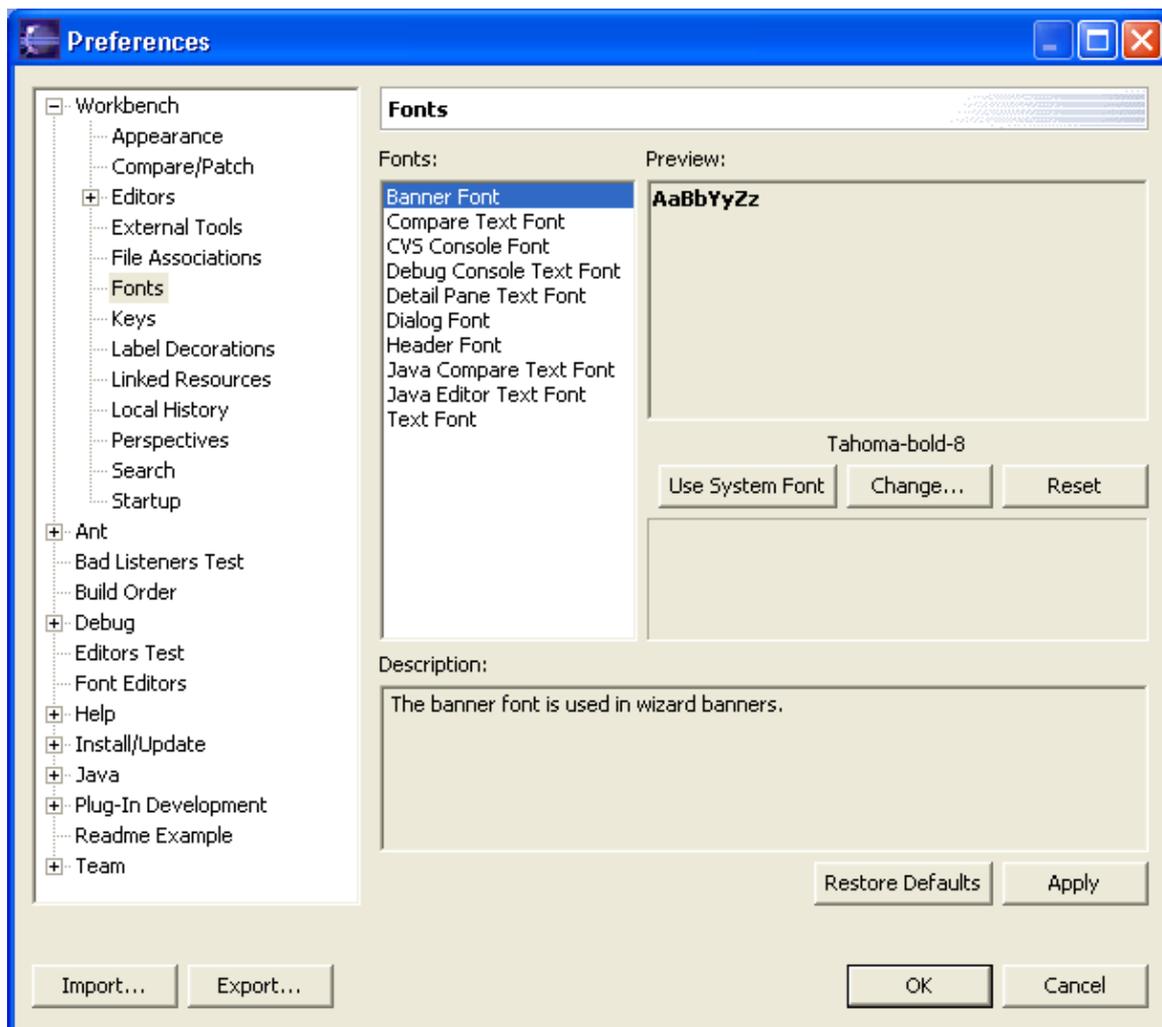
# Fonts

The fonts used for various types of text can be set using the Fonts preference page.

A Fonts list allows the selection of the kind of text where the font will appear while a Preview area shows the appearance of the current font setting.

The font setting can be changed either by pressing the Use System Font button to choose the Operating System font setting or pressing the Change button to open up a font selection dialog.

Here is what the Fonts preference page looks like:



## ■ Related tasks

[Changing fonts and colors](#)

# Keys

The function of the keyboard can be extensively customized in Eclipse. Within Eclipse, key strokes and key sequences are assigned to invoke particular commands.

## Key strokes, Key sequences, and Key bindings

A 'key stroke' is the pressing of a key on the keyboard, while optionally holding down one or more of these modifier keys: `Ctrl`, `Alt` (`Option` on the Macintosh), `Shift`, or `Command` (only on the Macintosh.) For example, holding down `Ctrl` then pressing `A` produces the key stroke `Ctrl+A`. The pressing of the modifier keys themselves do not constitute key strokes.

A 'key sequence' is one or more key strokes. Traditionally, emacs assigned two or three key stroke key sequences to particular commands. For example, the normal key sequence assigned to `Close All` in emacs is `Ctrl+X Ctrl+C`. To enter this key sequence, one presses the key stroke `Ctrl+X` followed by the key stroke `Ctrl+C`. Eclipse supports key sequences of arbitrary lengths.

A 'key binding' is the assignment of a key sequence to a command.

## Configurations

A 'configuration' is a set of key bindings. Eclipse includes two configurations:

- Standard
- Emacs (extends Standard)

The *Standard* configuration contains a general set of key bindings, in many cases recognizable to users as traditional key sequences. For instance, `Ctrl+A` is assigned to `Select All`, `Ctrl+S` is assigned to `Save`, etc.

The *Emacs* configuration contains a set of key bindings familiar to users of emacs. For instance, `Ctrl+X H` is assigned to `Select All`, `Ctrl+X S` is assigned to `Save`, etc.

It is important to understand why the *Emacs* configuration says that it 'extends Standard'. The *Emacs* configuration is not a complete set of key bindings like the *Standard* configuration. Rather, it borrows from the *Standard* configuration where possible, only defining explicit emacs style key bindings where they vary from the *Standard* configuration. Generally, only well known commands like `Select All`, `Save`, etc. have specific emacs key sequences associated with them.

The user decides which configuration they are most comfortable using by changing the 'Active Configuration' setting on the keys preference page. If the user chooses the *Standard* configuration, all *Emacs* key bindings are ignored. If the user chooses the *Emacs* configuration, explicit emacs style key sequence assignments take precedence over any conflicting assignments in the *Standard* configuration.

## Scopes

Key bindings can vary based on the current context of Eclipse.

## Crash recovery

Sometimes the active part might be a java file editor, for instance, where a different set of key sequence assignments may be more appropriate than if the active part was an html file editor. As a specific example, typically `Ctrl+B` is assigned to `Build` in a context such as java file editing, while `Ctrl+B` is assigned to `Make Text Bold` in a context such as html file editing. This context is called 'scope' and is determined by the active part. If the active part does not choose a particular scope, the workbench will set the active scope to *Global*.

Eclipse includes three different scopes. They are:

- Global
- Text Editor (extends Global)
- Java Editor (extends Text Editor)

Much like configurations, scopes can extend other scopes. For example, the *Java Editor* scope borrows key bindings from the *Text Editor* scope, which in turn borrows key bindings from the *Global* scope.

## Platform and Locale

Key bindings also vary by platform and locale. On the Macintosh platform, `Command+S` is assigned to `Save`, instead of the usual `Ctrl+S`. On Chinese locales (zh), `Alt+/` is assigned to `Content Assist`, instead of the usual `Ctrl+Space`.

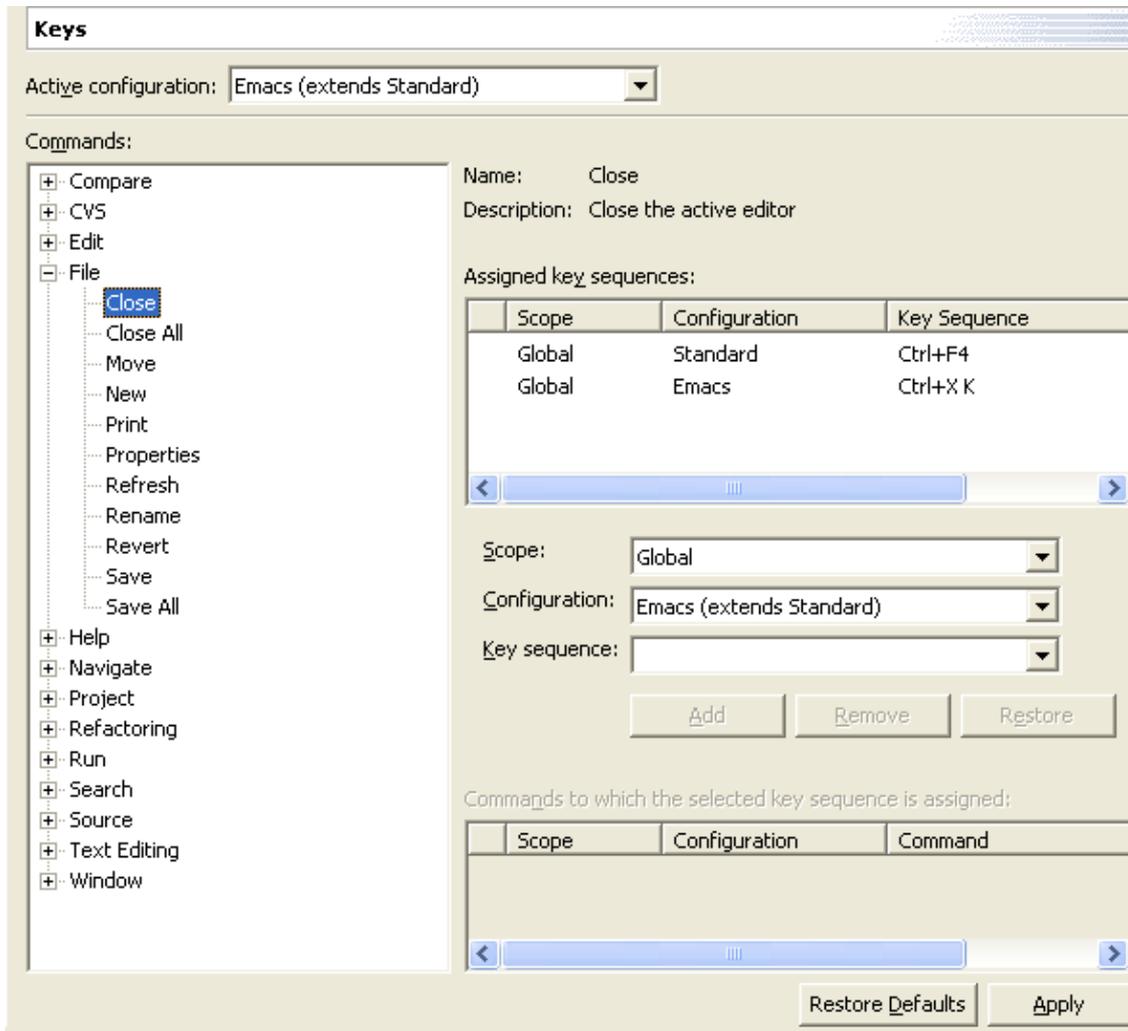
The current platform and locale is determined when Eclipse starts, and does not vary over the course of an Eclipse instance.

## Customizing Key bindings

With multi-stroke key sequences, configurations, and scopes, there are a lot of things to keep in mind when customizing key bindings. To make things easier, all key customization is done on the Keys preference page.

Select **Window > Preferences > Workbench > Keys** for the Keys preference page.

## Crash recovery



In this example, we have chosen to select *Emacs* as the active configuration, and have selected the command `Close` from the list of commands. Information on this command, along with its current key bindings, is shown on the right side.

Notice that `Close` has two key sequences assigned to it: `Ctrl+F4` in the *Standard* configuration and `Ctrl+X K` in the *Emacs* configuration. Both are assigned in the *Global* scope. Thus, if the user sets the active configuration to *Standard*, `Ctrl+F4` will be assigned to `Close` while `Ctrl+X K` will not. If the user sets the active configuration to *Emacs*, however, `Ctrl+X K` will be assigned to `Close`. As well, because the *Emacs* configuration also borrows key bindings from the *Standard* configuration, `Ctrl+F4` will also be assigned to `Close` provided that `Ctrl+F4` has not been assigned to another command in the *Emacs* configuration.

Below the list of key sequences assigned to `Close`, there is a place to add or remove key bindings. By default, it selects the scope as *Global* and the configuration to be the user's active configuration (recall above where we changed this to *Emacs*).

We type in the key sequence `Ctrl+W`, and the 'Add' button becomes enabled. Also, a list of all the commands to which the key sequence `Ctrl+W` is already assigned is shown below the Add button. We can see that `Ctrl+W` is currently assigned to the command `Cut` in the *Global* scope and *Emacs* configuration. We click the 'Add' button to assign `Ctrl+W` to `Close`.

## Crash recovery

Scope:

Configuration:

Key sequence:

Commands to which the key sequence 'Ctrl+W' is assigned:

Scope	Configuration	Command
Global	Emacs	Cut
Text Editor	Standard	Show Ruler Context Menu

Now we see Ctrl+W added to

the list of key sequences assigned to Close. Notice the small 'change' graphic  indicating that this key binding changes an existing key binding. Our new key binding, effectively removes the previous assignment of Ctrl+W to Cut. We can remove this change at any time by selecting our new key binding and clicking the 'Remove' button. The previous assignment of Ctrl+W to Cut will automatically be restored.

Name: Close

Description: Close the active editor

Assigned key sequences:

Scope	Configuration	Key Sequence
 Global	Emacs	Ctrl+W (was: Cut)
Global	Standard	Ctrl+F4
Global	Emacs	Ctrl+X K

Scope:

Configuration:

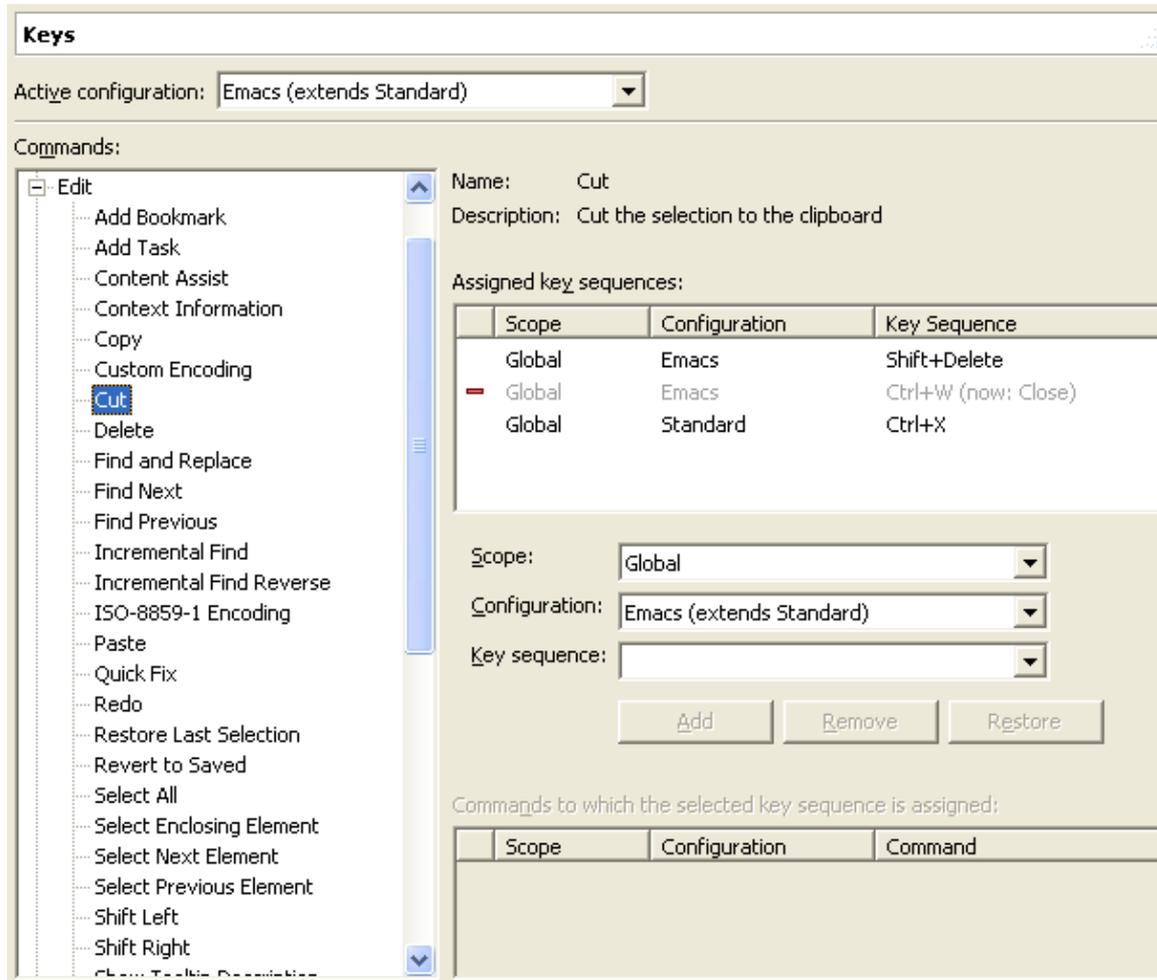
Key sequence:

Commands to which the key sequence 'Ctrl+W' is assigned:

Scope	Configuration	Command
 Global	Emacs	Close (was: Cut)
Text Editor	Standard	Show Ruler Context Menu

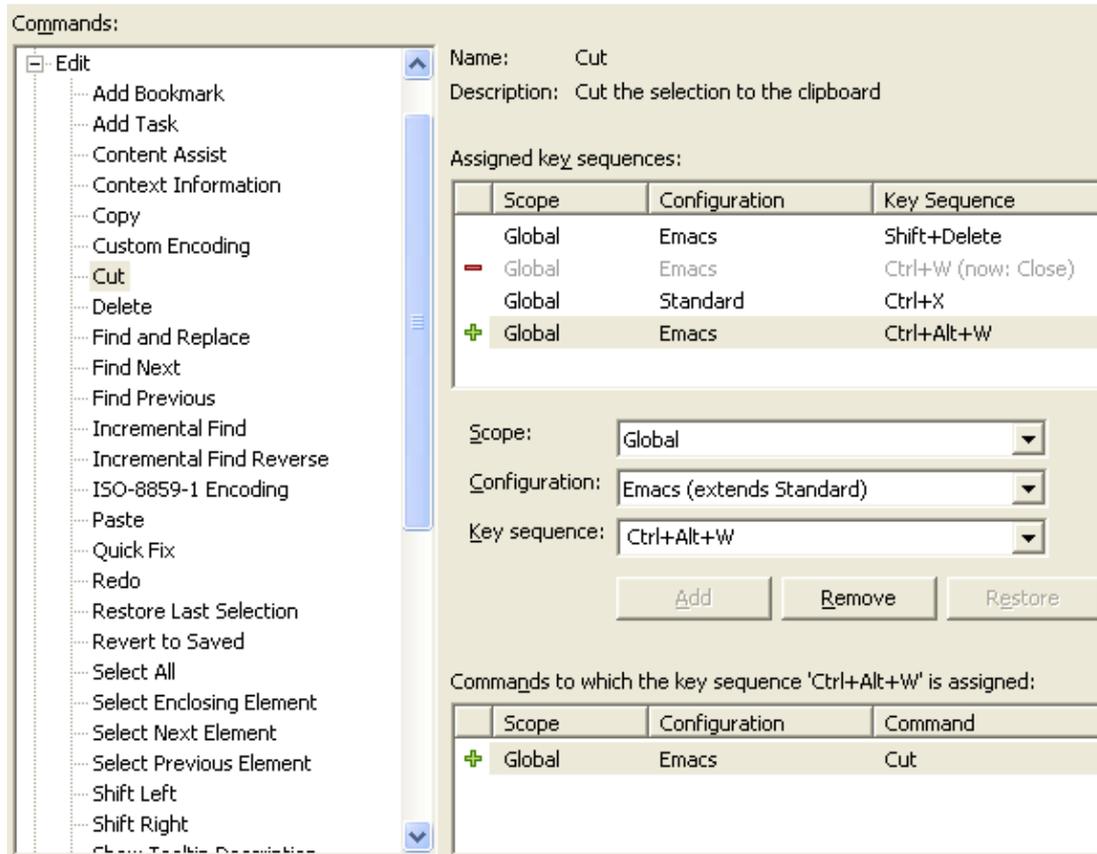
By selecting the command Cut, we can see the results of our change. Notice the small 'minus' graphic  indicating that the key binding was removed. We can restore this key binding at any time by selecting it here and clicking the 'Restore' button, effectively removing the new key binding we added in the previous step.

## Crash recovery



Perhaps we choose to assign another key to Cut in the *Emacs* configuration, say `Ctrl+Alt+W`. Adding this in the previous manner above produces the following. Notice the small 'plus' graphic  indicating that the key binding was added by the user, and was previously unassigned:

## Crash recovery



## The Dynamic Nature of Key bindings

Key bindings are provided by plug-ins, and in Eclipse, plug-ins can be added or removed. This can cause key bindings declared by these plug-ins to be added or removed. Eclipse stores custom key bindings in a way to automatically compensate for this. Consider the example above where `Ctrl+Alt+W` was assigned to `Cut` in the *Emacs* configuration. Consider the user installs a new plug-in which assigns `Ctrl+Alt+W` to a particular command. Eclipse will preserve the user's assignment to `Cut`, but show the key binding with the small 'change' graphic instead of the 'plus' graphic.

## Conflict Resolution

There are only a finite number of simple, common key strokes available to assign to a multitude of commands. We have seen that configuration, scope, platform, and locale all partition key sequence assignments into domains where they don't conflict with one another. Consider the case for `Ctrl+B` above if scopes did not exist. One plug-in would assign `Ctrl+B` to `Build`, the other plug-in would assign `Ctrl+B` to `Make Bold Text`. How would Eclipse properly resolve this conflict?

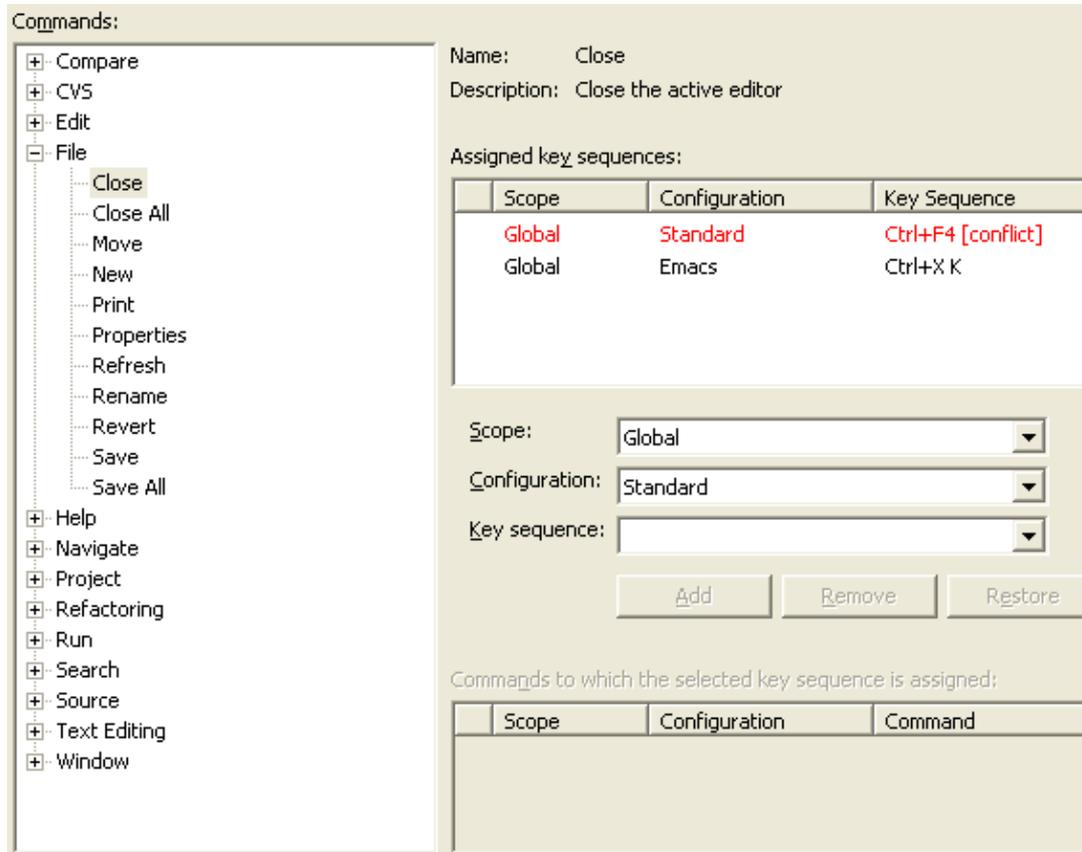
Though conflicts are drastically reduced by employing the above mechanisms, they can still occur. Two plug-ins, independent of one another, could assign the same key sequence to different commands with the same scope, configuration, platform, and locale. Consider if a plug-in assigned `Ctrl+F4` in the *Global* scope and *Standard* configuration to one of its commands. This directly conflicts with Eclipse assigning `Ctrl+F4` to the close command in the same scope and configuration.

This is a conflict. It wouldn't be proper to invoke both commands, nor would it be proper to simply choose

## Crash recovery

one of the two commands to receive the key stroke. The only proper thing to do is to ignore both key bindings, making `Ctrl+F4` effectively useless in this scope and configuration.

Conflicts of this nature are shown in the Keys preference page as follows. Notice the red text and the word "[conflict]":



These types of conflicts can be resolved by the user explicitly assigning the key sequence to one of the commands.

Another type of conflict can be caused by multiple-key stroke key sequences. For example, in the *Emacs* configuration, there are many multiple-key stroke key sequences beginning with the key stroke `Ctrl+X`. `Ctrl+H K` is assigned to `Close`. `Ctrl+X H` is assigned to `Select All`.

As previously mentioned, the *Emacs* configuration borrows key bindings from the *Standard* configuration. In the standard configuration, `Ctrl+X` is assigned to `Cut`. Though the *Emacs* configuration doesn't explicitly redefine `Ctrl+X`, pressing `Ctrl+X` is required as part of many of its key bindings. In the *Emacs* configuration, when one presses `Ctrl+X`, one is half way to entering one of many possible assigned key sequences. One would not expect the `Cut` action to be invoked at this time.

For this type of conflict, the rule is that the `Ctrl+X` key sequence assigned to `Cut` would be ignored. Otherwise, it would not be possible to complete many of the key bindings in the *Emacs* configuration.

### ■ Related concepts

[Accessibility Features in Eclipse](#)

[Changing the key bindings](#)

[Online help system](#)

■ **Related reference**

[Font and color settings in Eclipse](#)

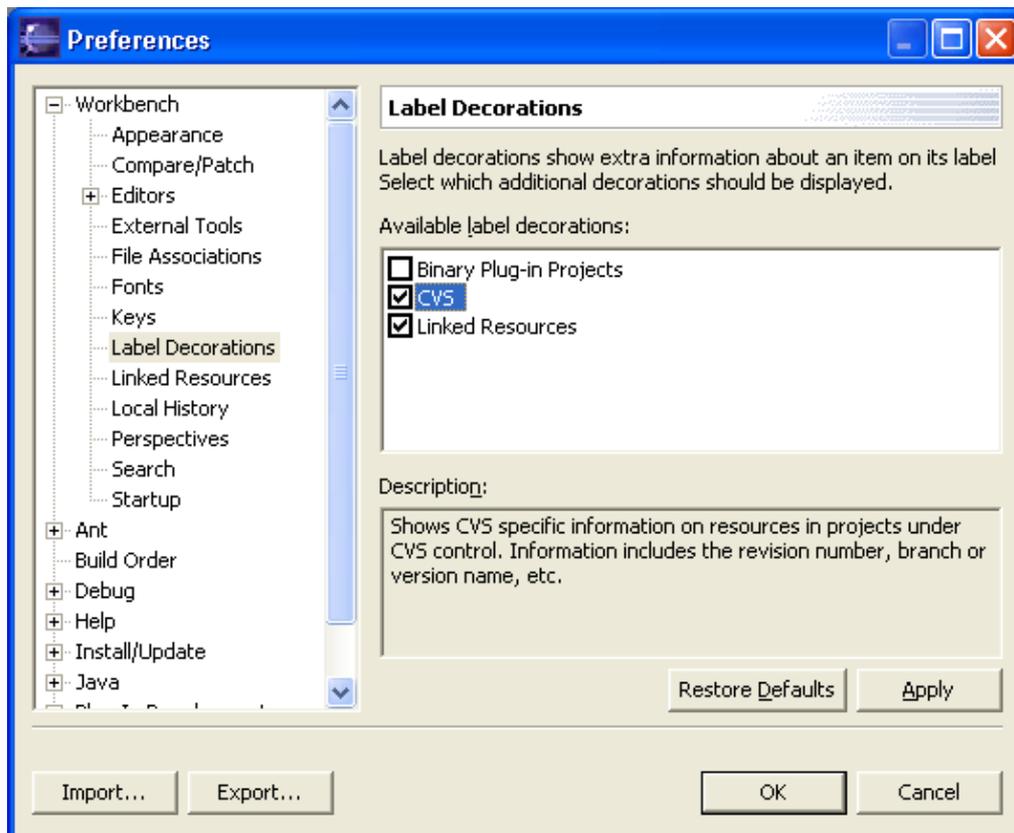
© Copyright IBM Corporation and others 2000, 2003.

# Label Decorations

Label Decorations allow additional information to be displayed in an item's label and icon.

The Label Decorations preference page provides a description of each decoration and allows the selection of which decorations are visible.

Here is what the Label Decorations preference page looks like:



## ● Related concepts

### Label Decorations

© Copyright IBM Corporation and others 2000, 2003.

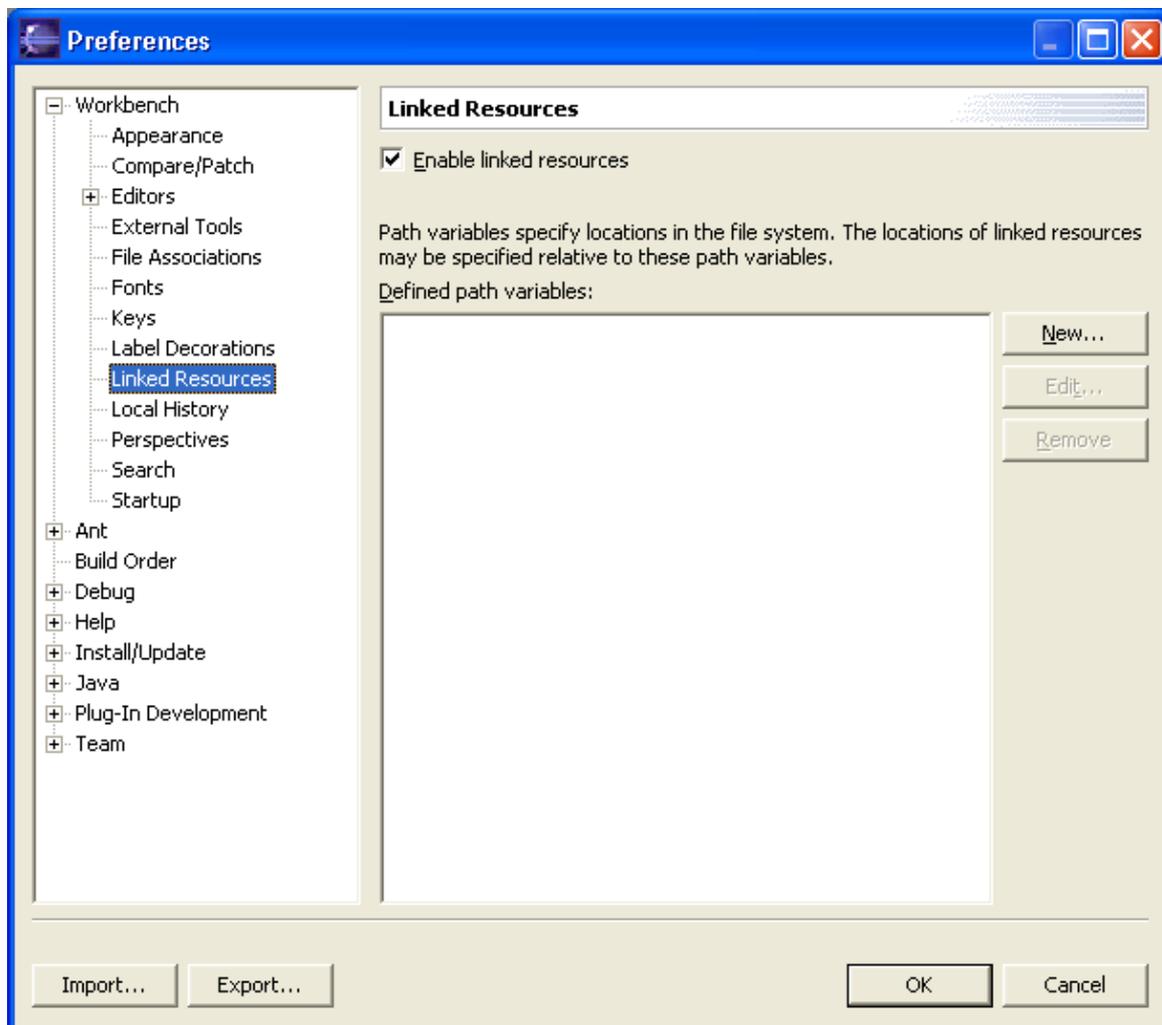
# Linked Resources

This preference page is used when working with linked resources. The preference **Enable linked resources** is used to globally enable or disable the linked resource feature for the entire workspace. By default, linked resources are enabled. If you disable linked resources, then you will not be able to create any new linked resources, or import existing projects that contain linked resources.

Not all versions of the workbench support linked resources and recognize them as such. You may not want to use linked resources if you plan to share your workspace data with other users. Disable this preference if they will not be able to work with linked resources.

The remainder of this page is for defining path variables that are used when creating linked resources. Use the **New** button to define new variables, the **Edit** button to change the value of an existing variable, and the **Remove** button to get rid of an existing variable. Note if you change a path variable that is currently in use, you will need to perform a local refresh on those projects to "discover" what is different in the file system. You can refresh a resource by opening the Navigator context menu for that resource and selecting **Refresh**. It is not recommended that you remove a path variable that is currently in use.

Here is what the Linked Resources preference page looks like:



## Crash recovery

- Related concepts

[Linked resources](#)

[Path variables](#)

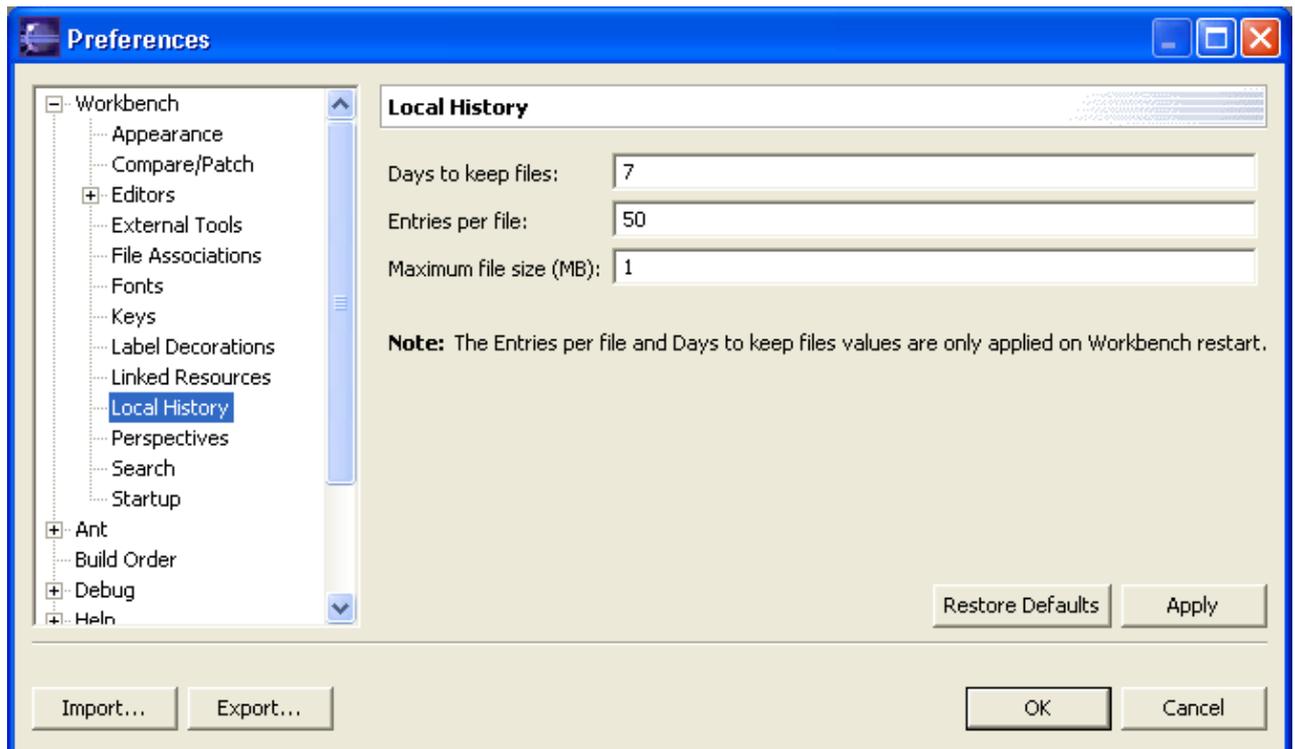
© Copyright IBM Corporation and others 2000, 2003.

# Local History

The following preferences can be changed on the Local History page.

Option	Description	Default
Days To Keep Files	Indicates for how many days you want to maintain changes in the local history. History state older than this value will be lost.	7 days
Entries Per File	Indicates how many history states per file you want to maintain in the local history. If you exceed this value, you will lose older history to make room for new history.	50 entries
Maximum File Size (MB)	Indicates the maximum size of individual states in the history store. If a file is over this size, it will not be stored.	1 MB

Here is what the Local History preference page looks like:



## ● Related reference

[Local History](#)

© Copyright IBM Corporation and others 2000, 2003.

# Perspectives

On the Perspectives preference page, you can manage the various perspectives defined in the Workbench.

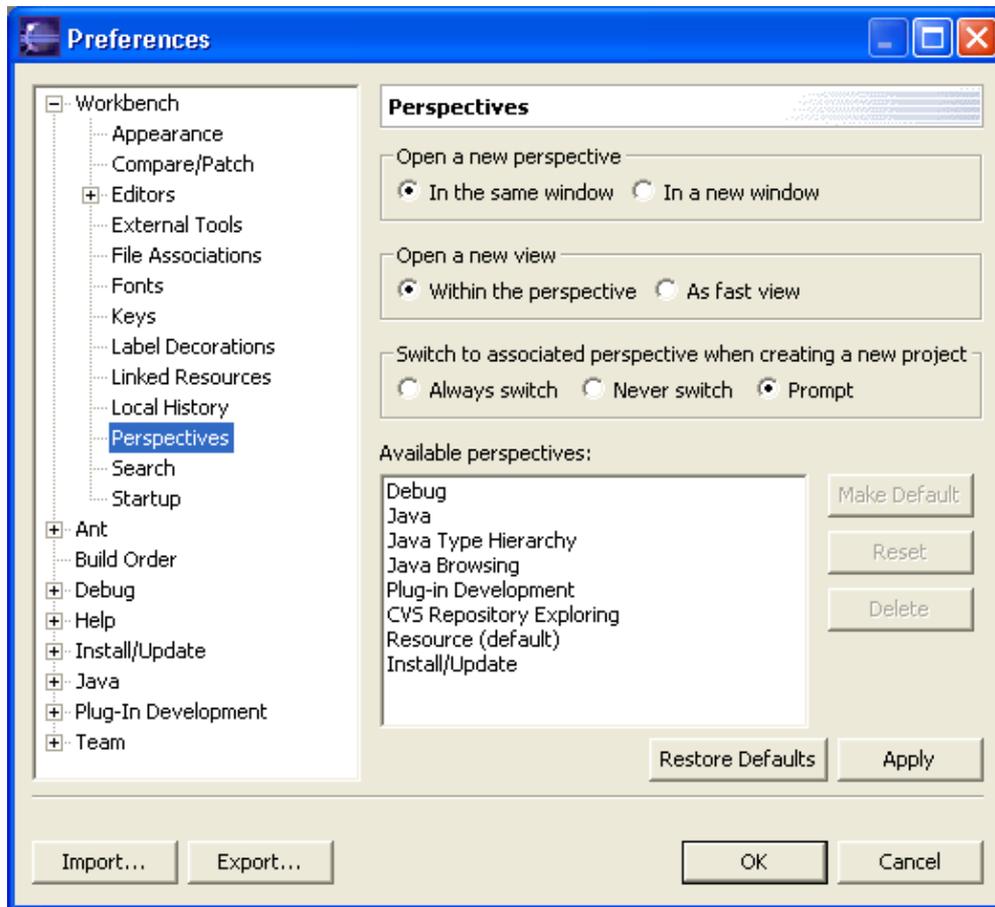
Option	Description	Default
Open a new perspective	Use this option to set what happens when you open a new perspective. Do you want the perspective opened within the current Workbench window or opened in a new window?	In the same window
Open a new view	Use this option to specify what happens when a new view is opened. It is either opened to its default position within the current perspective or it is opened as a fast view and docked to the side of the current perspective.	Within the perspective
New project options	Use this option to specify the perspective behavior when a new project is created. You can set it to switch the current perspective to be the one associated with the project type and open the perspective in the same Workbench window as the current one, switch the perspective and open it in a new Workbench window, or not to switch perspectives at all.	Open perspective in the same window

## Available Perspectives Options:

Option	Description	Default
Make Default	Sets the selected perspective as the default perspective.	Resource
Reset	Resets the definition of the selected perspective to the default configuration. This option is only applicable to built-in perspectives that have been overwritten using Window > Save Perspective As...	n/a
Delete	Deletes the selected perspective. This option is only applicable to user-defined perspectives (built-in perspectives can not be deleted).	n/a

Here is what the Perspectives preferences page looks like:

## Crash recovery



### ■ Related reference

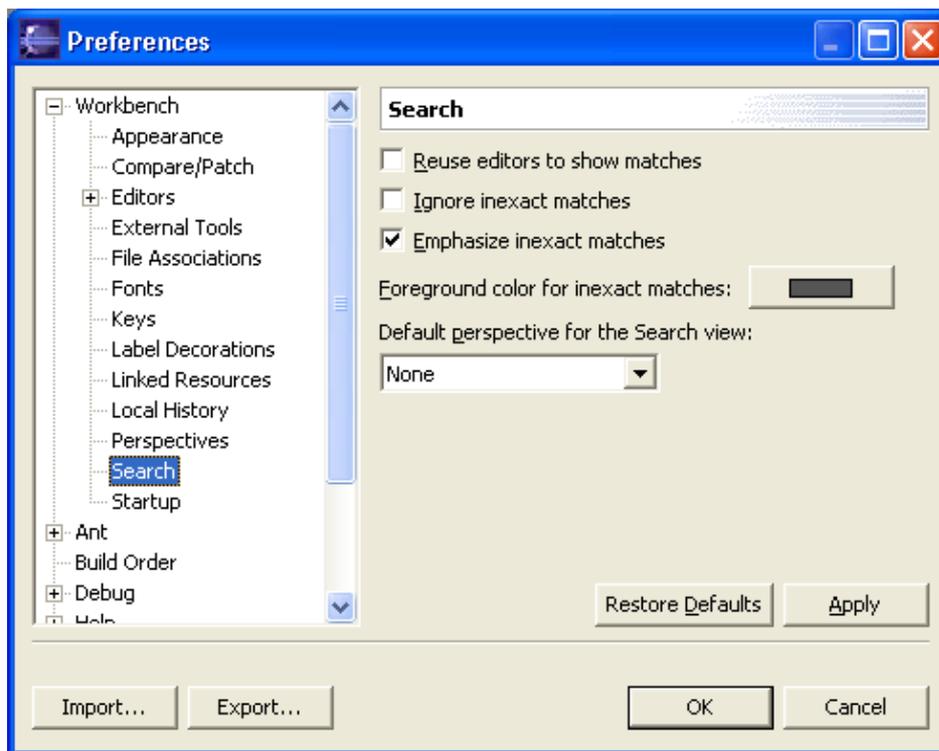
#### Perspectives

© Copyright IBM Corporation and others 2000, 2003.

# Search

The Search preference page allows the user to set preferences for searches. The *Reuse editors to show matches* option allows the user to keep using the same editor for search results to reduce the number of open editors. *Emphasize inexact matches* is an option that allows you to highlight them in the Search view. If the Search engine isn't 100% sure about the match then the match is considered inexact. You may also set the *foreground color for inexact matches*. If you only want to see exact matches you can check *Ignore inexact matches*. The *Default perspective for the Search view* allows you to define which perspective should be brought to front when there are new search results.

Here is what the Search preference page looks like:



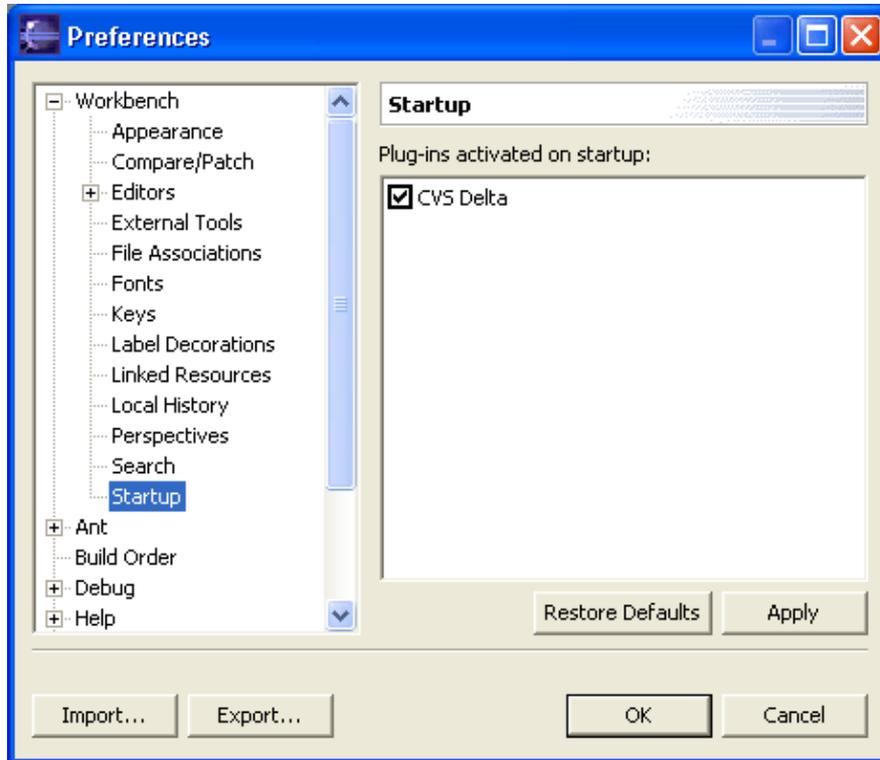
© Copyright IBM Corporation and others 2000, 2003.

# Startup

The Startup preference page allows the selection of plug-ins to be automatically activated during workbench startup.

Normally plug-ins are not activated until they are needed. However some plug-ins may specify that they wish to be activated during startup. This preference page allows the selection of which of these plug-ins will actually be activated during startup.

Here is what the Startup preference page looks like:



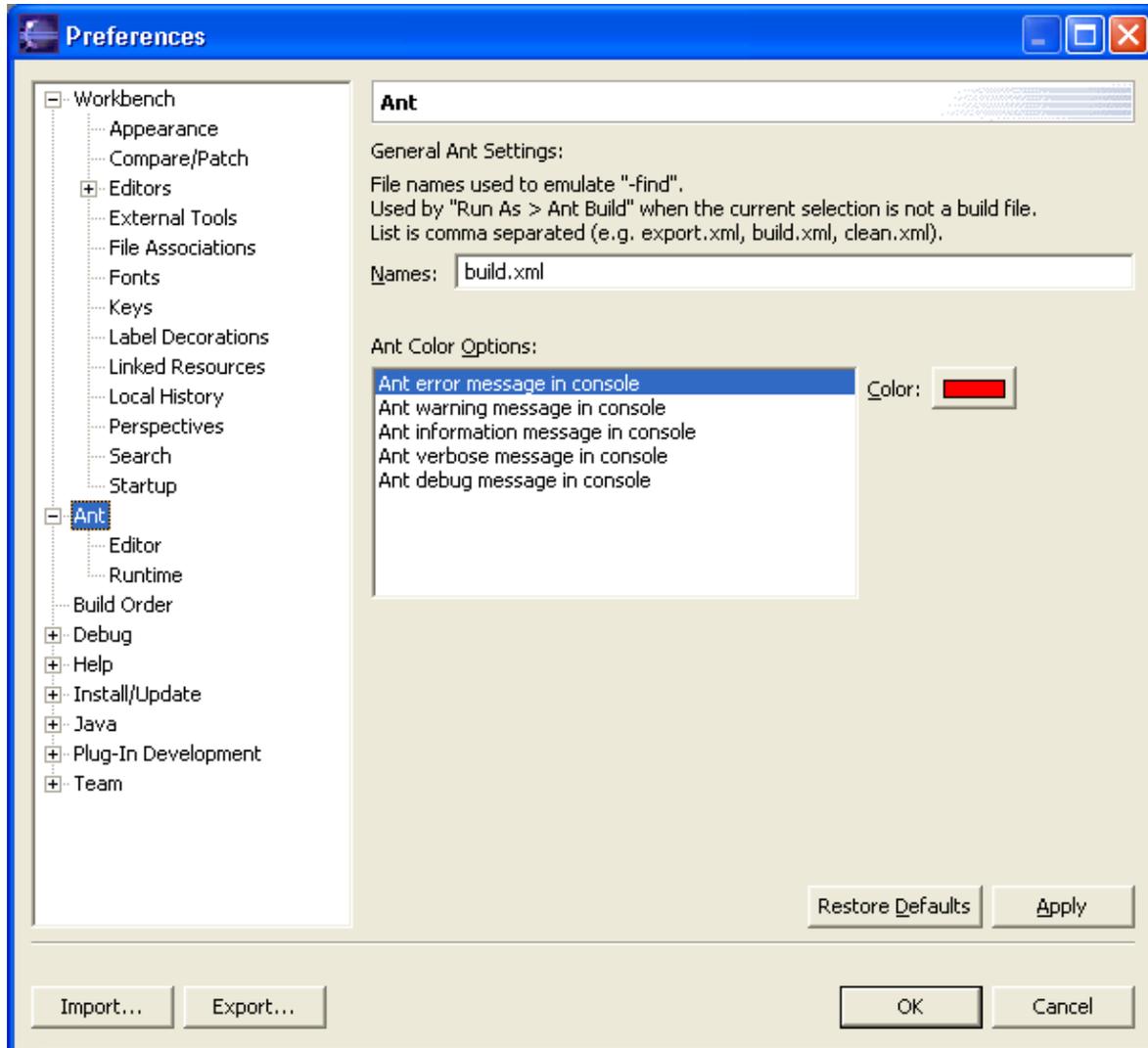
© Copyright IBM Corporation and others 2000, 2003.

# Ant

The following preferences can be changed on the Ant page.

You can configure the buildfiles that Ant's "-find" emulation will search for.

You can also configure the color of the Ant build output.



## ● Related concepts

[Ant support](#)

## ● Related reference

[Ant runtime preferences](#)

[Ant editor preferences](#)

## Crash recovery

© Copyright IBM Corporation and others 2000, 2003.

# Ant Editor

The following preferences can be changed on the Ant Editor page.

## Appearance options

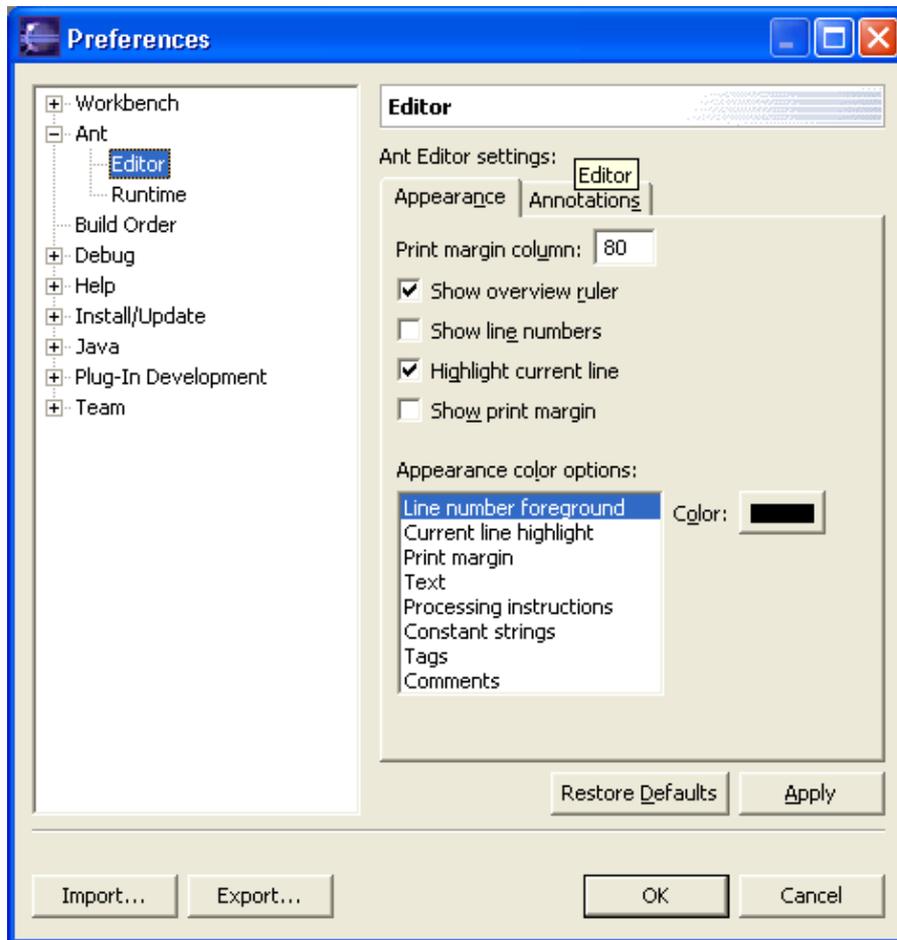
Option	Description	Default
Print margin column	This option allows to set the print margin for the Ant editor.	80
Show overview ruler	This option controls whether the overview ruler is shown on the right side of the Ant editor	On
Show line numbers	This option controls whether line numbers are shown in on the left side of the Ant editor.	Off
Highlight current line	This option controls whether the current line is highlighted or not.	On
Show print margin	This option controls whether the print margin is visible or not.	Off
Appearance color options	This option controls various appearance colors.	

## Annotations options

Option	Description
Show in text	This option controls whether the selected annotation type is shown in the text. The corresponding text will be underlined with squiggles.
Show in overview ruler	This option controls whether the selected annotation type is shown in the overview ruler.
Color	This option controls the color for the selected annotation type.

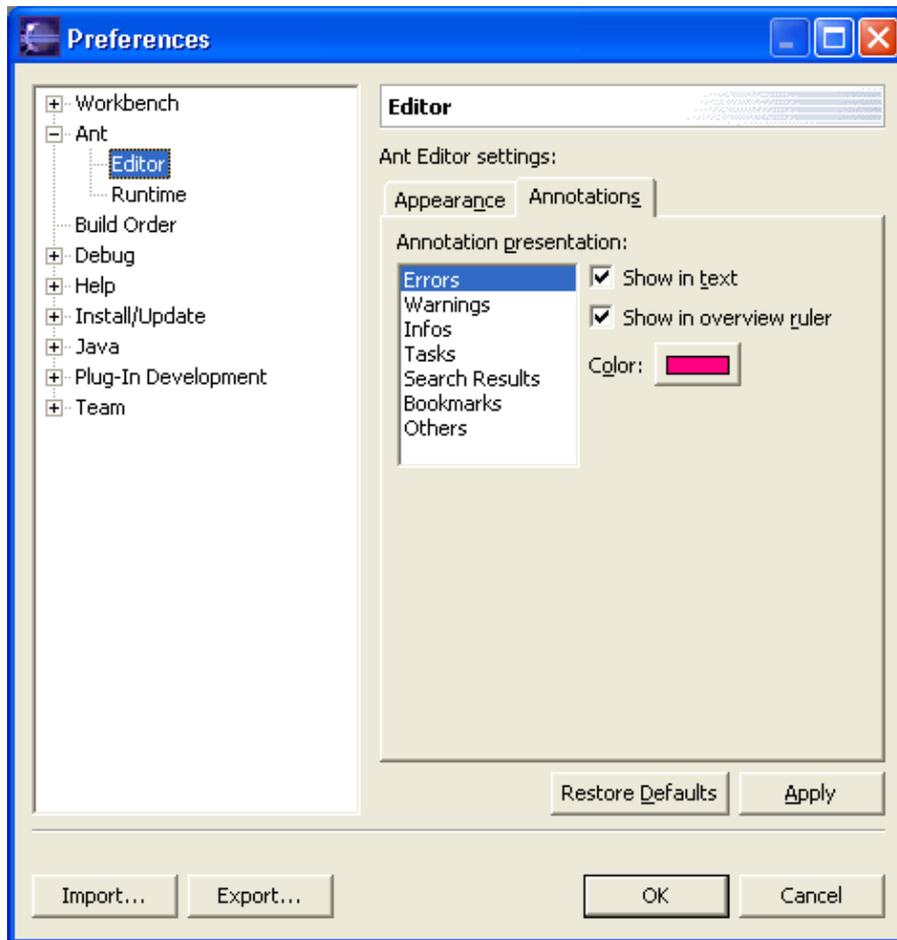
On the appearance page presents the appearance options for the Ant editor.

## Crash recovery



On the annotations page presents the annotations options for Ant editor.

## Crash recovery



### ● Related concepts

[Ant support](#)

### ● Related reference

[Ant Editor](#)

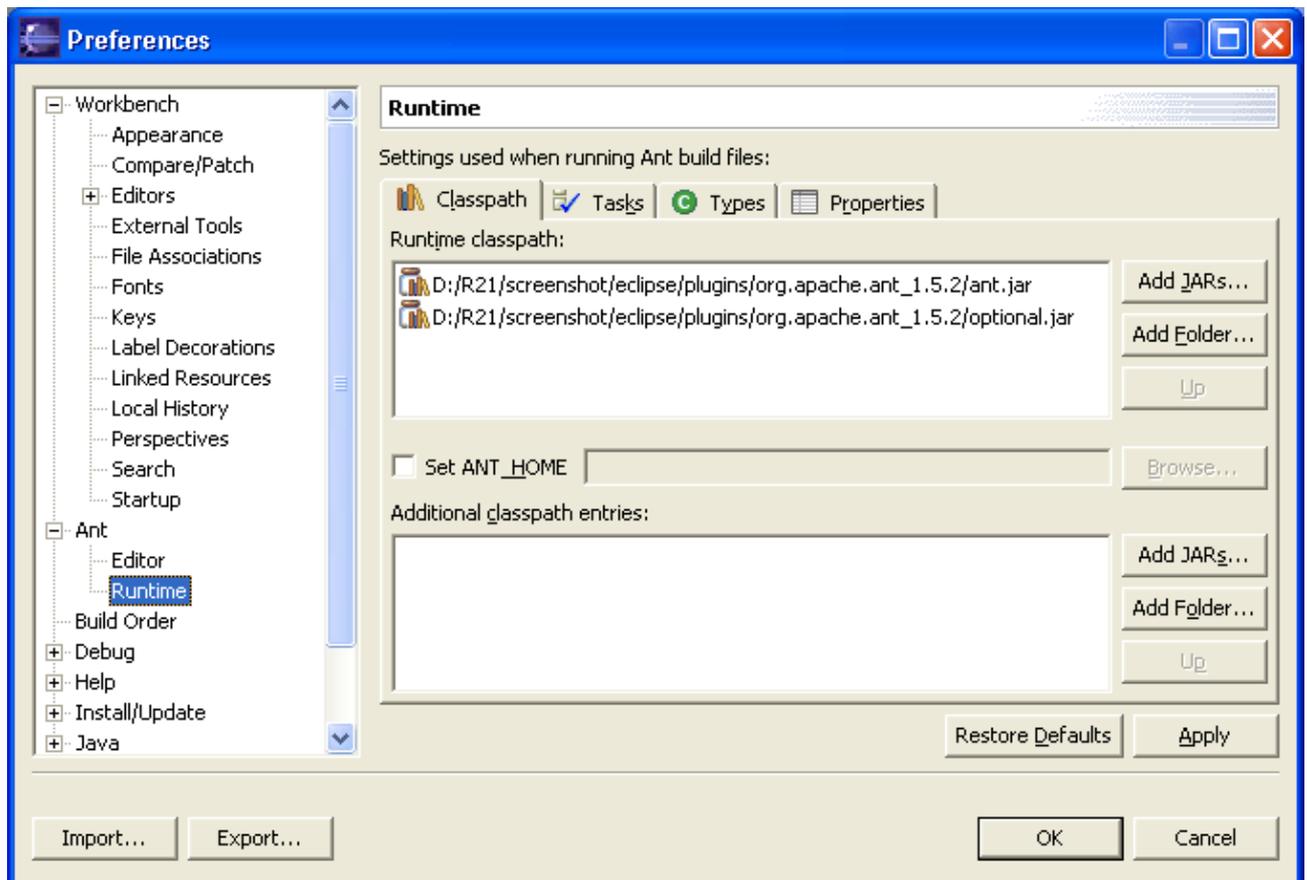
[Ant preferences](#)

[Ant runtime preferences](#)

© Copyright IBM Corporation and others 2000, 2003.

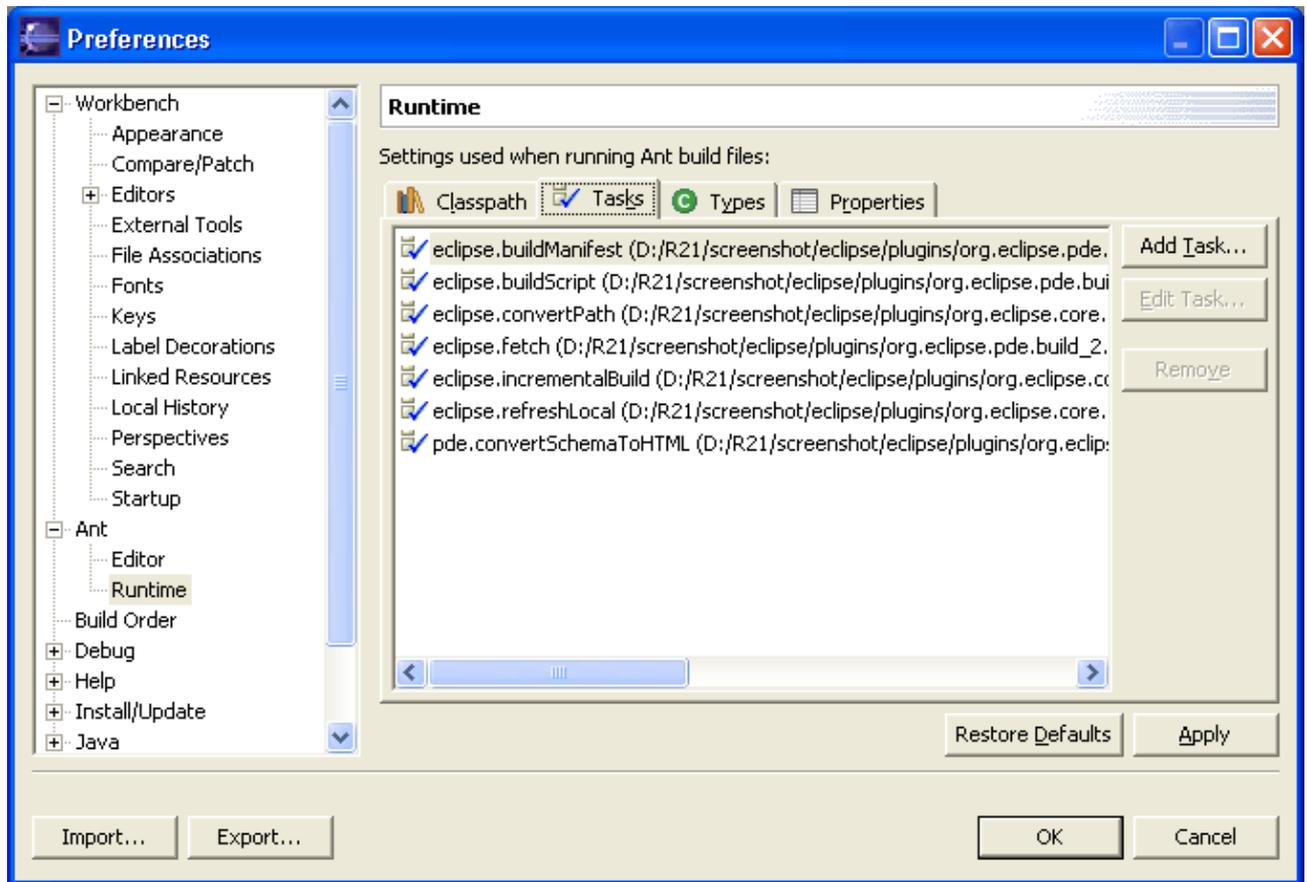
# Ant Runtime

On the classpath page you can add additional classes defining tasks and types to the Ant classpath.



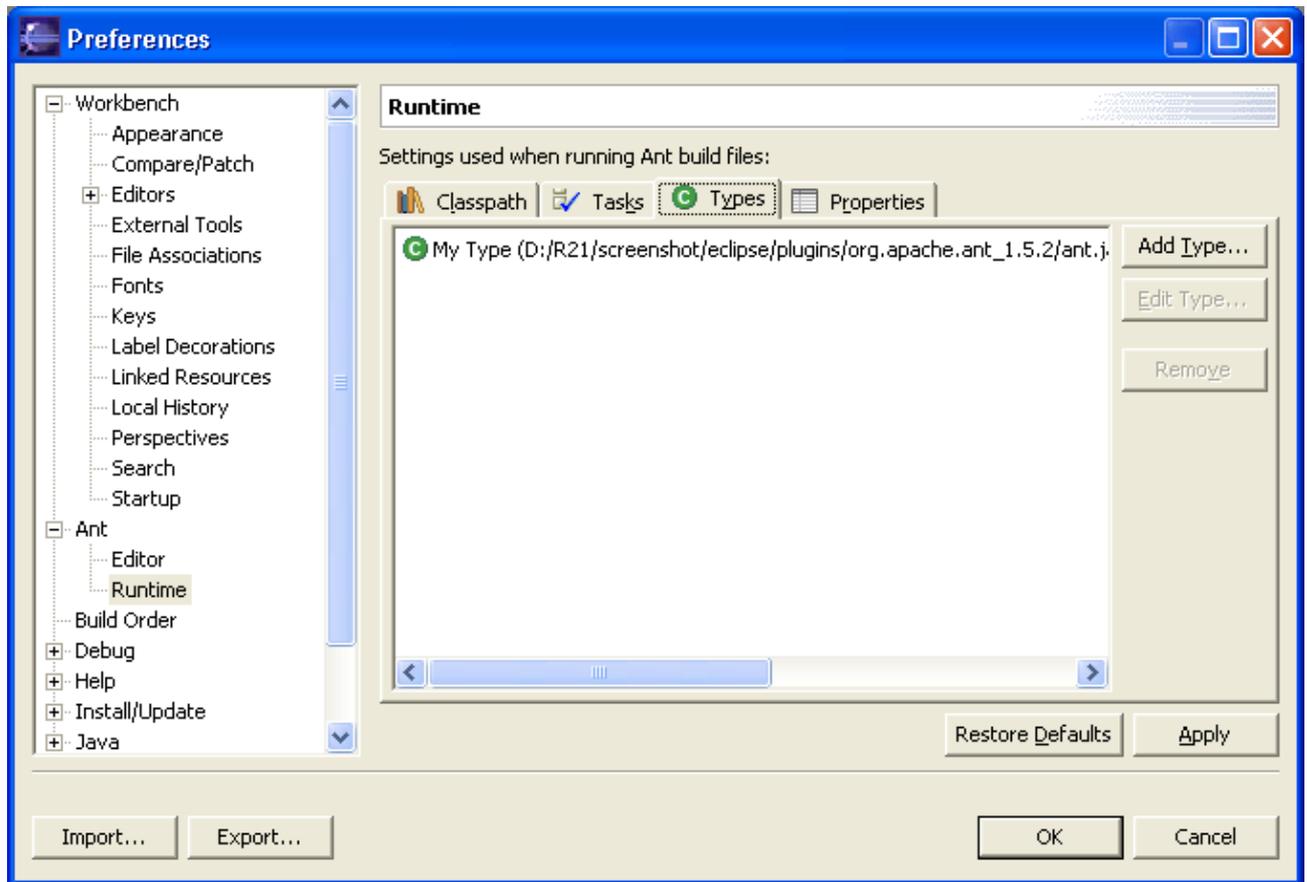
On the tasks page you can add tasks defined in one of the classes on the classpath.

## Crash recovery



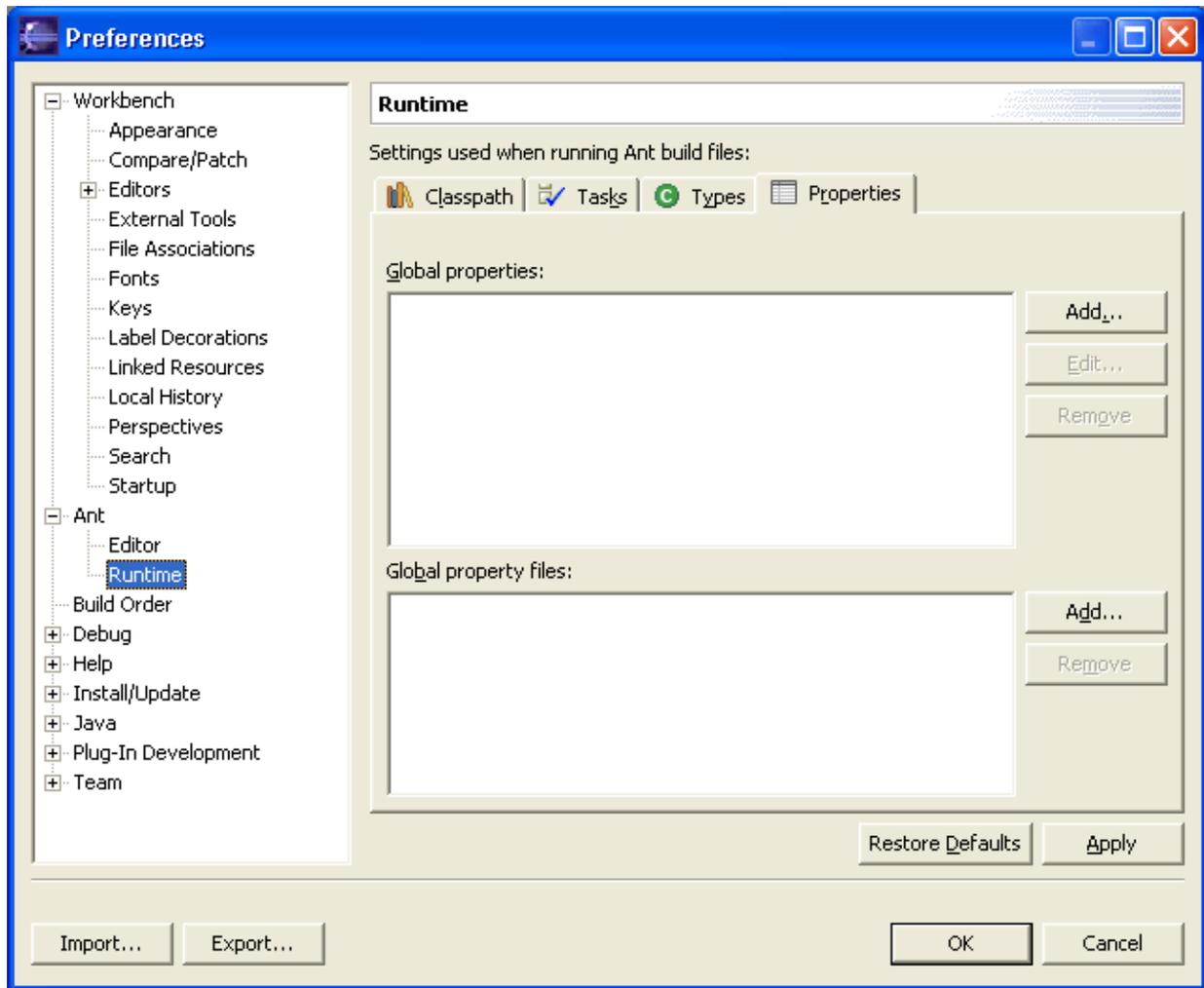
On the types page you can add types defined in one of the classes on the classpath.

## Crash recovery



On the properties page you can add properties and property files that will be passed into Ant.

## Crash recovery



### ● Related concepts

[Ant support](#)

### ● Related reference

[Ant preferences](#)

[Ant editor preferences](#)

© Copyright IBM Corporation and others 2000, 2003.

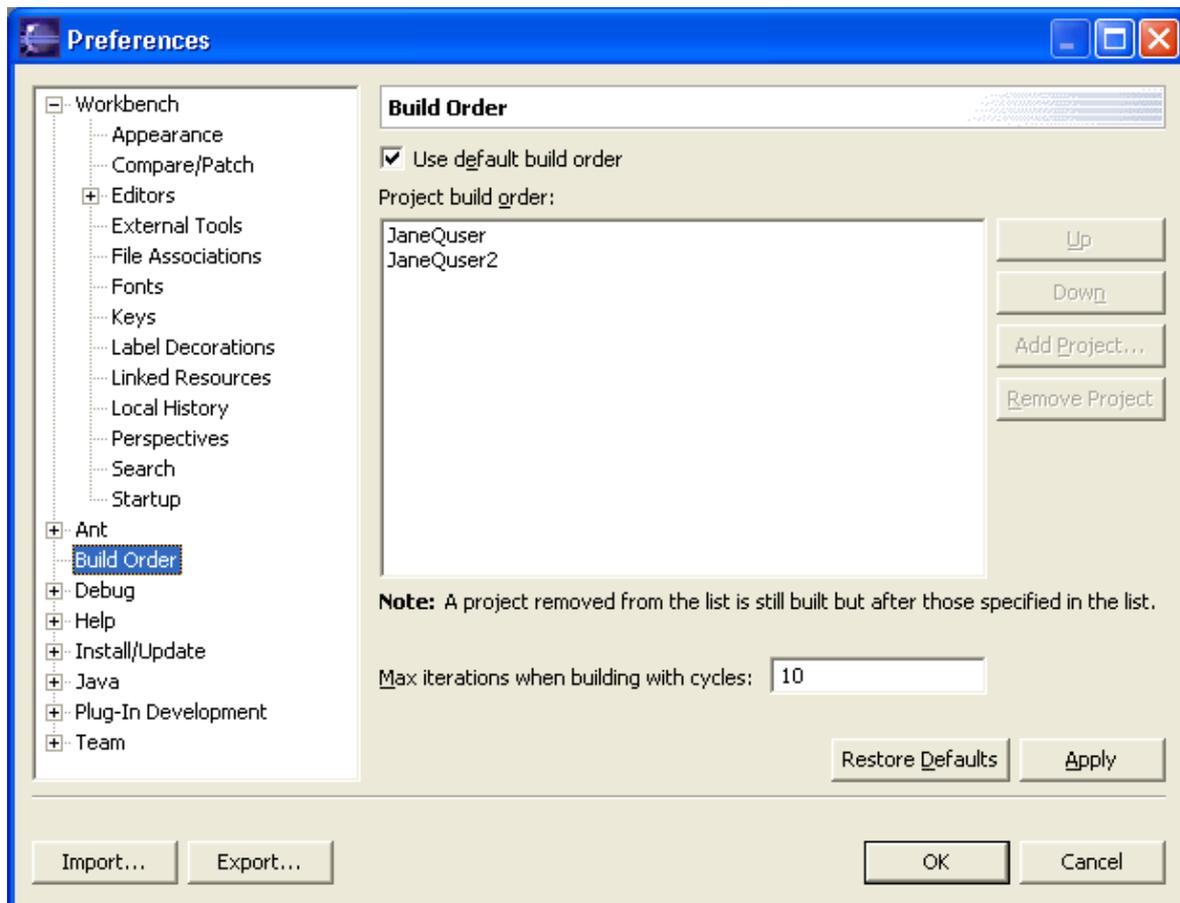
# Build Order

Often the order in which projects are built is important. For example, if one project requires the Java classes which were defined in another project, the first project must be built after its prerequisite classes have been built. The Workbench allows users to explicitly define the order in which projects are built. Alternatively, users can let the platform compute the build order by interpreting project references as prerequisite relationships. The build order is applied for both building the entire workspace or for a group of projects.

You can change this order on the Build Order preferences page. Initially the **Use default builder order** option is on, in which case the platform computes the build ordering. Turning off this option enables access to the projects list, the ordering of which can be manipulated. Select projects and use the **Up** and **Down** buttons to change the build order. Add and remove projects in the build order using the **Add Project** and **Remove Project** buttons. Projects removed from the build order *will* be built, but they will be built after all projects in the build order are built.

At the bottom of this page, there is a preference for dealing with build orders that contain cycles. Ideally, you should avoid cyclic references between projects. Projects with cycles really logically belong to a single project, and so they should be collapsed into a single project if possible. However, if you absolutely must have cycles, it may take several iterations of the build order to correctly build everything. Changing this preference will alter the maximum number of times the workbench will attempt to iterate over the build order before giving up.

Here is what the Build Order preference page looks like:



■ Related reference

[Builds](#)

[Project menu](#)

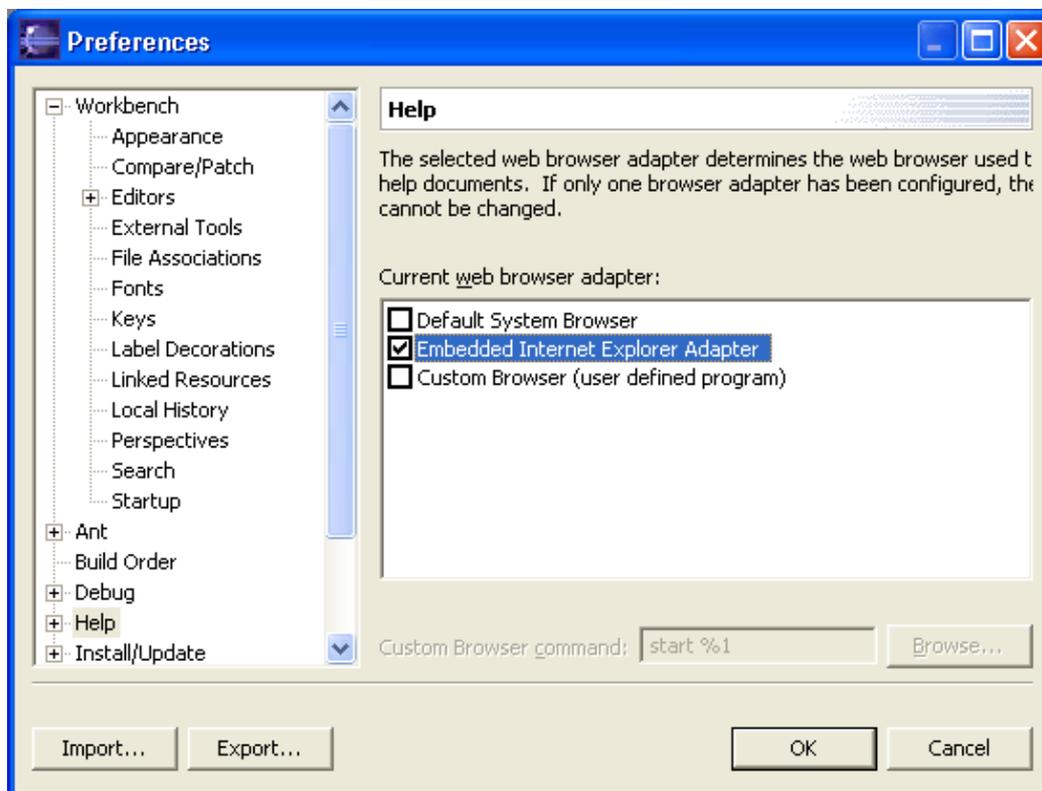
© Copyright IBM Corporation and others 2000, 2003.

# Help

On the Help preferences page, you can indicate how to display help books.

Option	Description	Default
Current web browser adapter	The help browser uses an adapter to display the on-line help in a web browser. If there are multiple adapters capable of opening a browser on your system, the one that is selected will be used to display help.	depends on operating system
Custom browser command	If "Custom Browser" is selected from the browser adapter list, this field is used to specify command that will launch the browser program. If URL is not the last parameter accepted by the browser program, use "%1" string to indicate position of the URL in the command.	mozilla %1 (the field is disabled)

Here is what the Help preferences page looks like:



**Note:** Selection performed on this page can affect how the help view is presented. If the selected browser is not fully compatible with Internet Explorer or Mozilla, or has JavaScript disabled, the help view shown in the browser might be a simplified version.

## Related tasks

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

## Related reference

[Help View](#)

## Crash recovery

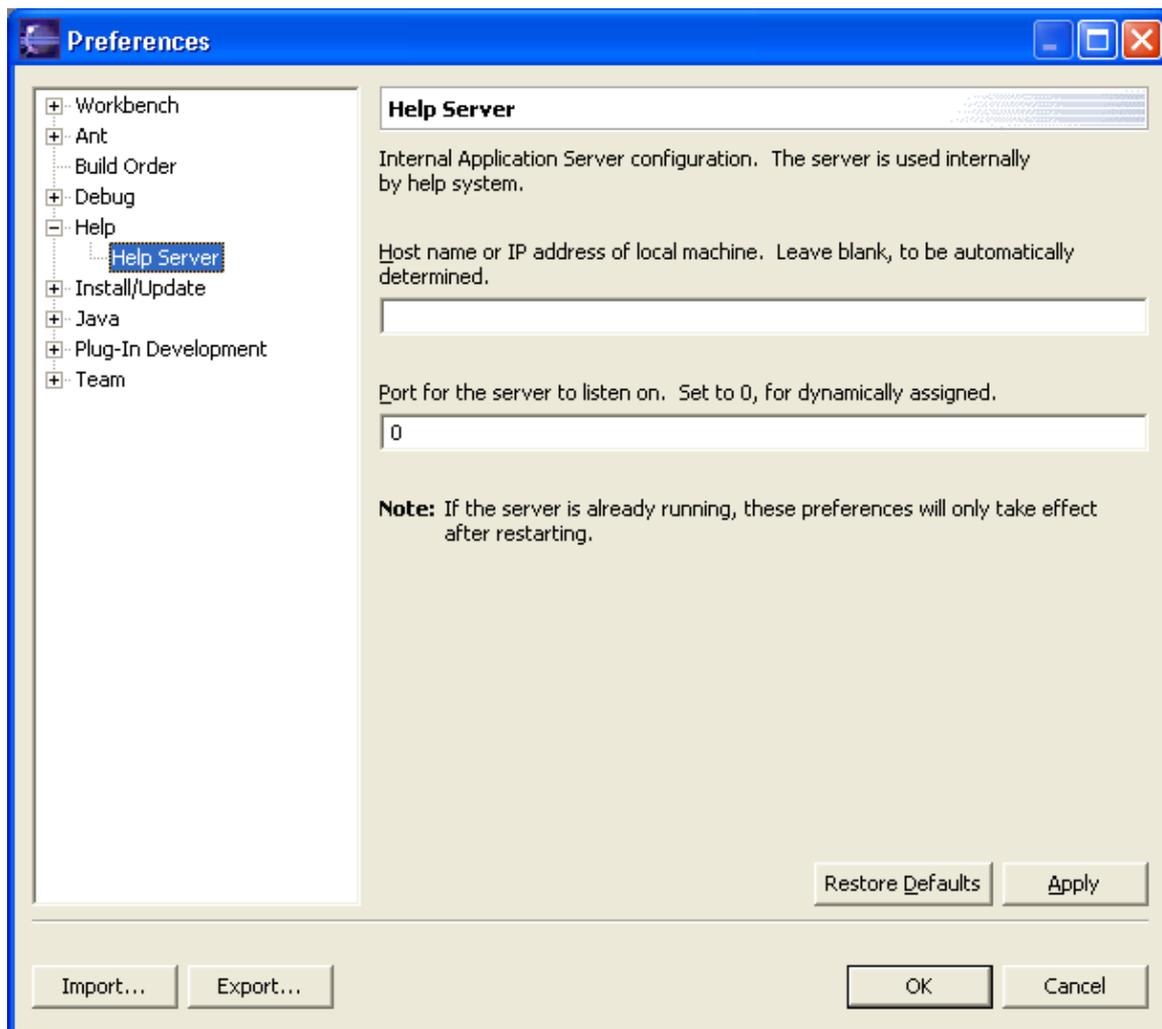
© Copyright IBM Corporation and others 2000, 2003.

# Help Server

Help system includes an internal server that serves help contents to the browser. Use this preference page to change the interface and port that the servers uses. You should only change these settings if you experience problems and cannot view help with the default preferences.

Option	Description	Default
Host	Name or address of a local IP interface to be used by the server.	blank
Port	IP port for server to listen on. If 0 is specified for the value, a port will be assigned by the operating system.	0

Here is what the Help Server preferences page looks like:



© Copyright IBM Corporation and others 2000, 2003.

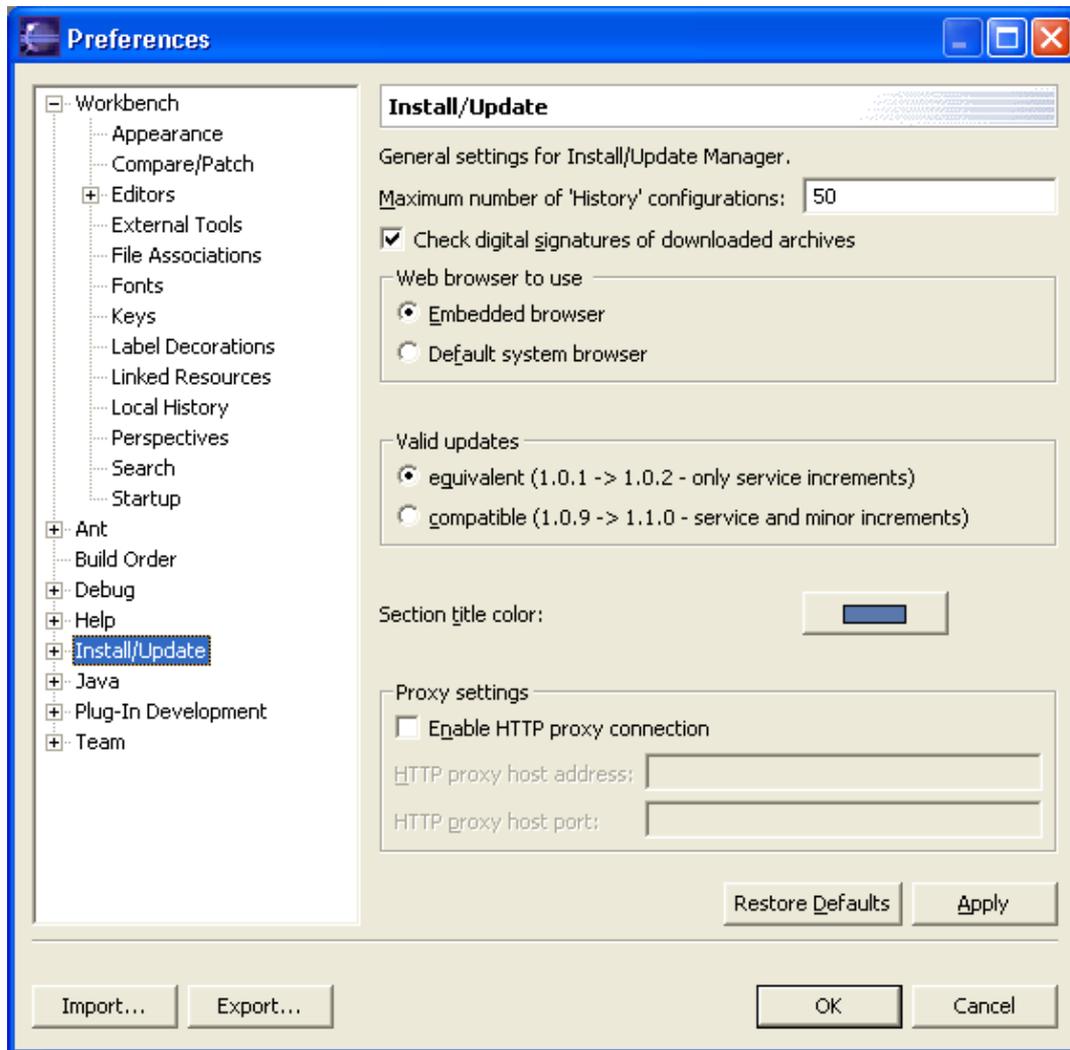
# Install/Update

The following preferences can be changed on the Install/Update page:

Option	Description	Default
Maximum number of 'History' configurations	Maximum number of configurations you want maintained in the configuration history. These configurations are maintained to allow you to revert to a previous configuration of installed feature versions.	50
Web browser to use	This setting determines whether links on install/update Web sites will be opened in the built in Eclipse browser or in your standard Web browser. You can select one of the following options: <ul style="list-style-type: none"><li>• Embedded Browser</li><li>• Default System Browser</li></ul>	Embedded Browser
Valid updates	Assuming that feature versions use the form 'major.minor.service', you can select what update level you want to choose from:  <i>equivalent</i> Only service updates will be displayed. <i>compatible</i> Service updates and minor updates will be displayed.	equivalent
Section title color	Color to use for section titles.	Blue
Proxy settings	Allows connection to the remote server from behind the firewall. When enabled, proxy host address (required) and port number (optional) can be specified.	No proxy

Here is what the Install/Update preference page looks like:

## Crash recovery



### ● Related tasks

[Updating Eclipse with the update manager](#)

### ● Related reference

[Help Menu](#)

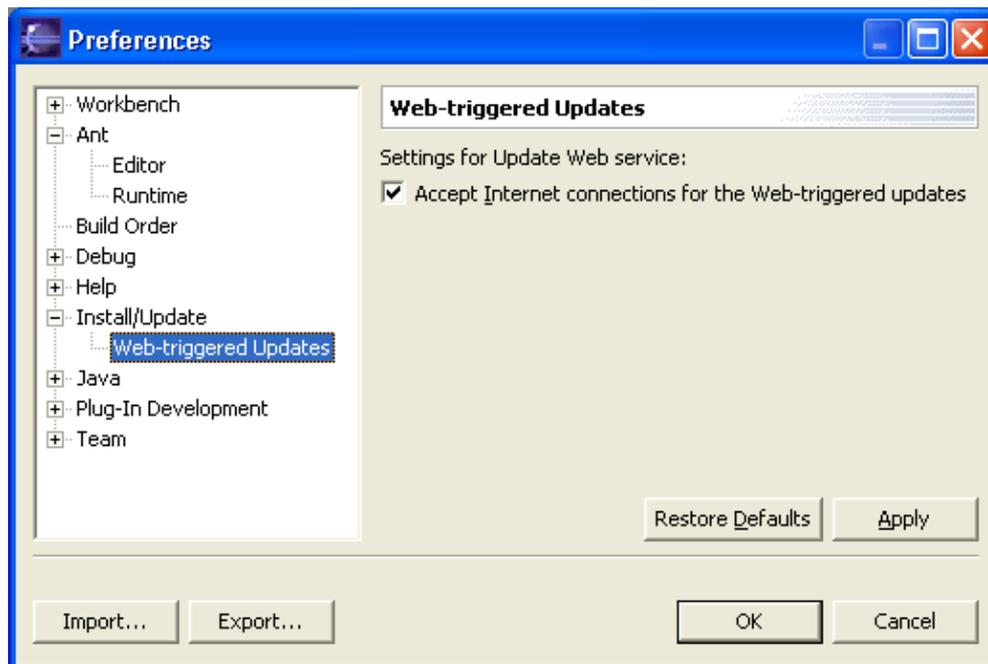
© Copyright IBM Corporation and others 2000, 2003.

# Web-triggered Updates

The following preferences can be changed on the Web-triggered Updates page:

Option	Description	Default
Accept Internet connections for the Web-triggered updates	When selected, Update manager will act as an application server and will be able to accept connections from Web pages capable of initiating install/update operations in Eclipse.	not selected

Here is what the Web-triggered Updates preference page looks like:



## ■ Related tasks

[Updating Eclipse with the update manager](#)

## ■ Related reference

[Help Menu](#)

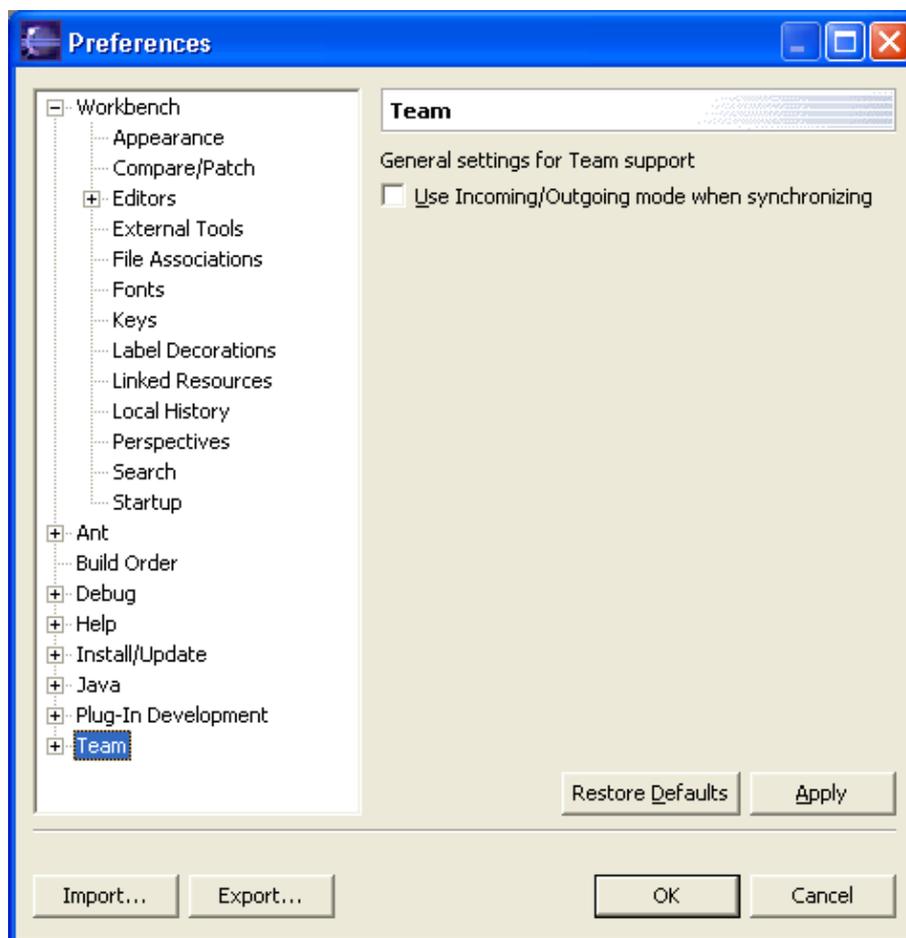
© Copyright IBM Corporation and others 2000, 2003.

# Team

The Team preferences page contains options which affect the version management Team support.

Option	Description	Default
Use Incoming/Outgoing mode when synchronizing	The Synchronize view can open in incoming mode, outgoing mode, or incoming/outgoing mode. Turning this option on will open the Synchronize view in incoming/outgoing mode, regardless of what you are doing (e.g., updating, committing, or both).	Off

Here is what the Team preferences page looks like:



## ■ Related reference

[CVS Synchronize view](#)

© Copyright IBM Corporation and others 2000, 2003.

# CVS

On the CVS preference page you can customize several aspects of the CVS Plug-in.

## General CVS Preferences:

Option	Description	Default
Prune empty directories	Use this option to specify the pruning of empty directories on update and in the synchronization view. Although pruned directories aren't shown in the workbench there is actually still an empty directory in the repository. This is helpful because CVS doesn't provide a client with the ability to remove directories from the server.	Enabled
Consider file contents in comparison	Use this option to set the default compare method for comparing and synchronizing CVS resources. By default, time stamps are used to compare CVS files, and this is by far the fastest method. However, in some cases a more accurate comparison can be achieved by comparing file content. This setting can also be toggled from within a compare editor or the synchronization view.	Disabled
Show deleted file markers	Use this option to enable markers to be added to the task view for resources that have been locally removed but not committed. This is useful for users who prefer update/commit instead of using the synchronize view. The deleted tasks allow quick fix options for committing the deletion to the server.	Enabled
Delete unmanaged resources on replace	Use this option to allows resources not under CVS control to be deleted when replacing with resources from the repository.	Enabled
Treat all new files as binary	Use this option to override the file content settings and treat all new files as binary.	Disabled
Validate server version compatibility on first connection	Use this option to enable a query of the CVS server version on the first connection to determine server compatibility. The server version will be output to the console and if an incompatibility is detected a warning message will be logged when connecting.	Enabled
Communication timeout	Use this option to configure the amount of time in seconds to wait before timing out from a connection to the CVS server.	60 seconds
Quietness level	Sets the amount of status information CVS prints for a command. In <b><i>Somewhat quiet</i></b> mode the printing of unimportant informational messages is suppressed. The consideration of what is important depends on each command. In <b><i>Very quiet</i></b> mode all output except what is absolutely necessary to complete the command is suppressed. In <b><i>Very Quiet</i></b> mode, some CVS servers may not communicate important information about errors that have occurred. You may want to consider using <b><i>Somewhat quiet</i></b> mode instead.	Verbose
Default keyword substitution	Use this option to set the default keyword substitution for text files.	ASCII with keyword expansion -kkv
Save dirty editors before CVS	Use this option to configure what happens when there are open editors with unsaved changes when a CVS operation is performed. Options are:	Prompt

## Crash recovery

operations	<p><b><i>Never</i></b></p> <p>Continue CVS operations even if there are unsaved changes in open editors.</p> <p><b><i>Prompt</i></b></p> <p>Ask the user what to do with unsaved changes in open editors.</p> <p><b><i>Auto-save</i></b></p> <p>Automatically save unsaved changes in open editors before each CVS operation.</p>	
------------	---	--

### Console preferences:

Option	Description
Console text color settings	<p>Use these options to change the colors for the text shown in CVS Console.</p> <ul style="list-style-type: none"> <li>• Command line text (<i>black</i>)</li> <li>• Message text (<i>blue</i>)</li> <li>• Error text (<i>red</i>)</li> </ul>

### Ext Connection Method preferences:

CVS_RSH	Use this option to configure the program that will be called to connect to the remote CVS server. The RSH command is invoked with following calling pattern:  <i>CVS_RSH Parameters CVS_SERVER</i>	ssh
Parameters	Use this option to configure the parameters passed to the CVS_RSH program. The default parameter pattern is {host} -l {user}. It can be tailored using the {host}, {user}, {password} and {port} variables.	{host} -l {user}
CVS_SERVER	Use this option to configure the name of the remote CVS server program to run. Change this setting only if the remote CVS server binary name is different than the default.	cvs

### Label Decorations preferences:

Text	Use the options on this page to configure how CVS information will be added to text labels.
Icons	Use the options on this page to configure the which icons can be used as overlays to show CVS specific information in views.
General	<p>Use the options on this page to configure general preferences about the decorators:</p> <p><b><i>Compute deep outgoing state for folders</i></b></p> <p>Use this option to configure if the outgoing indicators on folders should be calculated. Disabling this option improves the performance of the decorators because calculating the dirty state for folders requires computing the dirty state for all child resources. (<i>enabled by default</i>)</p>
Synchronize	Use the options on this page to configure how label decorations appear in the synchronize

## Crash recovery

view	<p>view:</p> <p><b><i>Show all CVS synchronization information in the text label</i></b></p> <p>Use this option to show the synchronization state of a resource in the text label. This is useful for accessibility requirements. If you are using a UI text reader, than enabling this option will allow the text reader to correctly interpret the states shown in the view. Ordinarily synchronization states are shown using icon decorations.</p>
------	--

## Watch/Edit preferences:

Configure projects to use Watch/edit on checkout	Use this option to indicate that files checked out from the repository should be made read-only.	disabled
When read-only files are modified in an editor	<p>Use this option to configure what occurs when a read-only file is modified in an open editor or by another tool.</p> <p><b><i>Send a cvs edit notification to the server</i></b></p> <p>Issues a cvs edit notification to the server before making the file writable. If other editors exist on the file, the user will be prompted and may continue or cancel.</p> <p><b><i>Edit the file without informing the server</i></b></p> <p>Makes the file read-only without notifying the server.</p>	Send a cvs edit notification to the server

### ■ Related reference

[Perspectives](#)

© Copyright IBM Corporation and others 2000, 2003.

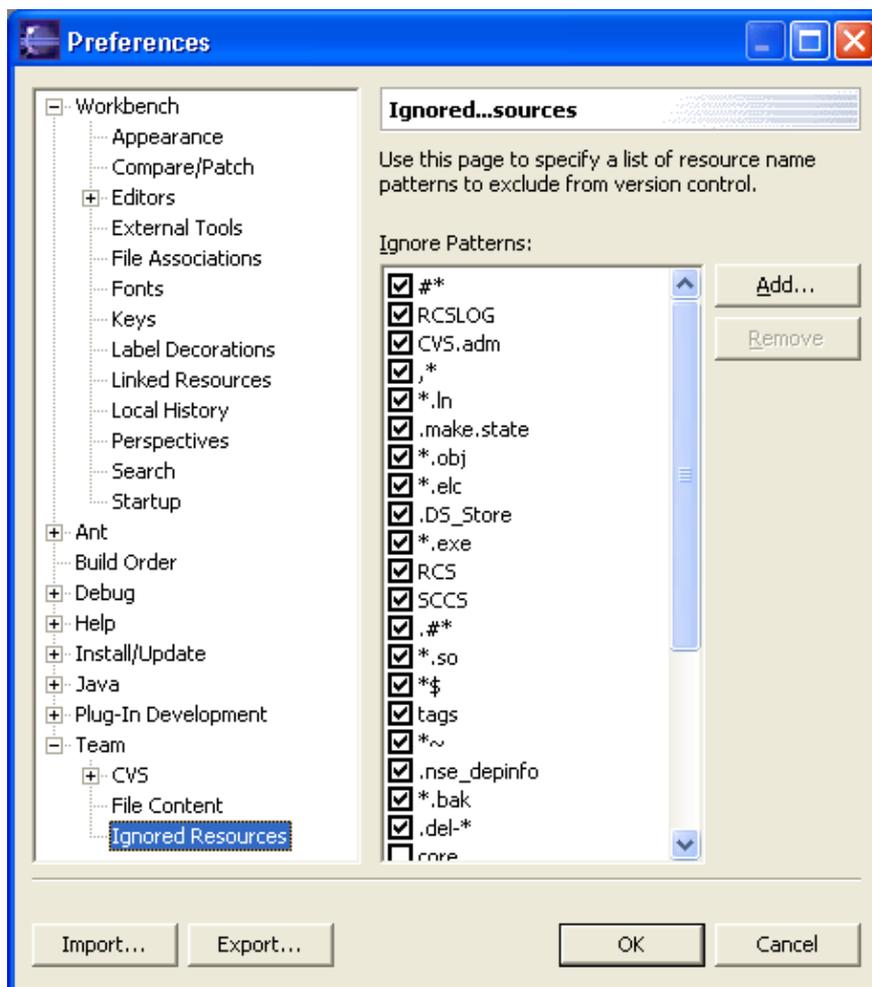
# Ignored Resources

On the Ignored Resources page, you can specify file name patterns to exclude from the version control management system. There is a list of file patterns against which resources will be matched before they are considered as version control candidates. These patterns may contain the wildcard characters "\*" and "?". This pattern "\*" represents any sequence of zero or more characters. This pattern "?" represents any one character. For example, you can specify a pattern of "\*~", which would match any temporary files that end with "~". Any file or directory that matches any one of the patterns will be ignored during update or commit operations.

To add a file type to the ignore list, simply click on the Add button. In the dialog, enter a file type (e.g. \*.class). To remove a file type from the ignore list, simply select the file type in the ignore list and click on the Remove button.

You can temporarily disable ignoring the file pattern by de-selecting it from the list. You do not have to remove the specified file pattern from the list to temporarily disable it.

Here is what the Ignored Resources preferences page looks like:



## ● Related reference

Ignoring Resources from Version Control

© Copyright IBM Corporation and others 2000, 2003.

# File Content

On the Team>File Content preference page, you can associate file extensions with the type of data the file contains. The two choices for file content type are ASCII and Binary. Repository providers such as CVS can then use this information to provide content type appropriate behavior. For example, for ASCII files, CVS ensures that line terminators conform to those of the OS platform.

Entries are added to the File Content page in two ways. The first is through contributions from workbench plug-ins. Tools integrated into the workbench provide the workbench with the file content types for file extensions specific to the tool. The workbench itself also defines the file content type for file extensions that are common and appear frequently in the workbench (e.g. html, gif, etc.).

The second method is for users to add file content types explicitly on the File Content preference page. To do so, the user can simply click on the **Add** button and enter an extension. Following this, they can toggle the type associated with the extension by selecting the entry for the extension in the table and clicking **Change**. Entries can be removed from the list by selecting them and clicking **Remove**.

## ■ Related tasks

[Setting the CVS keyword substitution mode](#)

© Copyright IBM Corporation and others 2000, 2003.

# CVS

The Workbench is shipped with a built in client for the Concurrent Versions System (CVS). With this client you can access CVS repositories.

You can find more information on CVS at <http://www.cvshome.org>.

You can also visit the [Eclipse CVS FAQ](#).

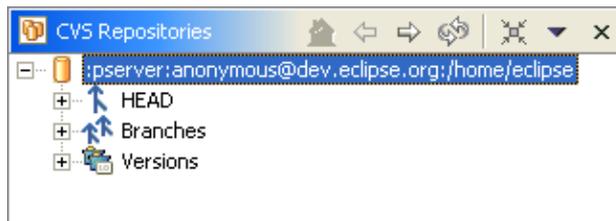
© Copyright IBM Corporation and others 2000, 2003.

# CVS Repositories view

The CVS Repositories view, which is part of the CVS Repository Exploring perspective, shows the CVS repository locations that you have added to your Workbench. Expanding a location reveals the main trunk (HEAD), project versions and branches in that repository. You can further expand the project versions and branches to reveal the folders and files contained within them.

The pop-up menu for this view also allows you to specify new repository locations. Use the CVS Repositories view to check out resources from the repository to the Workbench, configure the branches and versions shown by the view, view resource history, and compare resource versions.

Here is what the Repositories view looks like:



Like other views in the Workbench, the CVS Repositories view has its own toolbar. Toolbar buttons are provided for refreshing the view and for navigating. The toolbar also contains a drop-down menu that allows the user to filter the view by working set.

## Titlebar

### Go Home

This command redraws the view, showing the repository locations.

### Go Back

This command redraws the view, showing the hierarchy that was displayed immediately prior to the current display.

### Go Forward

This command redraws the view, showing the hierarchy that was displayed immediately after the current display.

### Refresh View

This command refreshes the contents of the Repositories view.

### Collapse All

This command collapses all expanded entries in the Repositories view.

## Drop–Down Menu

The drop down menu in the title bar allows the repositories view to be filtered by a working set.

## Context menu

From the context menu of the Repositories view you can perform a number of interesting operations.

### New > Repository Location

This command brings up the wizard to help you specify a new repository location.

### Configure Branches and Versions

This command brings up the wizard to help you discover the branch and version tags that exist in the repository for the selected folder so they can be added to the repositories view to allow the resources that have these tags can be browsed.

### Refresh Branches and Versions

This command, available on repository location entries, allows you to refresh the list of known branches and versions that are displayed in the repositories view for selected projects. This operation makes use of the defined auto–refresh files for each project. If the operation fails for a particular project, use Configure Branches and Versions on the project to select one or more appropriate refresh files.

### Add to Branch List...

This command adds the selected project to the list of projects that are displayed under the specified branch in the repositories view. This command only modifies the repositories view and does not effect the repository in any way. If you want to add the project to a branch, you can perform a *Tag with Existing* after performing this operation.

### Open

This command opens the selected file in an editor. Since file revisions in the repository are immutable, the editor opens in a read–only state, so it is not modifiable.

### Show in Resource History

This command shows the revision history of the selected file in the CVS Resource History view.

### Compare

This command will compare on two or three selected resources.

### Compare With...

This command will compare the selected folder with a branch or version of the same folder.

## Checkout as Project

This command checks the folder out into a project in the Workbench with the same name as the folder.

## Checkout as...

This command checks the folder out into a project in the Workbench into a project configured by the user.

## Checkout Into...

This command checks the folder out as a sub–folder of an existing project.

## Checkout Out Module

This command checks out the selected module into one or more projects in the workspace.

## Tag as Version...

This command versions the selected resource based on the current branch contents.

## Tag with Existing...

This command versions the selected resource based on the current branch contents, moving the tag from previously tagged resources if required.

### ■ Related concepts

[Team programming with CVS](#)

[CVS Repositories](#)

[Branches](#)

[Versions](#)

[Local history](#)

### ■ Related tasks

[Creating a CVS repository location](#)

[Discarding a CVS repository location](#)

[Refreshing the CVS repositories view](#)

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

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

[Replacing resources in the Workbench](#)

[Sharing a new project using CVS](#)

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

### ■ Related reference

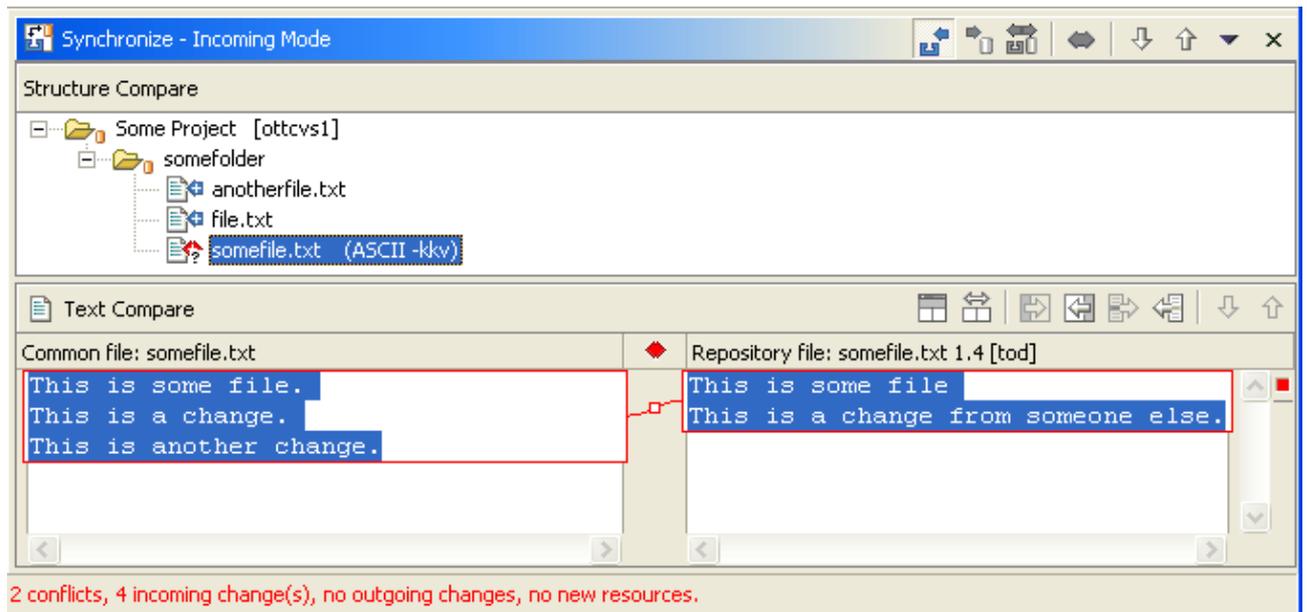
[CVS](#)

[CVS Resource History View](#)

# Synchronize view

This view allows you to update resources in the Workbench and commit resources from the Workbench to a repository.

Here is what the Synchronize view looks like:



## Modes

There are three different viewing modes (filters) that you can use in this view to make the synchronization process easier:

### *Incoming*

In this mode, only resources which have been changed in the repository since they were last loaded or synchronized with those in the Workbench (incoming changes) are visible. If you have incoming changes when you synchronize, you will start off in this mode.

### *Outgoing*

In this mode, only resources which have been modified in the Workbench (outgoing changes) are visible. If you do not have incoming changes when you synchronize, you will start in this mode.

### *Incoming/Outgoing*

In this mode both incoming and outgoing changes are shown, and you can both update and commit. You will remain in this mode with future synchronizations until you close the view. The advantage to using this dual mode is you can do either task as you choose. The disadvantage is that performing a simultaneous bi-directional merge is often complicated. In addition, you risk accidentally committing a conflict when you meant to update to it. You can choose to always use Incoming/Outgoing mode on the Workbench preferences page (**Window > Preferences > Team**).

Modes not only filter the set of visible changes but also restrict the set of operations available. You can switch modes using the toolbar of the Synchronization view.

The advantages to modes are:

## Crash recovery

- They support and encourage the ideal work flow.
- They reduce the amount of information you are faced with at any one time.
- They provide a degree of safety restricting you to those operations appropriate to that mode. For example, you can't accidentally commit a conflict when in incoming mode.

Regardless of the mode that you select, you will always see any conflicts that exist. You can filter out other outgoing and incoming changes by clicking the **Show Only Conflicts** button. You can also choose to Compare File Contents (and show only those resources whose contents differ), Ignore Whitespace differences or Refresh From Repository from the drop-down menu of this view.

**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.

## Synchronization state

The top left pane in the synchronize view shows the synchronization state of resources in your workspace compared to those in the repository. This state is shown by using icons and can also be configured to show the state as text appended to the resource name. A description of the icons is shown in the table below:

	An incoming addition means that a resource has been added to the repository. <i>Updating</i> will transfer the resource to your workspace.
	An incoming change means that the file has changed in the repository. <i>Updating</i> will transfer the new file revision to your workspace.
	An incoming deletion means that a resource was deleted from the server. <i>Updating</i> will delete your local resource.
	An outgoing addition means that the file was added to your workspace and is not yet in the repository. <i>Adding</i> then <i>Committing</i> will transfer the new file to the repository.
	An outgoing change means that the file was change locally. <i>Committing</i> the file will transfer the changes to the repository and create a new revision of the file.
	An outgoing deletion is a resource that has been deleted locally. <i>Committing</i> these resources will cause the remote resource to be deleted. <b>Note:</b> in CVS directories are never really deleted from the repository. Instead, files are deleted and empty directories are pruned from your workspace.
	A conflicting additions means that the resource has been added locally and remotely.
	A conflicting change means that the file has been changed locally and remotely. A manual or automatic merge will be required by the user.
	A conflicting deletion means that the resource was deleted locally and remotely.
	An auto-mergeable conflict is a conflict that can be automatically merged by CVS. This is because the file was changed locally and remotely but in different locations in the file. This indicator is important because it can allow you to review the changes from within the synchronization view and then perform a <i>Override and Update</i> that will perform the merge automatically for you.
	A non-added resource that has never been committed to the repository.

## Structure compare pane

This pane allows you to view a hierarchy tree of all the resource differences between the repository and the Workbench.

## Text compare pane

This pane allows you to view the content differences of the resource selected in the structure compare pane. You can choose to ignore whitespace differences from the view's drop-down menu.

### ■ Related concepts

[Team programming with CVS](#)

[Synchronizing with a CVS repository](#)

### ■ Related tasks

[Synchronizing with the repository](#)

[Updating](#)

[Resolving conflicts](#)

[Merging from a branch](#)

[Committing](#)

[Comparing resources](#)

[Merging changes in the compare editor](#)

### ■ Related reference

[CVS](#)

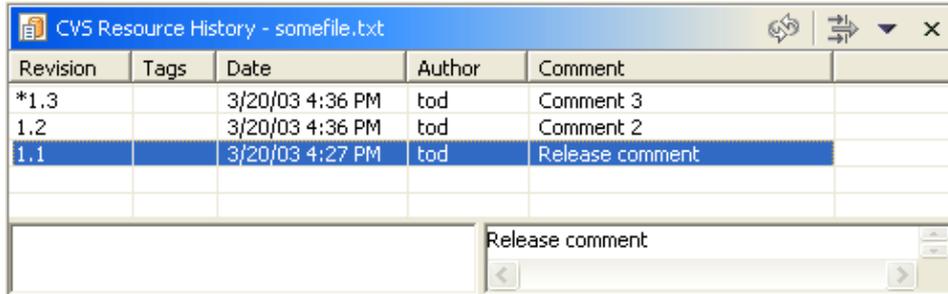
[Compare editor](#)

© Copyright IBM Corporation and others 2000, 2003.

# CVS Resource History view

This view provides a list of all the revisions of a resource in the repository. From this view you can compare two revisions, replace (get sticky) or revert (get contents) the corresponding workspace file to the revision, or open an editor on a revision.

Here is what the CVS Resource History View looks like:



The screenshot shows a window titled "CVS Resource History - somefile.txt". It contains a table with the following data:

Revision	Tags	Date	Author	Comment
*1.3		3/20/03 4:36 PM	tod	Comment 3
1.2		3/20/03 4:36 PM	tod	Comment 2
1.1		3/20/03 4:27 PM	tod	Release comment

Below the table, there is a text area displaying the comment "Release comment" for the selected revision (1.1).

For each resource, the following is displayed in the table:

## Revision

This column displays the revision number in the history.

## Tags

The tags that are associated with the revision. Selecting a revision line will list the tags in the lower left pane of the view.

## Date

This column displays the creation date and time of the revision in the history.

## Author

This column displays the name of the person who created and authored the version.

## Comment

This column displays the comment (if any) supplied for this revision at the time it was committed. Selecting a revision line will show the complete comment in the lower right pane of the view.

### ■ Related reference

[CVS Repositories view](#)

### ■ Related concepts

[Resources](#)

CVS Repositories

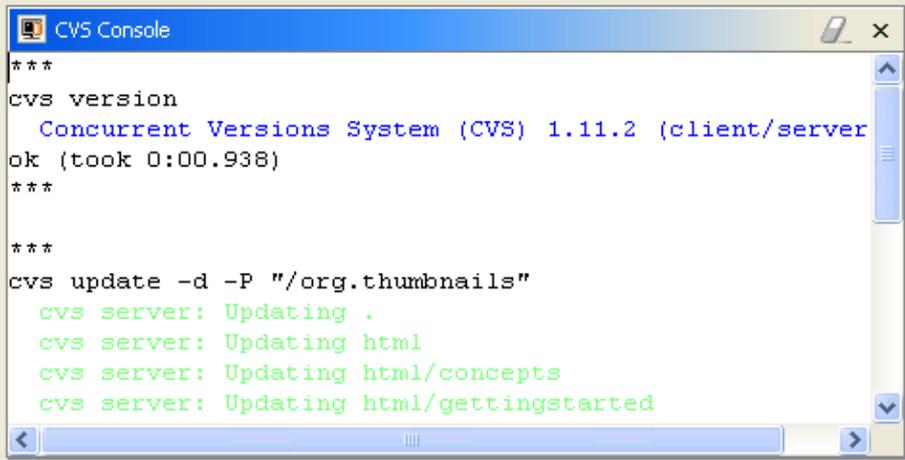
Local history

© Copyright IBM Corporation and others 2000, 2003.

# CVS Console view

This view shows the output from CVS commands. This output is similar to that of the CVS command line client. If an error occurs during a CVS operation opening the console can help troubleshoot the cause of the error.

Here is what the CVS Console view looks like:

A screenshot of a window titled "CVS Console". The window contains the following text:

```
***  
cvs version  
  Concurrent Versions System (CVS) 1.11.2 (client/server)  
ok (took 0:00.938)  
***  
  
***  
cvs update -d -P "/org.thumbnails"  
  cvs server: Updating .  
  cvs server: Updating html  
  cvs server: Updating html/concepts  
  cvs server: Updating html/gettingstarted
```

The window has a blue title bar and a scroll bar on the right side.

## ■ Related reference

[CVS Repositories view](#)

## ■ Related concepts

[Resources](#)

[CVS Repositories](#)

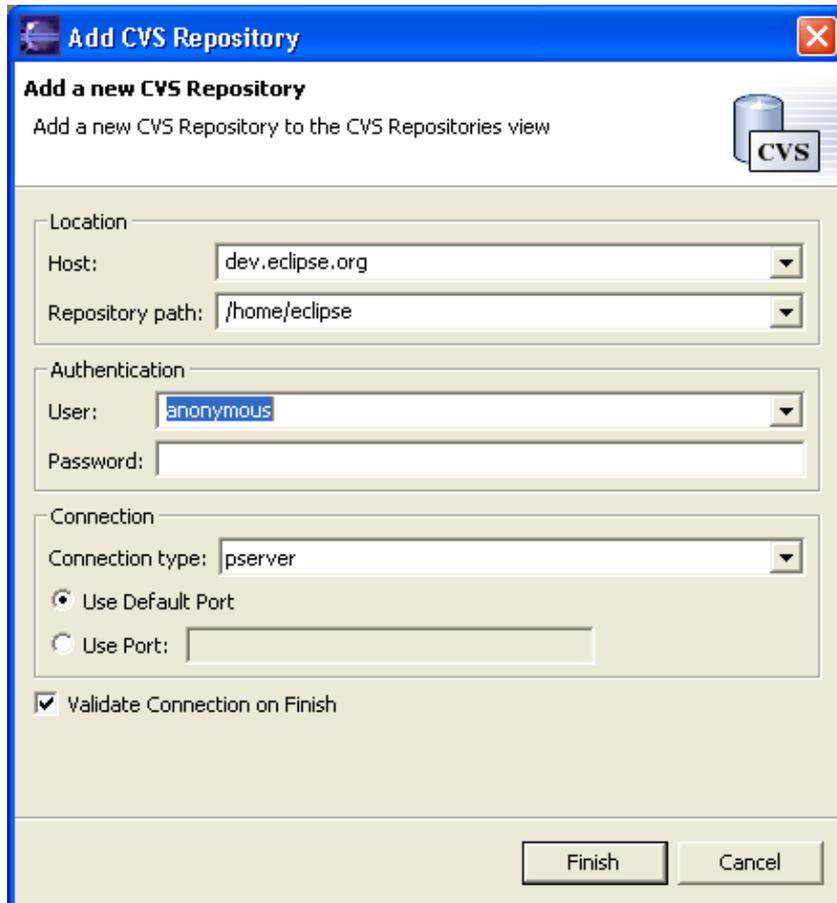
[Local history](#)

© Copyright IBM Corporation and others 2000, 2003.

# New CVS Repository Location wizard

This wizard helps you create a repository location.

Here is what the New CVS Repository Location wizard looks like:



The screenshot shows a dialog box titled "Add CVS Repository" with a close button in the top right corner. The main title is "Add a new CVS Repository" and the subtitle is "Add a new CVS Repository to the CVS Repositories view". There is a CVS logo icon in the top right. The dialog is divided into three sections: "Location", "Authentication", and "Connection".

**Location:**

- Host:
- Repository path:

**Authentication:**

- User:
- Password:

**Connection:**

- Connection type:
- Use Default Port
- Use Port:

Validate Connection on Finish

Buttons:

# CVS repository location

The following are the editable fields for a new CVS repository location:

Option	Description	Default
Host	The address of the host (e.g., "hostmachine.com").	<blank>
Repository Path	The path to the repository on the host (e.g., "/x/y/z", "d:/myrepo").	<blank>
User Name	The user name under which you want to connect to the server.	<blank>
Password	The password required for the above user name to access the above host.	
Connection Type	The type of CVS connection for the repository, either "pserver", "extssh" or "ext".	pserver
Port	Option to use a custom port for the connection.	Use Default Port
Validate connection on finish	Check this option if you want to attempt a connection to the host upon completion of the wizard to validate that the entered information is correct. If this option is not enabled, you will not know whether the information entered is correct until you try to access the contents of a repository for the first time.	On

## ■ Related reference

[CVS Repositories View](#)

© Copyright IBM Corporation and others 2000, 2003.

# Merge wizard

This wizard helps you merge changes between two states of a project into your workspace. Often the Merge Wizard is used to move changes from one branch into another, for example after splitting a branch to work on a bug fix. The merge operation takes changes between two points in a branch, the **start point** and the **end point**, and merges them into your workspace. Typically the start point will be the root of a branch (version tag) and the end point can either be the tip (latest and best) of the branch or another version tag.

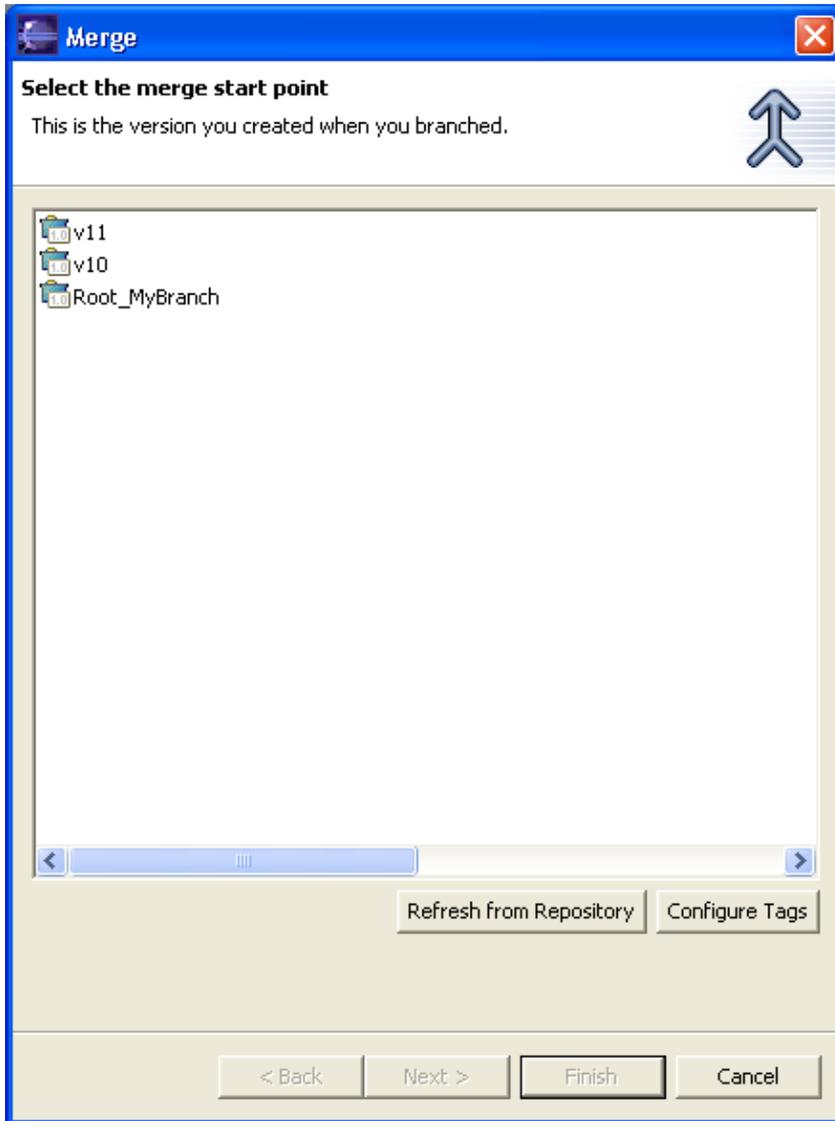
It is very important to understand that the destination of the merge is always the project in your workspace. After the merge has been completed you can test the changes locally and then commit them to the new branch (usually HEAD).

To start a merge, select a project (or one or more resources), and select **Team > Merge...** from the pop-up menu.

## Start Point

This is where you will select the starting point of the merge.

From the list, select a version of the project that will be the starting (base) point of the merge operation.



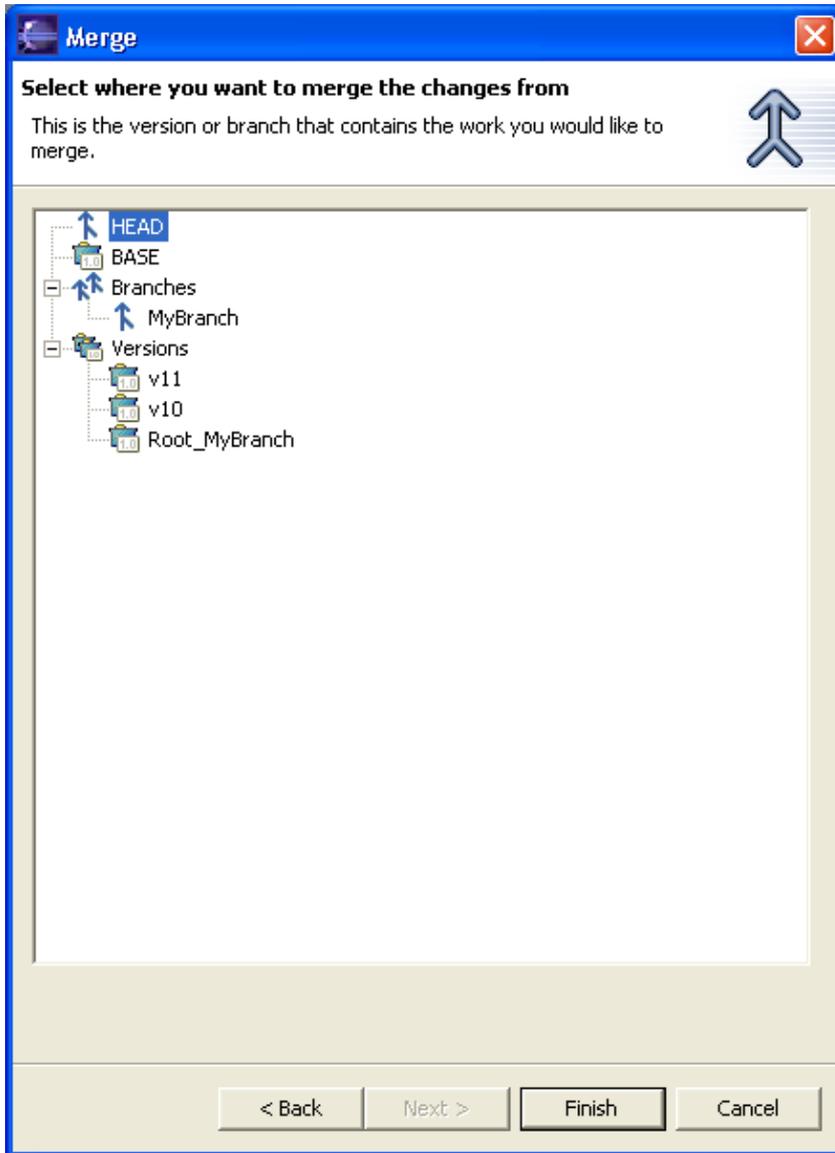
## End state

This is where you will choose the end state of the merge.

From the list, select either:

- A version of the project
- The branch

This will be the end point of the merge operation. Choose a version when you want to merge the differences between two versions of a project into your workspace. Choose a branch if you want to merge the changes made in the branch into your local workspace.



After the finish button is pressed the changes between the start point and end point are calculated and a merge editor is displayed. Depending on the size of the project you are merging, this may take some time. In the merge editor you can update or merge changes into your Workbench. You cannot commit from the merge view.

#### ■ Related concepts

[Team programming with CVS](#)

[Branches](#)

#### ■ Related tasks

[Branching](#)

[Merging from a branch](#)

#### ■ Related reference

[CVS](#)

Compare editor

© Copyright IBM Corporation and others 2000, 2003.

# Ignoring resources from version control

When synchronizing resources, it is possible that there are some resources that you do not want to commit to the repository. There are two ignore facilities provided, allowing the user to specify which resources should be excluded from update and commit operations.

The first is a global ignore facility, provided by the Workbench. The second is the CVS ignore facility, which reads the contents of a special file `.cvsignore` to determine what to ignore.

## Why ignore files when synchronizing?

There are many files that a user may not want to commit to the repository. For example, external editors may create temporary files in your project. Compilation of `.java` files creates `.class` files, and likewise many build operations result in binary files. These files, when taken together, may be quite large. In addition, they may be re-generated whenever a build is performed, resulting in many outgoing changes. Typically these are not files that one wants to share with other members of a team or persist in the repository.

## Global ignore facility

A global ignore facility is provided by the Workbench via the Team preference page. There is a list of file patterns against which resources will be matched before they are considered as version control candidates. These patterns may contain the wildcard characters "\*" and "?". "\*" represents any sequence of zero or more characters. "?" represents any one character. For example, you can specify a pattern of "\*~", which would match any temporary files that end with "~". Any file or directory that matches any one of the patterns will be ignored during update or commit operations. When you specify a file pattern to ignore, you can temporarily disable ignoring the file pattern by de-selecting it from the list; you do not have to remove the specified file pattern from the list.

The patterns in the global ignore facility are matched against resource names during a synchronize operation. 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.

## CVS ignore facility

The Eclipse CVS client recognizes a file named `".cvsignore"` in each directory of a project. This is a standard CVS facility and many existing CVS projects may contain this file.

This text 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. It is important to note that the semantics of this file differs from that of the global ignore facility in that it applies only to files and directories in the same directory as the `.cvsignore` file itself. A project may contain one `.cvsignore` file in each directory. For more information, please visit <http://www.cvshome.org>.

Resources that have not been added to CVS control can be ignored by selecting **Team > Add to .cvsignore** from the pop-up menu of the resource in the navigator. This menu option is also available in the Synchronize view.

## Crash recovery

© Copyright IBM Corporation and others 2000, 2003.

# CVS Label Decorations

Label Decorations are used by CVS to show important synchronization information about workspace resources. Decorations can effect the text or the icon of the label associated with a resource. The text decorators are configurable on the [CVS Label Decorations Preference page](#) which can also be used to indicate which icons are to be enabled. Here is a description of the icons used by CVS.

<i>Decoration</i>	<i>Resource type</i>	<i>Description</i>
	Any	Indicates that the resource is under version control.
	Any	Indicates that the resource or one of its children contains outgoing changes.
	File	Indicates that the project containing the file has been configured to use watch/edit and the file is being edited locally.
	File	Indicates that the file contains a merge conflict which has not yet been resolved. The conflict is considered resolved once the file is modified and saved.
	Project or Folder	Indicates that the folder is under version control but does not correspond to an existing remote directory but is instead most likely a module defined in the CVSROOT/modules file.
	File or Folder	Indicates that the resource is not under version control.

*Note:* Resources that are ignored by CVS will not be decorated by any of the above CVS decorations.

## ■ Related concepts

[Label Decorations](#)

## ■ Related tasks

[Enabling the CVS resource decorations](#)

© Copyright IBM Corporation and others 2000, 2003.

# Workbench toolbar

The Workbench toolbar is displayed at the top of the Workbench window, directly underneath the menu bar. The contents of the toolbar change based on the active editor. Actions in the toolbar may apply to particular views, so these actions may be enabled or disabled based on the state of the currently active view or editor.

Here is an example of the toolbar in the Resource perspective:



## New Wizard

This command brings up a dialog where you can choose the type of resource to create.

## Save The Open Editor Contents

This command saves the file currently displayed in the editor area.

## Save To Another Location

This command saves a copy of the file currently displayed in the editor area to a user–specified location. The specified name and location can be different from the file's original name and location.

## Print

This command opens a dialog which allows you to specify where you would like to print the contents of the file currently being displayed in the editor.

## Search

This command opens the search dialog, which allows you to search the workspace for specified text.

## External Tools

This command presents a drop–down menu which allows you to run or configure external tools.

### ■ Related reference

[Shortcut Bar](#)

[View Toolbars](#)

© Copyright IBM Corporation and others 2000, 2003.

# Shortcut bar

The shortcut bar appears along the left edge of the Workbench window. This toolbar is always active; it cannot be closed or hidden.

The shortcut bar allows quick access to perspectives that are currently open, as well as providing an easy way to open a new perspective. The shortcut bar is also the place where fast views are created and opened.

Here is an example of what the shortcut bar looks like:



## Open Perspective

This command opens a new perspective which is selected by the user from a drop-down menu. All of the perspectives that are open within a single Workbench window are shown on the shortcut bar.

## Perspective Buttons

These buttons provide a quick way to switch to one of the open perspectives in the current Workbench window.

## View Buttons

These buttons provide a quick way to display the fast views in the current perspective. Fast views are essentially minimized views that have been dragged onto the shortcut bar. Fast views pop up when selected, and revert back to their minimized state when the user clicks outside of the view. To convert a fast view back into a normal view, click on the *Fast View* action in the view's toolbar.

### ■ Related reference

[Perspectives](#)

[View Toolbars](#)

[Workbench Toolbar](#)

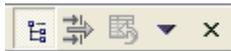
[Workbench Window Layout](#)

© Copyright IBM Corporation and others 2000, 2003.

# View toolbars

View toolbars contain actions that apply only to the particular view in which they appear. The view toolbar also contains a context menu that contains other actions for that view. This menu is opened by clicking on the downwards pointing triangle.

Here is an example of the view toolbar for the Properties view:



## Close View

This action closes the view.

## Title Bar

View title bars contain the view name, its icon, and the view toolbar.

### ■ Related reference

[Shortcut Bar](#)

[Workbench Toolbar](#)

[Workbench Window Layout](#)

© Copyright IBM Corporation and others 2000, 2003.

# Builds

Builders create or modify workspace resources, usually based on the existence and state of other resources. They are a powerful mechanism for enforcing the constraints of some domain. For example, a Java builder converts Java source files (.java files) into executable class files (.class files), a web link builder updates links to files whose name/location have changed, etc. As resources are created and modified, builders are run and the constraints are maintained. This transform need not be one to one. For example, a single .java file can produce several .class files.

## Auto-build vs. Manual Build

There are two distinct user work modes with respect to building: relying on Auto-build, or user initiated manual building.

Users who do not need fine-grained control over when builds occur may simply choose to turn on auto-building. With auto-building on, builds occur after every set of resource changes (e.g., saving a file, importing a ZIP, ...). Auto-building is efficient because the amount of work done is proportional to the amount of change done. The benefit of auto-building is that your derived resources (e.g., Java .class files) are always up to date. Auto-building is turned on/off via the *Perform build automatically on resource modification* option on the Window > Preferences > Workbench page.

Users needing more control over when builds occur can turn off auto-building and manually invoke builds. This is sometimes desirable in cases where, for example, you know building is of no value until you finish a large set of changes. In this case there is no benefit to paying the cost of auto-building. Builds can be invoked manually in numerous ways, for example, by selecting Rebuild All from a project's context menu.

The disadvantage of manual building is that the tasks which were 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 (e.g. before running your Java program).

## Full vs. Incremental Builds

There are two kinds of builds: full and incremental. An incremental build works with a previous built state and applies the transforms of the configured builders only on the resources which have changed since that previous state was computed (i.e., since the last build). Auto-building uses incremental building for efficiency.

A full build (a.k.a. *rebuild*) discards any existing built state and transforms all requested resources according to the domain rules of the configured builders. The first incremental build is equivalent to a full build as there is no previous state to work from.

Depending on the user's needs, full and incremental builds can be done over a specific set of projects or the workspace as a whole. Specific files and folders cannot be built.

# Perspectives

A perspective defines the initial set and layout of views in the Workbench window. One or more perspectives can exist in a single Workbench window.

Perspectives can be opened either in the same (existing) Workbench window, hiding the current perspective, or in a new Workbench window

Perspectives define visible action sets, which you can change to customize a perspective. You can save a perspective that you build in this manner, making your own custom perspective that you can open again later.

The Workbench defines the *Resource* perspective by default. This perspective shows views relevant to resource management.

## ■ Related reference

[Window Menu](#)

[Editor Area](#)

[Navigator View](#)

[Outline View](#)

[Tasks View](#)

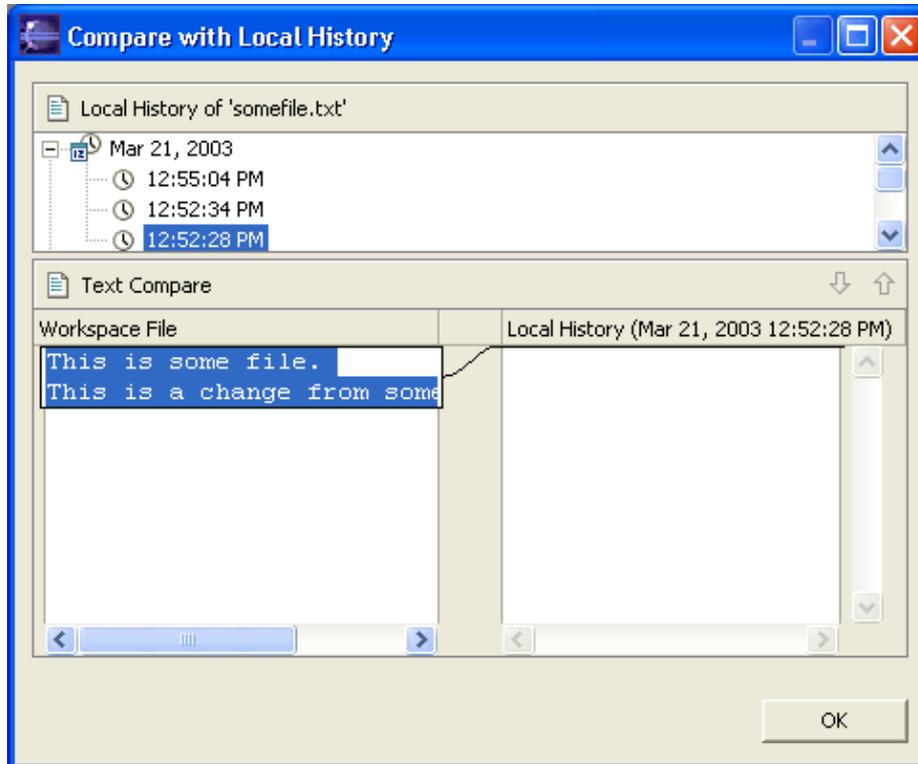
© Copyright IBM Corporation and others 2000, 2003.

# Local history

Local history of a file is maintained when you create or modify a file. Each time you edit and save a file, a copy of it is saved. This allows you to compare your current file state to a previous state, or replace the file with a previous state. Each state in the local history is identified by the date and time the file was saved.

Neither projects nor folders have local history.

Here is a look at what the local history of a Workbench file might look like:



To view the local history of a file, choose Team > Compare with > Local History... from the pop-up menu. You can select different local states in the list, which are compared against the current file. You can also revert to the local history when you select a file and select the Team > Replace With > Local History... menu item.

© Copyright IBM Corporation and others 2000, 2003.

# Keys

The function of the keyboard can be extensively customized in Eclipse. Within Eclipse, key strokes and key sequences are assigned to invoke particular commands.

## Key strokes, Key sequences, and Key bindings

A 'key stroke' is the pressing of a key on the keyboard, while optionally holding down one or more of these modifier keys: `Ctrl`, `Alt` (`Option` on the Macintosh), `Shift`, or `Command` (only on the Macintosh.) For example, holding down `Ctrl` then pressing `A` produces the key stroke `Ctrl+A`. The pressing of the modifier keys themselves do not constitute key strokes.

A 'key sequence' is one or more key strokes. Traditionally, emacs assigned two or three key stroke key sequences to particular commands. For example, the normal key sequence assigned to `Close All` in emacs is `Ctrl+X Ctrl+C`. To enter this key sequence, one presses the key stroke `Ctrl+X` followed by the key stroke `Ctrl+C`. Eclipse supports key sequences of arbitrary lengths.

A 'key binding' is the assignment of a key sequence to a command.

## Configurations

A 'configuration' is a set of key bindings. Eclipse includes two configurations:

- Standard
- Emacs (extends Standard)

The *Standard* configuration contains a general set of key bindings, in many cases recognizable to users as traditional key sequences. For instance, `Ctrl+A` is assigned to `Select All`, `Ctrl+S` is assigned to `Save`, etc.

The *Emacs* configuration contains a set of key bindings familiar to users of emacs. For instance, `Ctrl+X H` is assigned to `Select All`, `Ctrl+X S` is assigned to `Save`, etc.

It is important to understand why the *Emacs* configuration says that it 'extends Standard'. The *Emacs* configuration is not a complete set of key bindings like the *Standard* configuration. Rather, it borrows from the *Standard* configuration where possible, only defining explicit emacs style key bindings where they vary from the *Standard* configuration. Generally, only well known commands like `Select All`, `Save`, etc. have specific emacs key sequences associated with them.

The user decides which configuration they are most comfortable using by changing the 'Active Configuration' setting on the keys preference page. If the user chooses the *Standard* configuration, all *Emacs* key bindings are ignored. If the user chooses the *Emacs* configuration, explicit emacs style key sequence assignments take precedence over any conflicting assignments in the *Standard* configuration.

## Scopes

Key bindings can vary based on the current context of Eclipse.

## Crash recovery

Sometimes the active part might be a java file editor, for instance, where a different set of key sequence assignments may be more appropriate than if the active part was an html file editor. As a specific example, typically `Ctrl+B` is assigned to `Build` in a context such as java file editing, while `Ctrl+B` is assigned to `Make Text Bold` in a context such as html file editing. This context is called 'scope' and is determined by the active part. If the active part does not choose a particular scope, the workbench will set the active scope to *Global*.

Eclipse includes three different scopes. They are:

- Global
- Text Editor (extends Global)
- Java Editor (extends Text Editor)

Much like configurations, scopes can extend other scopes. For example, the *Java Editor* scope borrows key bindings from the *Text Editor* scope, which in turn borrows key bindings from the *Global* scope.

## Platform and Locale

Key bindings also vary by platform and locale. On the Macintosh platform, `Command+S` is assigned to `Save`, instead of the usual `Ctrl+S`. On Chinese locales (zh), `Alt+/` is assigned to `Content Assist`, instead of the usual `Ctrl+Space`.

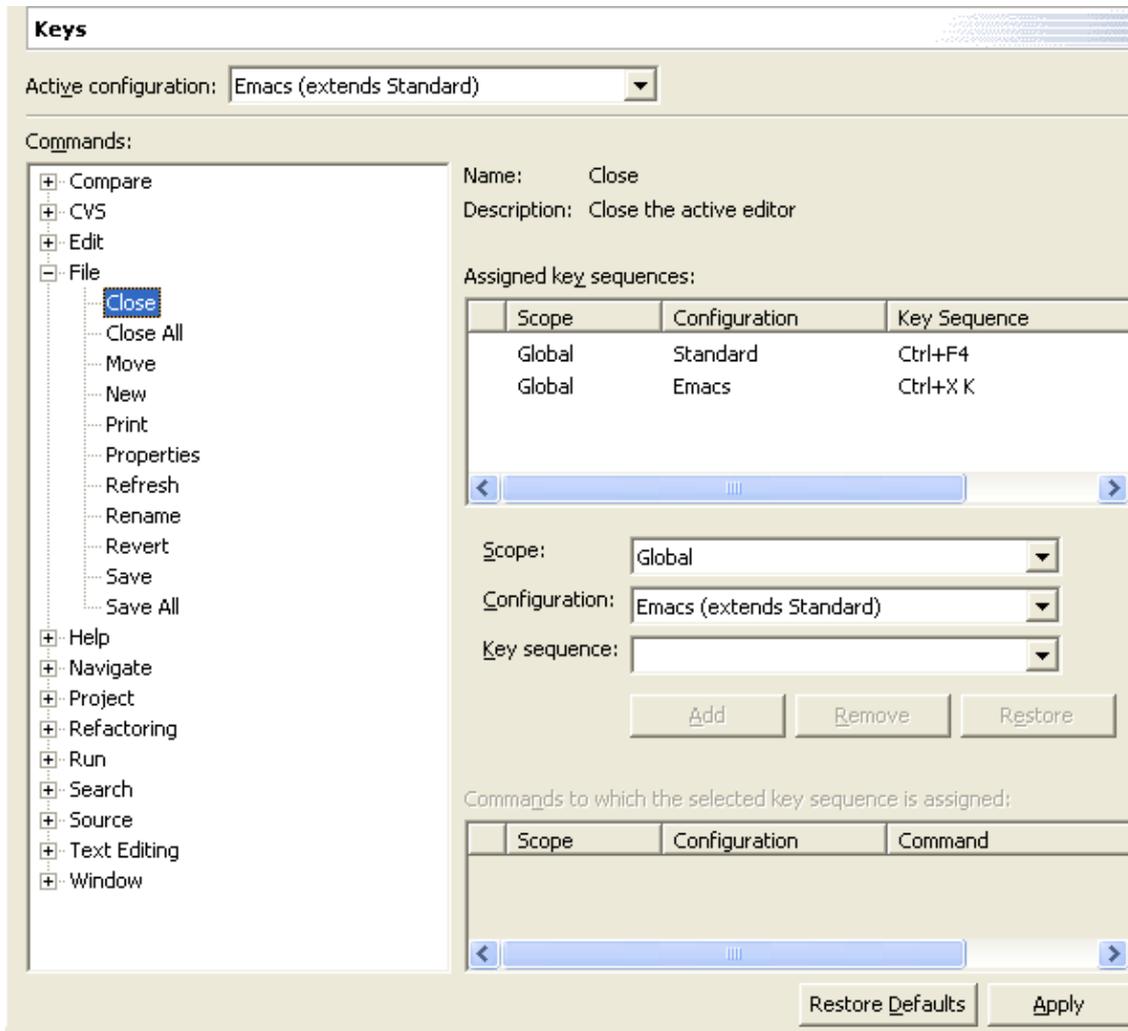
The current platform and locale is determined when Eclipse starts, and does not vary over the course of an Eclipse instance.

## Customizing Key bindings

With multi-stroke key sequences, configurations, and scopes, there are a lot of things to keep in mind when customizing key bindings. To make things easier, all key customization is done on the Keys preference page.

Select **Window > Preferences > Workbench > Keys** for the Keys preference page.

## Crash recovery



In this example, we have chosen to select *Emacs* as the active configuration, and have selected the command `Close` from the list of commands. Information on this command, along with its current key bindings, is shown on the right side.

Notice that `Close` has two key sequences assigned to it: `Ctrl+F4` in the *Standard* configuration and `Ctrl+X K` in the *Emacs* configuration. Both are assigned in the *Global* scope. Thus, if the user sets the active configuration to *Standard*, `Ctrl+F4` will be assigned to `Close` while `Ctrl+X K` will not. If the user sets the active configuration to *Emacs*, however, `Ctrl+X K` will be assigned to `Close`. As well, because the *Emacs* configuration also borrows key bindings from the *Standard* configuration, `Ctrl+F4` will also be assigned to `Close` provided that `Ctrl+F4` has not been assigned to another command in the *Emacs* configuration.

Below the list of key sequences assigned to `Close`, there is a place to add or remove key bindings. By default, it selects the scope as *Global* and the configuration to be the user's active configuration (recall above where we changed this to *Emacs*).

We type in the key sequence `Ctrl+W`, and the 'Add' button becomes enabled. Also, a list of all the commands to which the key sequence `Ctrl+W` is already assigned is shown below the Add button. We can see that `Ctrl+W` is currently assigned to the command `Cut` in the *Global* scope and *Emacs* configuration. We click the 'Add' button to assign `Ctrl+W` to `Close`.

## Crash recovery

Scope:

Configuration:

Key sequence:

Commands to which the key sequence 'Ctrl+W' is assigned:

Scope	Configuration	Command
Global	Emacs	Cut
Text Editor	Standard	Show Ruler Context Menu

Now we see Ctrl+W added to

the list of key sequences assigned to Close. Notice the small 'change' graphic  indicating that this key binding changes an existing key binding. Our new key binding, effectively removes the previous assignment of Ctrl+W to Cut. We can remove this change at any time by selecting our new key binding and clicking the 'Remove' button. The previous assignment of Ctrl+W to Cut will automatically be restored.

Name: Close

Description: Close the active editor

Assigned key sequences:

Scope	Configuration	Key Sequence
 Global	Emacs	Ctrl+W (was: Cut)
Global	Standard	Ctrl+F4
Global	Emacs	Ctrl+X K

Scope:

Configuration:

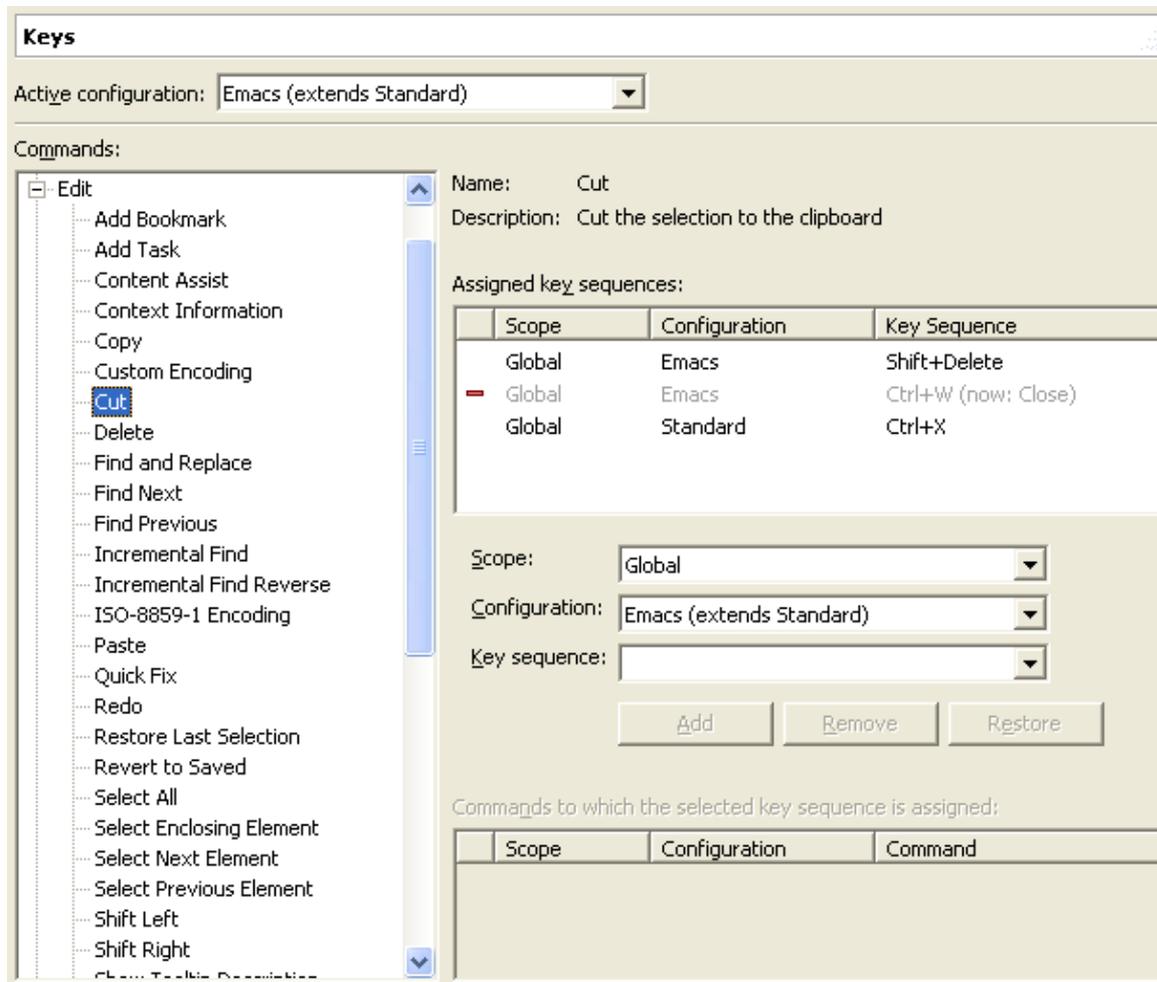
Key sequence:

Commands to which the key sequence 'Ctrl+W' is assigned:

Scope	Configuration	Command
 Global	Emacs	Close (was: Cut)
Text Editor	Standard	Show Ruler Context Menu

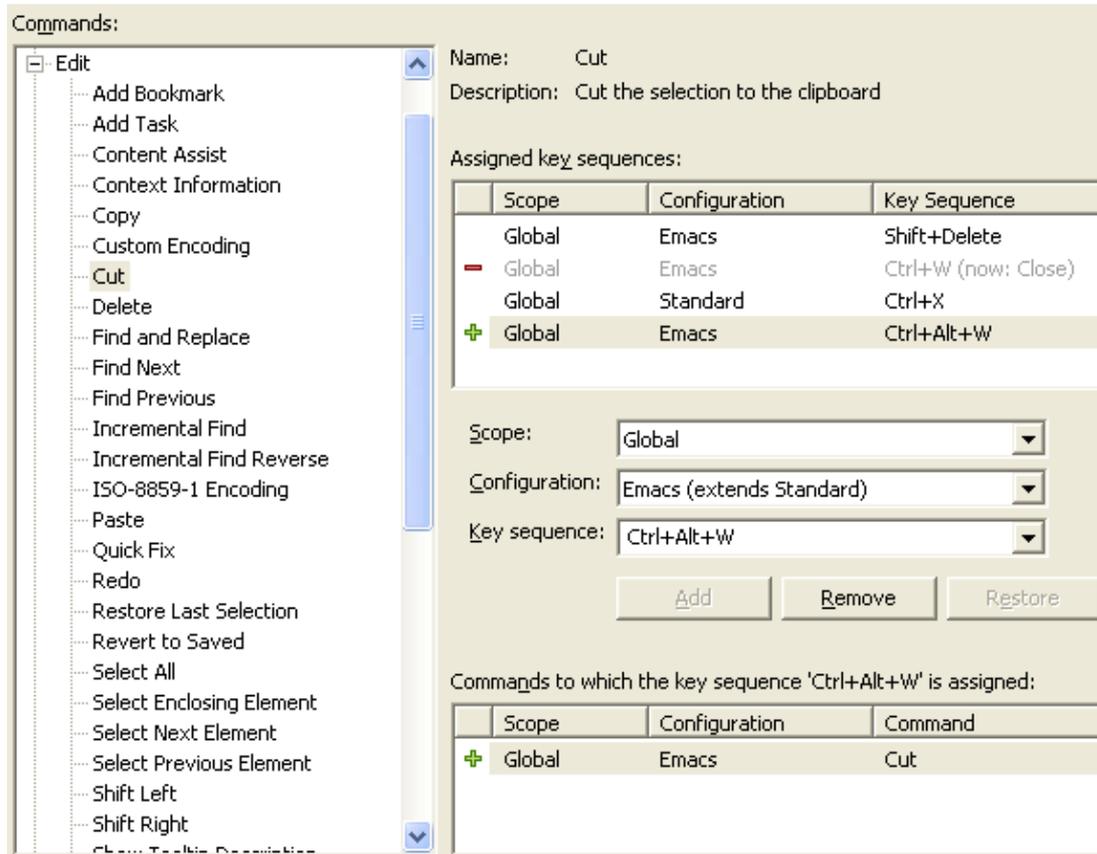
By selecting the command Cut, we can see the results of our change. Notice the small 'minus' graphic  indicating that the key binding was removed. We can restore this key binding at any time by selecting it here and clicking the 'Restore' button, effectively removing the new key binding we added in the previous step.

## Crash recovery



Perhaps we choose to assign another key to Cut in the *Emacs* configuration, say `Ctrl+Alt+W`. Adding this in the previous manner above produces the following. Notice the small 'plus' graphic  indicating that the key binding was added by the user, and was previously unassigned:

## Crash recovery



## The Dynamic Nature of Key bindings

Key bindings are provided by plug-ins, and in Eclipse, plug-ins can be added or removed. This can cause key bindings declared by these plug-ins to be added or removed. Eclipse stores custom key bindings in a way to automatically compensate for this. Consider the example above where `Ctrl+Alt+W` was assigned to `Cut` in the *Emacs* configuration. Consider the user installs a new plug-in which assigns `Ctrl+Alt+W` to a particular command. Eclipse will preserve the user's assignment to `Cut`, but show the key binding with the small 'change' graphic instead of the 'plus' graphic.

## Conflict Resolution

There are only a finite number of simple, common key strokes available to assign to a multitude of commands. We have seen that configuration, scope, platform, and locale all partition key sequence assignments into domains where they don't conflict with one another. Consider the case for `Ctrl+B` above if scopes did not exist. One plug-in would assign `Ctrl+B` to `Build`, the other plug-in would assign `Ctrl+B` to `Make Bold Text`. How would Eclipse properly resolve this conflict?

Though conflicts are drastically reduced by employing the above mechanisms, they can still occur. Two plug-ins, independent of one another, could assign the same key sequence to different commands with the same scope, configuration, platform, and locale. Consider if a plug-in assigned `Ctrl+F4` in the *Global* scope and *Standard* configuration to one of its commands. This directly conflicts with Eclipse assigning `Ctrl+F4` to the close command in the same scope and configuration.

This is a conflict. It wouldn't be proper to invoke both commands, nor would it be proper to simply choose

## Crash recovery

one of the two commands to receive the key stroke. The only proper thing to do is to ignore both key bindings, making `Ctrl+F4` effectively useless in this scope and configuration.

Conflicts of this nature are shown in the Keys preference page as follows. Notice the red text and the word "[conflict]":

Commands:

- Compare
- CVS
- Edit
- File
  - Close
  - Close All
  - Move
  - New
  - Print
  - Properties
  - Refresh
  - Rename
  - Revert
  - Save
  - Save All
- Help
- Navigate
- Project
- Refactoring
- Run
- Search
- Source
- Text Editing
- Window

Name: Close  
Description: Close the active editor

Assigned key sequences:

Scope	Configuration	Key Sequence
Global	Standard	Ctrl+F4 [conflict]
Global	Emacs	Ctrl+X K

Scope: Global  
Configuration: Standard  
Key sequence:

Add Remove Restore

Commands to which the selected key sequence is assigned:

Scope	Configuration	Command
-------	---------------	---------

These types of conflicts can be resolved by the user explicitly assigning the key sequence to one of the commands.

Another type of conflict can be caused by multiple-key stroke key sequences. For example, in the *Emacs* configuration, there are many multiple-key stroke key sequences beginning with the key stroke `Ctrl+X`. `Ctrl+H K` is assigned to `Close`. `Ctrl+X H` is assigned to `Select All`.

As previously mentioned, the *Emacs* configuration borrows key bindings from the *Standard* configuration. In the standard configuration, `Ctrl+X` is assigned to `Cut`. Though the *Emacs* configuration doesn't explicitly redefine `Ctrl+X`, pressing `Ctrl+X` is required as part of many of its key bindings. In the *Emacs* configuration, when one presses `Ctrl+X`, one is half way to entering one of many possible assigned key sequences. One would not expect the `Cut` action to be invoked at this time.

For this type of conflict, the rule is that the `Ctrl+X` key sequence assigned to `Cut` would be ignored. Otherwise, it would not be possible to complete many of the key bindings in the *Emacs* configuration.

### Related concepts

[Accessibility Features in Eclipse](#)

[Changing the key bindings](#)

[Online help system](#)

■ **Related reference**

[Font and color settings in Eclipse](#)

© Copyright IBM Corporation and others 2000, 2003.

# Workbench window layout

You can rearrange the layout of Workbench windows as follows:

- Drag views to different positions within the Workbench window.
- Drag views to the shortcut bar to create a fast view.
- Drag editors such that they are simultaneously visible beside, above, or below another editor.
- Resize views and editors by dragging the sashes which separate them.

## Drop cursors

*Drop cursors* indicate where a view will dock when you release your mouse button. This indication is relative to the view or editor area underneath the cursor.

Drop Cursors

Cursor	Name	Description
	Dock above	The view will appear above the view underneath the cursor.
	Dock below	The view will appear below the view underneath the cursor.
	Dock to the right	The view will appear to the right of the view underneath the cursor.
	Dock to the left	The view will appear to the left of the view underneath the cursor.
	Stack	The view will appear as a tab in the same pane as the view underneath the cursor.

## Fast views

From a view's title bar context menu, you can select Fast View to minimize the view as a button on the shortcut bar.

If a view is minimized in this way, you can click its button on the shortcut bar to bring it up in a fast view. To revert the fast view back to a docked view again, select the *Fast View* button on its title bar.

[See Shortcut Bar](#)

## Double-Click

Double-clicking a view's title bar maximizes the view in the Workbench window.

## Title bar context menu and fast view toolbars

From the context menu of a view's title bar, you can select how you want the view to appear within the Workbench window.

View title bar context menu options

Option	Description
Restore	Restores the view to its originating (non-maximized/non-minimized) size and position within the Workbench.
Fast View (i.e., minimize)	Minimizes the view and places a button for it in the shortcut bar.  This option, when used in a fast view, will restore the view to its originating (docked) position in the Workbench.  <u>See Shortcut Bar</u>
Maximize	Maximizes the view in the Workbench window.
Minimize	Minimizes the view in the Workbench window.
Close	Closes the view.

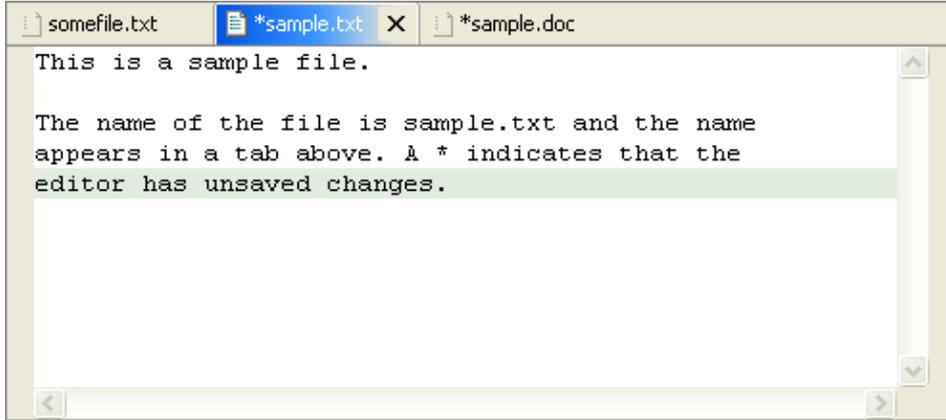
Here is what a view's context menu looks like:



# Editor area

The editor area is where you modify the contents of files in the Workbench.

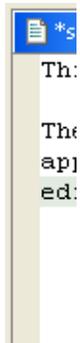
Here is what the editor area looks like when multiple files are open and a text file is being edited:



# Marker bar

The marker bar is the vertical bar located at the left of the editor area.

Here is what the marker bar looks like:



# Markers

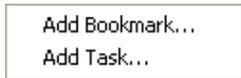
Markers are displayed in the marker bar, to the left of the text editor.

Depending on the type of file displayed in the editor area, three kinds of markers may be displayed:

- Bookmarks
- Task markers (for associated tasks)
- Debugging breakpoints

You can create and associate a marker with a specific line in a file by accessing the context menu from the marker bar, which is directly to the left of that line.

Here is what the context menu of the marker bar looks like:



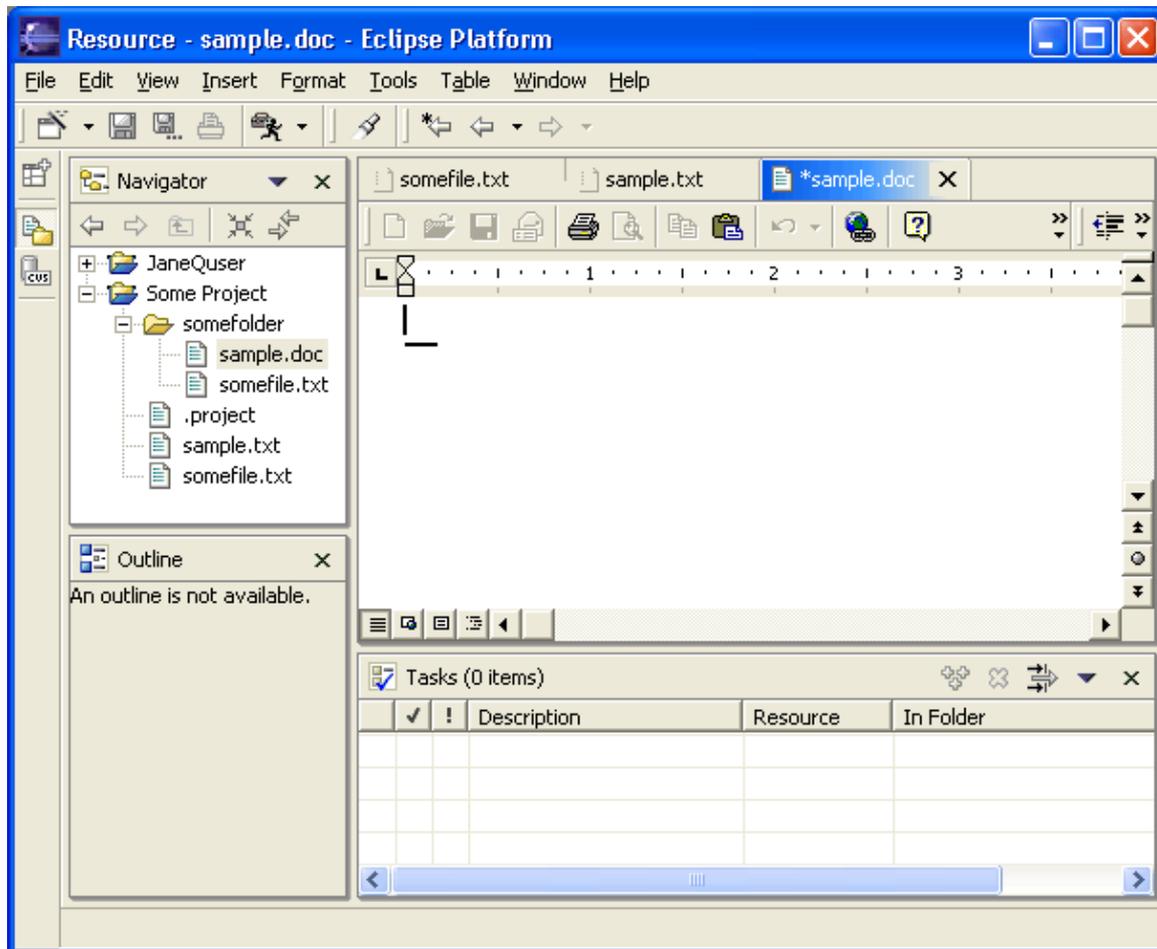
## Types of editors

The Workbench uses three types of editors:

- **Internal:** These editors are launched inside the editor area in the Workbench window.
- **External:** You can go outside the Workbench in the file system, edit a Workbench file outside the Workbench, and save the edited file. For example, imagine that you add an SGML file to the Workbench. Later, you go into the file system and open the file in an SGML editor, then save the file. The edited SGML file is still represented in the Workbench, even though you did not edit the file in the Workbench. If you associate a file type with an external editor in the Workbench (Window > Preferences > Workbench > File Associations preferences page), then the Workbench will launch this external editor.
- **ActiveX:** On Microsoft Windows platforms, the Workbench makes use of ActiveX controls for applications that allow for them. For example, Microsoft Word supports being embedded as an OLE document. Thus if you have a *.doc* file in the Workbench, and Word is registered as the editor for *.doc* files in your operating system, then opening the file will launch Word as an OLE document within the Workbench editor area. Notice how OLE documents also add such features as menus and toolbar buttons.

The following illustrates Microsoft Word embedded as an OLE document:

## Crash recovery



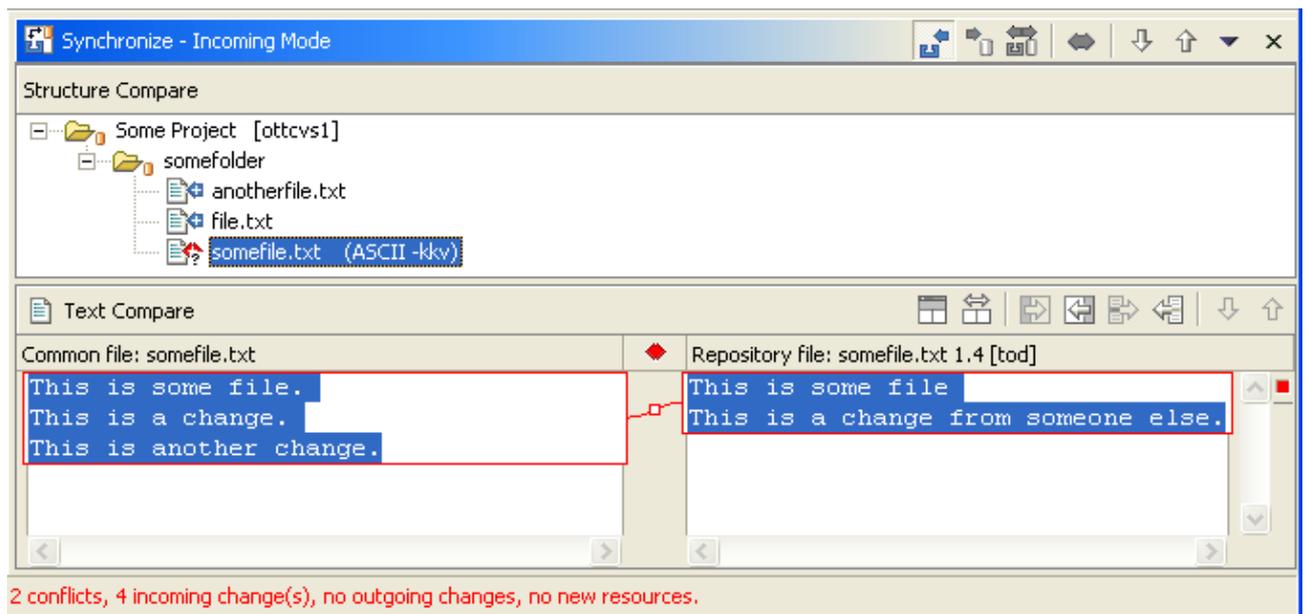
© Copyright IBM Corporation and others 2000, 2003.

# Compare editor

You can view the differences between two files by comparing them. You can compare different files, you can compare versions in the Workbench with versions in the repository, or with the local edit history. In some cases you can compare three files (when a common ancestor exists).

After a comparison is carried out, the compare editor opens in the editor area. In the compare editor, you can browse through all the differences and copy highlighted differences between the compared resources. You can save changes to resources that are made in the comparison editor.

Here is how the compare editor looks when you synchronize with a CVS repository. The compare editor is the bottom portion of the view (whose titlebar is Text Compare).



## Toolbar

The toolbar of the Compare editor includes the following buttons:

### *Control Visibility of Ancestor Pane*

There are two conditions under which a three way compare will occur, both when using the Team version management support: when comparing a file that is in conflict, and when comparing a file being merged from a branch. In both cases, the system will determine a common ancestor in the repository to compare the conflict or merge against. This button determines the visibility of the third editor. By default, the ancestor pane is not visible.

### *Perform Three way/Two way Compare*

The compare editor can be toggled between performing a three way compare or a two way compare which ignores the common ancestor.

### *Copy Whole Document from Left to Right*

Copies the entire contents of the file in the left pane into the file in the right pane, making the contents of the two files identical.

### *Copy Whole Document from Right to Left*

Does the opposite of the one just described.

## Crash recovery

### *Copy Current Change from Left to Right*

Merges changes in two files by copying the highlighted change in the left pane into the highlighted fragment on the right. This will overwrite the highlighted fragment in the right pane.

### *Copy Current Change from Right to Left*

Does the opposite of the one just described.

### *Select Next Difference*

Highlights the next difference that is found between the compared resources.

### *Select Previous Difference*

Highlights the previous difference that is found between the compared resources.

#### ■ Related concepts

[Synchronizing with a CVS repository](#)

[Three way comparisons](#)

#### ■ 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](#)

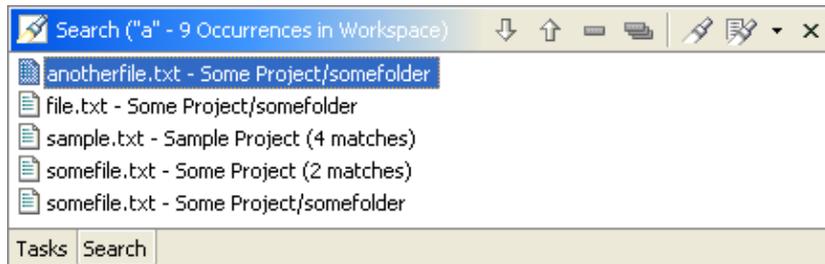
© Copyright IBM Corporation and others 2000, 2003.

# Search view

This view displays the results of a search.

Text searches will only search for expressions in files with extensions (file types) specified in the search dialog.

Here is what the Search view looks like:



## *Show Next Match*

This command highlights the next match of the search expression in the editor area, opening the file if required.

## *Show Previous Match*

This command highlights the previous match of the search expression in the editor area, opening the file if required.

## *Remove Selected Matches*

Removes all highlighted matches from the search results.

## *Search*

This command opens the Search dialog.

## *Previous Search Results*

This command allows you to browse previously conducted searches and repeat a previous search. You can select a previous search from the drop-down menu or clear the search history.

## ■ Related concepts

[Search view](#)

## ■ Related tasks

[Searching for files](#)

[Searching for text within a file](#)

## ■ Related reference

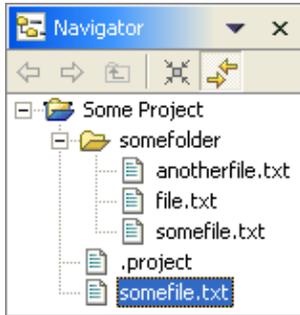
[File search](#)

© Copyright IBM Corporation and others 2000, 2003.

# Navigator view

This view provides a hierarchical view of the resources in the Workbench.

Here is what the Navigator view looks like:



# Toolbar

## Back

This command displays the hierarchy that was displayed immediately prior to the current display. For example, if you *Go Into* a resource, then the *Back* command in the resulting display returns the view to the same hierarchy from which you activated the *Go Into* command. The hover help for this button tells you where it will take you. This command is similar to the *Back* button in a web browser.

## Forward

This command displays the hierarchy that was displayed immediately after the current display. For example, if you've just selected the *Back* command, then selecting the *Forward* command in the resulting display returns the view to the same hierarchy from which you activated the *Back* command. The hover help for this button tells you where it will take you. This command is similar to the *Forward* button in a web browser.

## Up

This command displays the hierarchy of the parent of the current highest level resource. The hover help for this button tells you where it will take you.

## Collapse All

This command collapses the tree expansion state of all resources in the view.

## Link with Editor

This command toggles whether the Navigator view selection is linked to the active editor. When this option is selected, changing the active editor will automatically update the Navigator selection to the resource being edited.

# Menus

Click the icon at the left end of the view's title bar to open a menu of items generic to all views. Click the black upside-down triangle icon to open a menu of items specific to the Navigator view. Right-click inside the view to open a context menu.

## Select Working Set

Opens the **Select Working Set** dialog to allow selecting a working set for the Navigator view.

## Deselect Working Set

Deselects the current working set.

## Edit Active Working Set

Opens the **Edit Working Set** dialog to allow changing the current working set.

## Sort

This command sorts the resources in the Navigator view according to the selected schema:

- **By Name:** Resources are sorted alphabetically, according to the full name of the resource (e.g., A.TXT, then B.DOC, then C.HTML, etc.)
- **By Type:** Resources are sorted alphabetically by file type/extension (e.g., all DOC files, then all HTML files, then all TXT files, etc.).

## Filters

This command allows you to select filters to apply to the view so that you can show or hide various resources as needed. File types selected in the list will not be shown in the Navigator. Here is what the file filters dialog looks like:



## Link with Editor

See the toolbar item description above.

In addition to these menu items, the Navigator view menu shows a list of recently used working sets that have been selected in the view.

# Context menu

## New

This command allows you to create a new resource in the Workbench. Select the type of resource to create from the submenu.

## Go Into

This command displays a new hierarchy in the Navigator view, with the children of the selected resource as its contents. For example if you Go Into a project, the Navigator will be refocused on the immediate files and folders of the project.

## Open

This command opens the selected resource. If the resource is a file that is associated with an editor, then the Workbench launches the associated internal, external, or ActiveX editor and opens the file in that editor.

## Open With

This command allows you to open an editor other than the default editor for the selected resource. Specify the editor with which to open the resource by selecting an editor from the submenu.

## Copy

This command copies the selected resource to the clipboard.

## Paste

This command pastes resources on the clipboard into the selected project or folder. If a resource is selected the resources on the clipboard are pasted as siblings of the selected resource.

## Delete

This command deletes the selected resource from the workspace.

## Move

This command moves the selected resource to another location. A dialog will appear, prompting for the destination location to which the resource will be moved.

## Rename

This command allows you to specify a new name for the selected resource.

## Import

This command opens the import wizard and allows you to select resources to import into the Workbench.

## Export

This command opens the export wizard and allows you to export resources to an external location.

## Add Bookmark

This command adds a bookmark that is associated with the selected resource (but not to a specific location within the resource).

## Refresh

This command refreshes the Workbench's view of the selected resource and its children. For example, this is used when you create a new file for an existing project outside the Workbench and want the file to appear in the Navigator view.

## Close Project

The close project command is visible when an open project is selected. This command closes the selected project.

## Open Project

The open project command is visible when a closed project is selected. This command opens the selected project.

## Team

Menu items in the Team submenu are related to version control management and are determined by the version control management system that is associated with the project. Eclipse provides the special menu item **Share Project...** for projects that are not under version control management. This command will present a wizard that allows the user to choose to share the project with any version control management systems that has been added to Eclipse. Eclipse ships with support for CVS.

## Compare With

Commands on the Compare With submenu allow you to do one of the following types of compares:

- Compare two or three selected resources with each other
- Compare the selected resource with remote versions (if the project is associated with a version control management system).
- Compare the selected resource with a local history state

After you select the type of compare you want to do, you will either see a compare editor or a compare dialog.

## Crash recovery

In the compare editor, you can browse and copy various changes between the compared resources. In the compare dialog, you can only browse through the changes.

## Replace With

Commands on the Replace With submenu allow you to replace the selected resource with another state from the local history. If the project is under version control management, there may be additional items supplied by the version control management system as well.

## Properties

This command displays the properties of the selected resource. The kinds of properties that are displayed depend on what type of resource is selected. Resource properties may include (but are not limited to):

- Path relative to the project in which it is held
- Type of resource
- Absolute file system path, or name of path variable when using linked resources
- Resolved path variable when using a path variable for a linked resource
- Size of resource
- Last modified date
- Read-only status
- Derived resource status
- Execution arguments, if it is an executable resource
- Program launchers, if it is launchable
- Project dependencies, if any

### ■ Related concepts

[Team programming with CVS](#)

[Three-way compare](#)

[Linked resources](#)

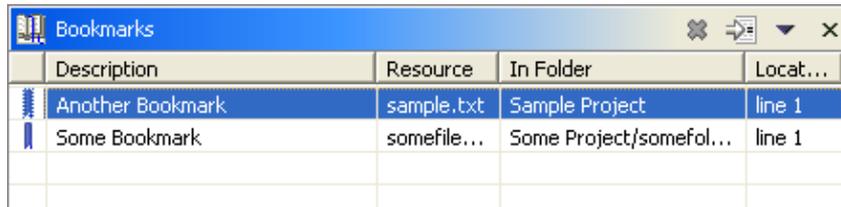
### ■ Related reference

[Compare editor](#)

© Copyright IBM Corporation and others 2000, 2003.

# Bookmarks view

The Bookmarks view displays user defined bookmarks.



Description	Resource	In Folder	Locat...
Another Bookmark	sample.txt	Sample Project	line 1
Some Bookmark	somefile...	Some Project/somefol...	line 1

To add the Bookmarks view to the current perspective, click **Window > Show View > Other > Basic > Bookmarks**.

The Description column contains a description of the bookmark. You can edit the description by selecting **Properties** from the context menu.

The Resource and In Folder columns provide the name and location of the resource associated with each bookmark.

The Location column indicates the line number of the bookmark within its associated resource.

## Toolbar

The toolbar of the Bookmarks view includes the following buttons.

### *Delete*

Delete the selected bookmark.

### *Go to*

Open the bookmark's resource and navigate to the bookmarked region.

## Menus

Click the icon at the left end of the view's title bar to open a menu of items generic to all views. Click the black upside-down triangle icon to open a menu of items specific to the Bookmarks view. Right-click inside the view to open a context menu.

### ■ Related tasks

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

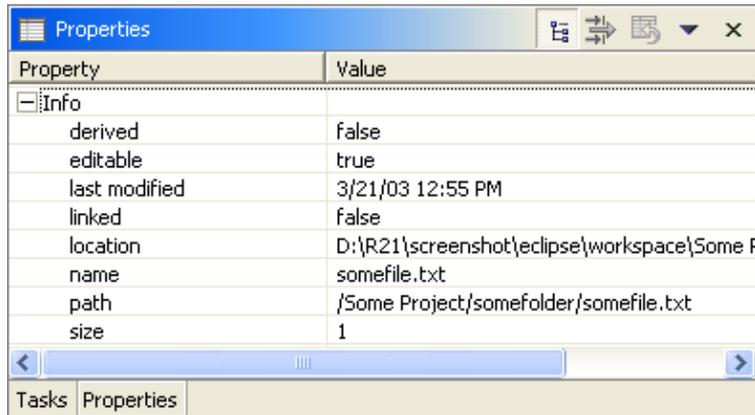
[Creating a bookmark within a file](#)

[Deleting a Bookmark](#)

© Copyright IBM Corporation and others 2000, 2003.

# Properties view

This view displays property names and basic properties of a selected resource. Here is an example:



The screenshot shows the Eclipse IDE's Properties view. The title bar reads 'Properties'. The view contains a table with two columns: 'Property' and 'Value'. The table is organized into a tree structure under an 'Info' category. The properties listed are: derived (false), editable (true), last modified (3/21/03 12:55 PM), linked (false), location (D:\R21\screenshot\eclipse\workspace\Some F...), name (somefile.txt), path (/Some Project/somefolder/somefile.txt), and size (1). Below the table is a scrollbar and a tab bar with 'Tasks' and 'Properties' tabs.

Property	Value
Info	
derived	false
editable	true
last modified	3/21/03 12:55 PM
linked	false
location	D:\R21\screenshot\eclipse\workspace\Some F...
name	somefile.txt
path	/Some Project/somefolder/somefile.txt
size	1

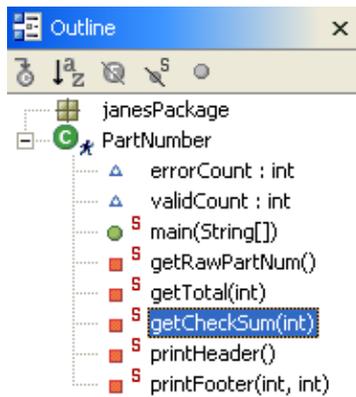
Toolbar buttons allow you to toggle whether to display properties by category and whether to filter advanced properties. Another toolbar button allows you to restore the selected property to its default value.

To see more detailed information about a resource than the Properties view gives you, right-click the resource name in the Navigator view and select Properties from the pop-up menu.

© Copyright IBM Corporation and others 2000, 2003.

# Outline view

This view displays an outline of a structured file that is currently open in the editor area, and lists structural elements. The contents of the outline view are editor-specific. In the example below, which is for a Java source file, the structural elements are classes, variables, and methods. The contents of the toolbar are also editor-specific.

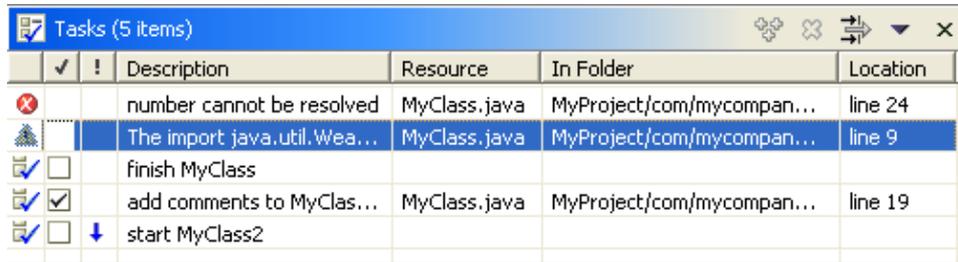


© Copyright IBM Corporation and others 2000, 2003.

# Tasks view

The Tasks view displays the following information:

- System-generated errors, warnings, or information associated with a resource. These are typically produced by builders. For example, if you save a Java source file that contains syntax errors, the errors will automatically be logged in this view.
- Tasks that you add manually. You can associate a task with a resource in the Workbench, but this is not required.



✓ !	Description	Resource	In Folder	Location
	number cannot be resolved	MyClass.java	MyProject/com/mycompan...	line 24
	The import java.util.We...	MyClass.java	MyProject/com/mycompan...	line 9
	finish MyClass			
	add comments to MyClas...	MyClass.java	MyProject/com/mycompan...	line 19
	start MyClass2			

By default, the Tasks view is included in the Resources perspective. To add it to the current perspective, click **Window > Show View > Other > Basic > Tasks**.

By default, the Tasks view is shown when a new compiler error or warning is generated. To remove this feature, click **Window > Preferences > Workbench** and deselect **Show Tasks view when build has errors or warnings**.

The following icons are used by the Tasks view:

Icon	Description
	Information
	Warning
	Error
	Task (normal priority)
	High priority task
	Low priority task
	Completed task

The first column indicates whether the line item is a task or a compiler generated error, warning or info.

The second column indicates whether the task is completed. Completed tasks are flagged with a check mark, which you add manually.

The third column indicates whether the task is high, normal, or low priority.

## Crash recovery

The Description column contains a description of the line item. You can edit the description of user-defined tasks by selecting **Properties** from the context menu.

The Resource and In Folder columns provide the name and location of the resource associated with each line item.

The Location column indicates the line number of the line item within its associated resource.

## Toolbar

The toolbar of the Tasks view includes the following buttons.

### *New task*

Manually add a "to do" item to the Tasks view.

### *Delete*

Delete the selected line item.

### *Filter*

Filter the view according to the type of item.

## Menus

Click the icon at the left end of the view's title bar to open a menu of items generic to all views. Click the black upside-down triangle icon to open a menu of items specific to the Tasks view. Right-click inside the view to open a context menu.

### ■ Related tasks

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

[Associating a task with a resource](#)

[Deleting tasks](#)

[Filtering the task view](#)

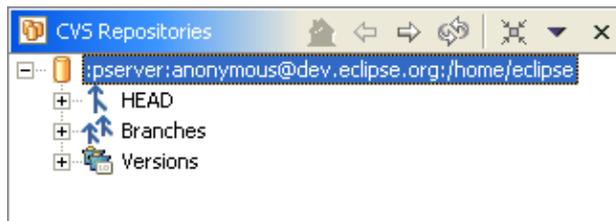
© Copyright IBM Corporation and others 2000, 2003.

# CVS Repositories view

The CVS Repositories view, which is part of the CVS Repository Exploring perspective, shows the CVS repository locations that you have added to your Workbench. Expanding a location reveals the main trunk (HEAD), project versions and branches in that repository. You can further expand the project versions and branches to reveal the folders and files contained within them.

The pop-up menu for this view also allows you to specify new repository locations. Use the CVS Repositories view to check out resources from the repository to the Workbench, configure the branches and versions shown by the view, view resource history, and compare resource versions.

Here is what the Repositories view looks like:



Like other views in the Workbench, the CVS Repositories view has its own toolbar. Toolbar buttons are provided for refreshing the view and for navigating. The toolbar also contains a drop-down menu that allows the user to filter the view by working set.

## Titlebar

### Go Home

This command redraws the view, showing the repository locations.

### Go Back

This command redraws the view, showing the hierarchy that was displayed immediately prior to the current display.

### Go Forward

This command redraws the view, showing the hierarchy that was displayed immediately after the current display.

### Refresh View

This command refreshes the contents of the Repositories view.

### Collapse All

This command collapses all expanded entries in the Repositories view.

## Drop–Down Menu

The drop down menu in the title bar allows the repositories view to be filtered by a working set.

## Context menu

From the context menu of the Repositories view you can perform a number of interesting operations.

### New > Repository Location

This command brings up the wizard to help you specify a new repository location.

### Configure Branches and Versions

This command brings up the wizard to help you discover the branch and version tags that exist in the repository for the selected folder so they can be added to the repositories view to allow the resources that have these tags can be browsed.

### Refresh Branches and Versions

This command, available on repository location entries, allows you to refresh the list of known branches and versions that are displayed in the repositories view for selected projects. This operation makes use of the defined auto–refresh files for each project. If the operation fails for a particular project, use Configure Branches and Versions on the project to select one or more appropriate refresh files.

### Add to Branch List...

This command adds the selected project to the list of projects that are displayed under the specified branch in the repositories view. This command only modifies the repositories view and does not effect the repository in any way. If you want to add the project to a branch, you can perform a *Tag with Existing* after performing this operation.

### Open

This command opens the selected file in an editor. Since file revisions in the repository are immutable, the editor opens in a read–only state, so it is not modifiable.

### Show in Resource History

This command shows the revision history of the selected file in the CVS Resource History view.

### Compare

This command will compare on two or three selected resources.

### Compare With...

This command will compare the selected folder with a branch or version of the same folder.

## Checkout as Project

This command checks the folder out into a project in the Workbench with the same name as the folder.

## Checkout as...

This command checks the folder out into a project in the Workbench into a project configured by the user.

## Checkout Into...

This command checks the folder out as a sub–folder of an existing project.

## Checkout Out Module

This command checks out the selected module into one or more projects in the workspace.

## Tag as Version...

This command versions the selected resource based on the current branch contents.

## Tag with Existing...

This command versions the selected resource based on the current branch contents, moving the tag from previously tagged resources if required.

### ■ Related concepts

[Team programming with CVS](#)

[CVS Repositories](#)

[Branches](#)

[Versions](#)

[Local history](#)

### ■ Related tasks

[Creating a CVS repository location](#)

[Discarding a CVS repository location](#)

[Refreshing the CVS repositories view](#)

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

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

[Replacing resources in the Workbench](#)

[Sharing a new project using CVS](#)

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

### ■ Related reference

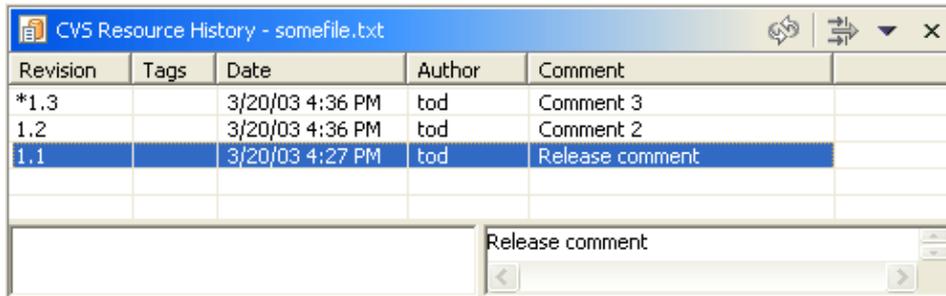
[CVS](#)

[CVS Resource History View](#)

# CVS Resource History view

This view provides a list of all the revisions of a resource in the repository. From this view you can compare two revisions, replace (get sticky) or revert (get contents) the corresponding workspace file to the revision, or open an editor on a revision.

Here is what the CVS Resource History View looks like:



The screenshot shows a window titled "CVS Resource History - somefile.txt". It contains a table with the following data:

Revision	Tags	Date	Author	Comment
*1.3		3/20/03 4:36 PM	tod	Comment 3
1.2		3/20/03 4:36 PM	tod	Comment 2
1.1		3/20/03 4:27 PM	tod	Release comment

Below the table, there is a text area displaying the comment "Release comment" for the selected revision (1.1).

For each resource, the following is displayed in the table:

## Revision

This column displays the revision number in the history.

## Tags

The tags that are associated with the revision. Selecting a revision line will list the tags in the lower left pane of the view.

## Date

This column displays the creation date and time of the revision in the history.

## Author

This column displays the name of the person who created and authored the version.

## Comment

This column displays the comment (if any) supplied for this revision at the time it was committed. Selecting a revision line will show the complete comment in the lower right pane of the view.

### ■ Related reference

[CVS Repositories view](#)

### ■ Related concepts

[Resources](#)

CVS Repositories

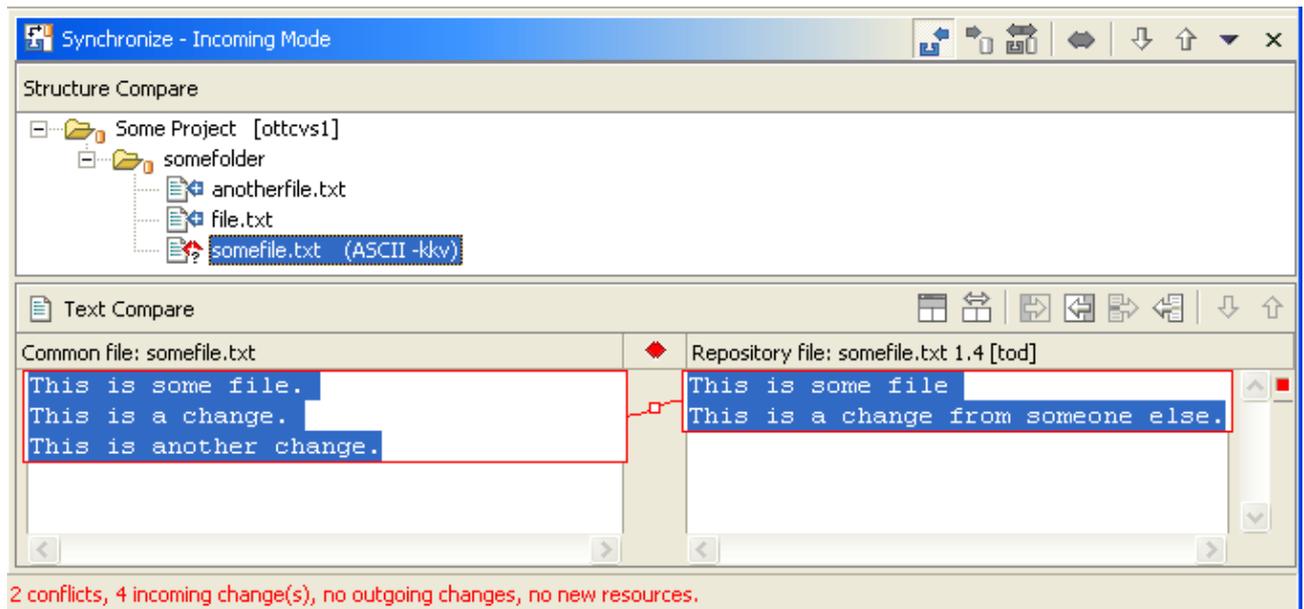
Local history

© Copyright IBM Corporation and others 2000, 2003.

# Synchronize view

This view allows you to update resources in the Workbench and commit resources from the Workbench to a repository.

Here is what the Synchronize view looks like:



## Modes

There are three different viewing modes (filters) that you can use in this view to make the synchronization process easier:

### *Incoming*

In this mode, only resources which have been changed in the repository since they were last loaded or synchronized with those in the Workbench (incoming changes) are visible. If you have incoming changes when you synchronize, you will start off in this mode.

### *Outgoing*

In this mode, only resources which have been modified in the Workbench (outgoing changes) are visible. If you do not have incoming changes when you synchronize, you will start in this mode.

### *Incoming/Outgoing*

In this mode both incoming and outgoing changes are shown, and you can both update and commit. You will remain in this mode with future synchronizations until you close the view. The advantage to using this dual mode is you can do either task as you choose. The disadvantage is that performing a simultaneous bi-directional merge is often complicated. In addition, you risk accidentally committing a conflict when you meant to update to it. You can choose to always use Incoming/Outgoing mode on the Workbench preferences page (**Window > Preferences > Team**).

Modes not only filter the set of visible changes but also restrict the set of operations available. You can switch modes using the toolbar of the Synchronization view.

The advantages to modes are:

## Crash recovery

- They support and encourage the ideal work flow.
- They reduce the amount of information you are faced with at any one time.
- They provide a degree of safety restricting you to those operations appropriate to that mode. For example, you can't accidentally commit a conflict when in incoming mode.

Regardless of the mode that you select, you will always see any conflicts that exist. You can filter out other outgoing and incoming changes by clicking the **Show Only Conflicts** button. You can also choose to Compare File Contents (and show only those resources whose contents differ), Ignore Whitespace differences or Refresh From Repository from the drop-down menu of this view.

**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.

## Synchronization state

The top left pane in the synchronize view shows the synchronization state of resources in your workspace compared to those in the repository. This state is shown by using icons and can also be configured to show the state as text appended to the resource name. A description of the icons is shown in the table below:

	An incoming addition means that a resource has been added to the repository. <i>Updating</i> will transfer the resource to your workspace.
	An incoming change means that the file has changed in the repository. <i>Updating</i> will transfer the new file revision to your workspace.
	An incoming deletion means that a resource was deleted from the server. <i>Updating</i> will delete your local resource.
	An outgoing addition means that the file was added to your workspace and is not yet in the repository. <i>Adding</i> then <i>Committing</i> will transfer the new file to the repository.
	An outgoing change means that the file was change locally. <i>Committing</i> the file will transfer the changes to the repository and create a new revision of the file.
	An outgoing deletion is a resource that has been deleted locally. <i>Committing</i> these resources will cause the remote resource to be deleted. <i>Note:</i> in CVS directories are never really deleted from the repository. Instead, files are deleted and empty directories are pruned from your workspace.
	A conflicting additions means that the resource has been added locally and remotely.
	A conflicting change means that the file has been changed locally and remotely. A manual or automatic merge will be required by the user.
	A conflicting deletion means that the resource was deleted locally and remotely.
	An auto-mergeable conflict is a conflict that can be automatically merged by CVS. This is because the file was changed locally and remotely but in different locations in the file. This indicator is important because it can allow you to review the changes from within the synchronization view and then perform a <i>Override and Update</i> that will perform the merge automatically for you.
	A non-added resource that has never been committed to the repository.

## Structure compare pane

This pane allows you to view a hierarchy tree of all the resource differences between the repository and the Workbench.

## Text compare pane

This pane allows you to view the content differences of the resource selected in the structure compare pane. You can choose to ignore whitespace differences from the view's drop-down menu.

### ■ Related concepts

[Team programming with CVS](#)

[Synchronizing with a CVS repository](#)

### ■ Related tasks

[Synchronizing with the repository](#)

[Updating](#)

[Resolving conflicts](#)

[Merging from a branch](#)

[Committing](#)

[Comparing resources](#)

[Merging changes in the compare editor](#)

### ■ Related reference

[CVS](#)

[Compare editor](#)

© Copyright IBM Corporation and others 2000, 2003.

# Ant editor

The Ant editor provides specialized features for editor Ant buildfiles. Associated with the editor is a Ant buildfile specific Outline view which shows the structure of the Ant build file. It is updated as the user edits the buildfile

The editor includes the following features:

- Syntax highlighting
- Content/code assist
- Annotations

The most common way to invoke the Ant editor is to open an Ant buildfile from the Navigator or Package explorer using pop-up menus or by clicking the file (single or double-click depending on the user preferences).

## ■ Related concepts

[Ant support](#)

[External tools](#)

## ■ Related tasks

[Running Ant buildfiles](#)

[Modifying the Ant classpath](#)

[Adding new Ant tasks and types](#)

[Using a different version of Ant](#)

## ■ Related reference

[Ant editor preferences](#)

[External Tools and Ant icons](#)

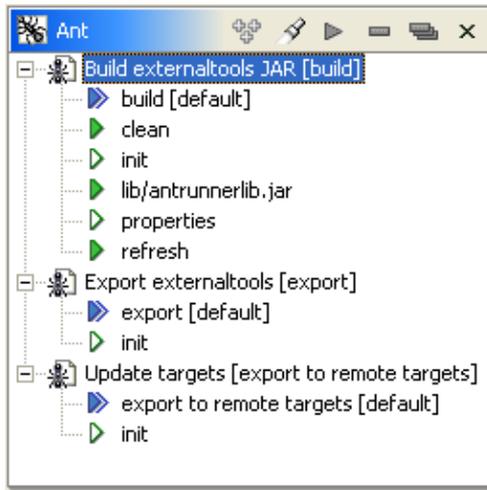
© Copyright IBM Corporation and others 2000, 2003.

# Ant view

The Ant view provides a place to view Ant buildfiles and makes it easy to execute a buildfile or a given target within a buildfile. You can add files to the view and expand buildfiles to reveal the targets defined within them.

The pop-up menu for this view also allows you to open an editor on a buildfile, execute the selected target or buildfile, and configure the launch configuration used when executing.

Here is what the Ant view looks like:



Like other views in the Workbench, the Ant view has its own toolbar. Toolbar buttons are provided for adding buildfiles to the view and for executing them.

## Titlebar

### Add Buildfile

This command prompts you to select a buildfile in the workspace to be added to the view.

### Add Buildfiles with Search

This command searches for buildfiles and adds them to the Ant view. You can specify a file name pattern to search for and limit the scope of the search.

### Run the Default Target of the Selected Buildfile

This command runs the default target of the selected buildfile or the selected target. The buildfile is executed using the launch configuration associated with the buildfile. Note that the target execution is based on the selection in the view; the targets selected in the launch configuration are ignored.

## Refresh View

This command refreshes the contents of the Repositories view.

## Collapse All

This command collapses all expanded entries in the Repositories view.

## Remove Selected Buildfile

This command removes the selected buildfiles from the view.

## Remove All Buildfiles

This command removes all buildfiles from the view.

## Context menu

The context menu contains a few actions in addition to the actions available in the toolbar.

## Properties...

This command brings up the launch configuration dialog to edit the configuration associated with the selected buildfile.

## Open With >

This menu allows you to open an editor on the selected buildfile.

### ■ Related concepts

[Ant support](#)

[External tools](#)

### ■ Related tasks

[Running Ant buildfiles](#)

[Modifying the Ant classpath](#)

[Adding new Ant tasks and types](#)

[Using a different version of Ant](#)

### ■ Related reference

[Ant runtime preferences](#)

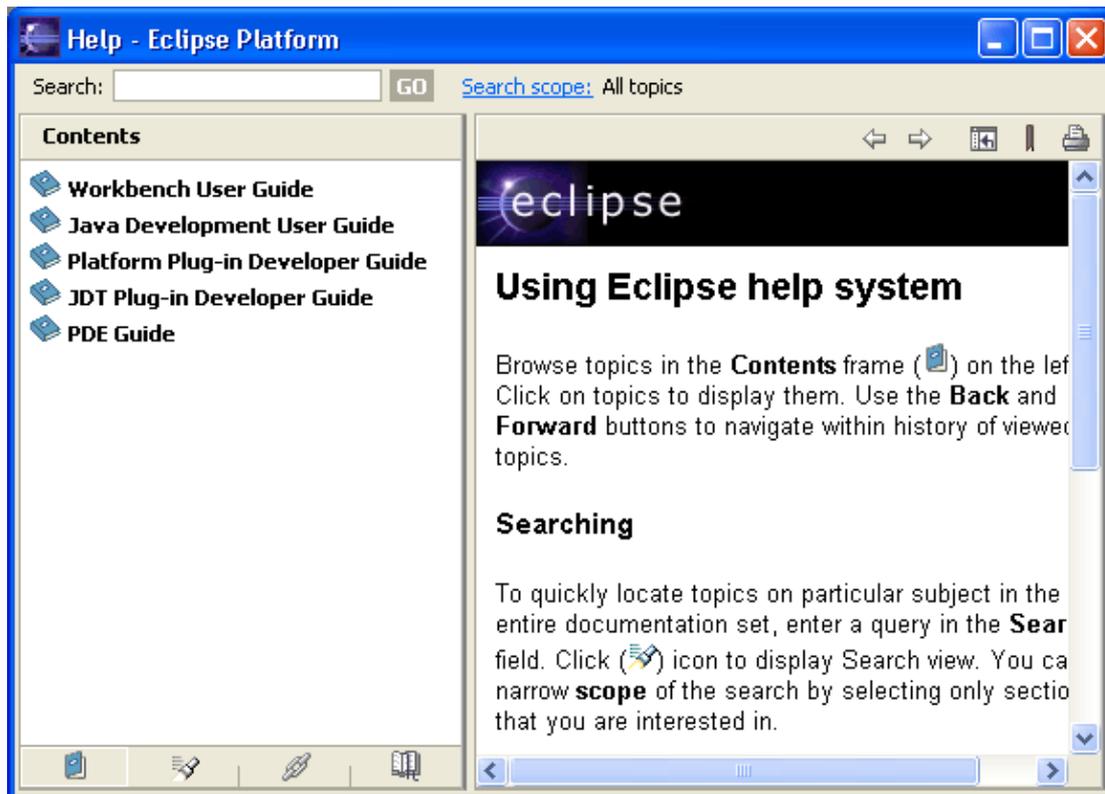
[Ant Editor](#)

[External Tools and Ant icons](#)

# Help view

This view displays help on using the Workbench. If you are reading this document, you've probably found this view already.

Here is what the Help view looks like, when using embedded browser:



## Table of Contents (Bookshelf)

This list contains all the available sets of information (or books). Click a book to see its topics tree.

## Contents Tab

This tab displays an outline of the available Help topics for the selected book.

## Search Results Tab

If you run a search of the documentation, the results list will be displayed on this tab.

## Links Tab

If you invoke the context sensitive (F1) help, all the related links for that context are displayed in a list on this tab.

## Bookmarks Tab

This list displays topics that were bookmarked using Bookmark Document button.

## Search

Type a search query in the field and click Go. Click Advanced Search to enter a longer query or to filter the results by available books.

## Search scope

Click the link to change the documentation set to be searched.

## Go Back

Click this button to display the previously displayed page.

## Go Forward

You can use this button if you have just used the Go Back button. Click it to display a previously displayed page.

## Synchronize Navigation

Click this button to match the navigation tree up to the topic currently shown in the content frame.

## Bookmark Document

This button adds the currently displayed document to the Bookmarks view.

## Print Page

This command prints the currently displayed topic.

### Related reference

[Workbench User Guide](#)

© Copyright IBM Corporation and others 2000, 2003.

# Update manager views

The update manager consists of four related views that are intended to be used as part of the Install/Update perspective. They are:

- Install Configuration view
- Feature Updates view
- Preview
- External Preview

## ■ Related concepts

[Features](#)

## ■ Related tasks

[Updating Eclipse with the update manager](#)

## ■ Related reference

[Install Configuration view](#)

[Feature Updates view](#)

[Preview](#)

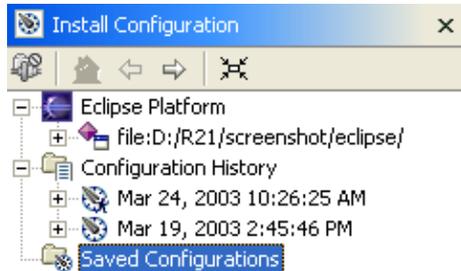
[External Preview](#)

© Copyright IBM Corporation and others 2000, 2003.

# Install Configuration view

The Install Configuration view, which is part of the Install/Update perspective, allows you to manage your workspace configuration. The **Current Configuration** folder shows which local installation locations are part of the workspace configuration, and which features are installed in those locations. The **Configuration History** folder shows a history of changes made to your workspace configuration. The **Saved Configurations** folder show any configurations that were preserved (typically representing a stable configuration that can be used as a base to recover to).

Here is what the Install Configuration view looks like:



Like other views in the Workbench, the Install Configuration view has its own toolbar. Toolbar buttons are provided for navigating and to toggle the view to show features that are currently disabled.

## Titlebar

### Go Home

This command redraws the view, showing the standard configuration folders.

### Go Back

This command redraws the view, showing the hierarchy that was displayed immediately prior to the current display.

### Go Into

This command redraws the view, re-parenting the view hierarchy at the selected item.

### Show Disabled Features

This command toggles the view to show features that are installed but are currently disabled.

## Context Menu – common

Context menus for all selections within the view have the following actions:

## **Go Home**

This action redraws the view, showing the standard configuration folders.

## **Go Back**

This action redraws the view, showing the hierarchy that was displayed immediately prior to the current display.

## **Go Into**

This action redraws the view, re-parenting the view hierarchy at the selected item.

## **Show Details**

This action shows details for the selected item in the Preview, opening the Preview if required.

## **Context Menu – features**

Context menu for feature selections have the following additional actions.

### **Show Status**

This action displays a dialog detailing any configuration errors or warnings associated with the selected feature.

## **Context Menu – histories**

Context menu for history selections have the following additional actions:

### **Save**

This action preserves the selected history in the Saved Configurations folder. The action is only available for items in the Configuration History folder.

### **Remove**

This action removes the selected history. The action is only available for items in the Saved Configurations folder.

### **Restore**

This action restores the selected configuration as the current configuration. This action is not available when the current configuration is selected (the top history item). Restoring a configuration requires the workbench to be restarted.

## Properties

This action allows the selected item properties to be edited.

### ■ Related concepts

[Features](#)

### ■ Related tasks

[Updating Eclipse with the update manager](#)

### ■ Related reference

[Feature Updates View](#)

[Preview](#)

[External Preview](#)

© Copyright IBM Corporation and others 2000, 2003.

# Feature Updates view

The Feature Updates view, which is part of the Install/Update perspective, allows you to locate and install new features and feature updates. The view contains three standard folders. The **Sites to Visit** folder contains updates sites preconfigured by the product you are using. The **Available Updates** is a built-in search results folder used for searching update sites. The **My Computer** folder is a file explorer that can be used to navigate to updates located in the file system.

In addition to the standard folders, you can create additional content in the Feature Updates view by adding update site bookmarks, and custom searches.

Here is what the Feature Updates view looks like:



Like other views in the Workbench, the Feature Updates view has its own toolbar.

## Titlebar

### Refresh

This command refreshes the content of the view.

### Drop Down Menu

The drop down menu in the title bar allows the following toggle settings:

- **Show My Computer Files** – whether the My Computer explorer shows only the directory structure, or all files
- **Show Site Categories** – whether features on updates sites are shown in their respective categories, or as a flat list
- **Show Matching Features Only** – whether update sites show only features that match the current execution environment, or all features

## Context Menu – common

Context menus for all selections within the view have the following actions:

### Refresh

This action refreshes the content of the selected item.

### New, Site Bookmark

This action shows a dialog for creating update site bookmarks.

### New, Folder

This action shows a dialog for creating new folders.

### New, Search

This action shows a dialog for creating custom searches. The following search types can be created:

- Available Updates
- Regular Expression
- Plug-ins in Features

### Show Details

This action shows details for the selected item in the Preview, opening the Preview if required.

## Context Menu – custom items

Context menu for items added to the Feature Updates view (bookmarks, folders, searches) have the following additional actions:

### Delete

This action removes the selected item.

#### ■ Related concepts

[Features](#)

#### ■ Related tasks

[Updating Eclipse with the update manager](#)

[Searching for features](#)

#### ■ Related reference

[Install Configuration View](#)

[Preview](#)

External Preview

© Copyright IBM Corporation and others 2000, 2003.

# Preview

The Preview view, which is part of the Install/Update perspective, displays detailed information about the current active selection. Its content changes as the selection changes.

In some cases, the Preview is simply an extended description of the selection, or a page offering suggested actions based on the selection. In other cases the preview can be used to perform specific actions against the selection.

Preview for the current configuration displays information about the primary feature defined for the workspace.



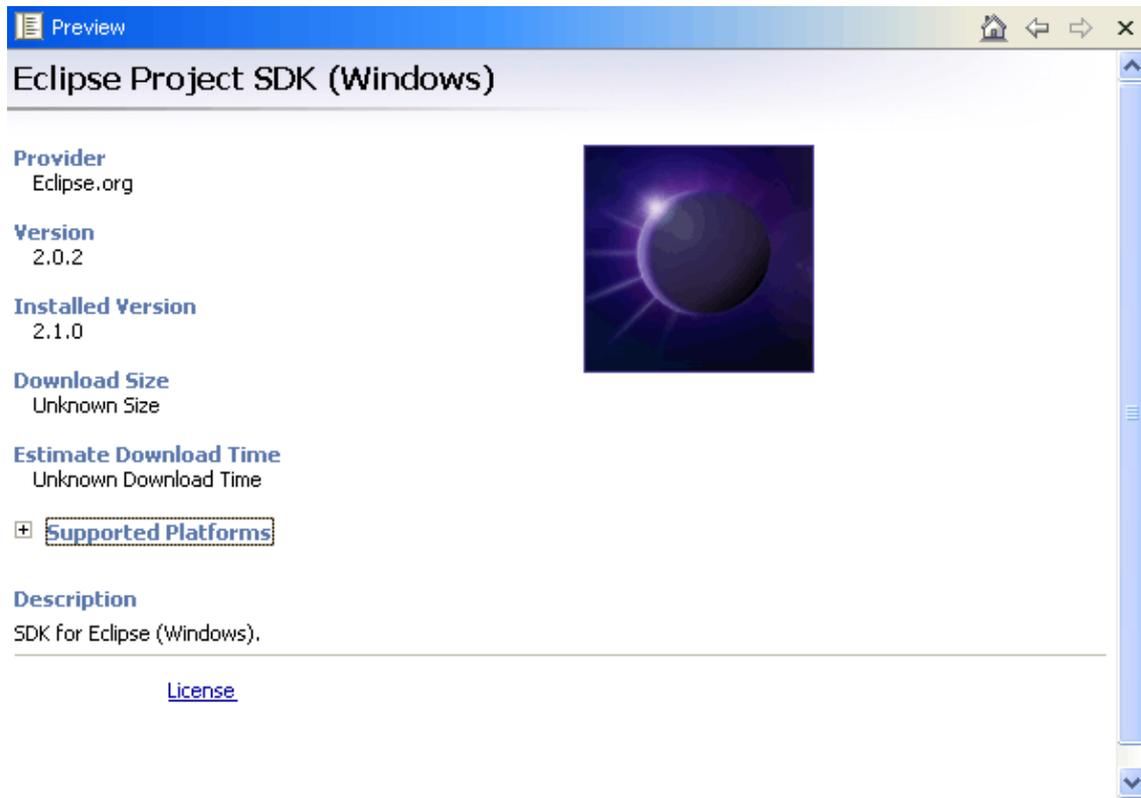
## Saving the Configuration

Configurations can be saved so that they can be restored at a later date. Saved configurations appear in **Saved Configurations** folder in the  [Install Configuration](#) view.

To save this configuration, press:



Preview for a selected feature displays relevant installation information about the feature. If an action can be taken against the selection, it shows as an action button in the preview.



Preview for a selected history displays relevant configuration information about the history entry. If an action can be taken against the selection, it shows as an action button in the preview.

Preview
🏠 ⬅️ ➡️ ✕

**Mar 19, 2003 2:45:46 PM**

**Created On**  
Mar 19, 2003 2:45:46 PM

**Current Configuration**  
No

**Activities**  
The following list shows activities that caused the creation of this configuration:

Date	Target	Action	Status
Mar 19, 2003 2:45:46 PM	file:D:/R21/screenshot/eclipse/	Site Installed	Success
Mar 19, 2003 2:45:46 PM	file:D:/R21/screenshot/eclip...fg.metadata/v2LocalSite.xml	Reconcile	Success

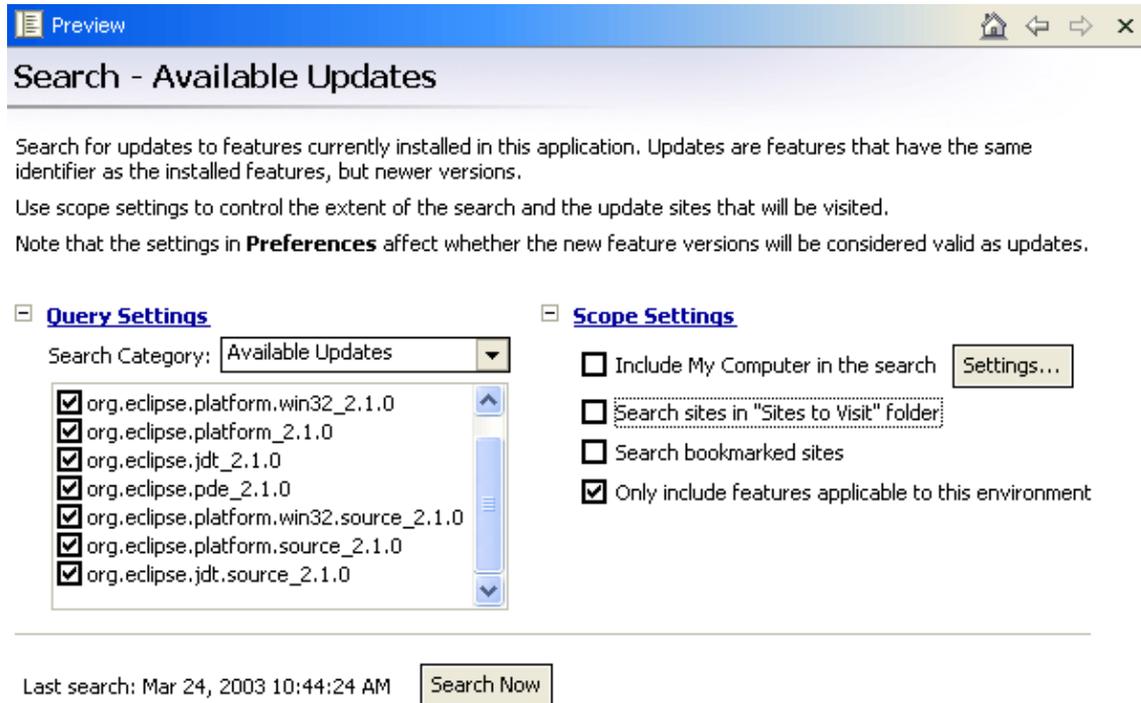
**Configuration Reversal**  
You can revert from the current to any of the compatible previous configurations. Configurations are compatible if they are not separated by a full system reconciliation in the timeline. Note that the reversal will be only partial if some of the features that are part of the desired configuration have since been deleted.

To restore this configuration, press:

**Saving the Configuration**  
Configurations can be saved so that they can be restored at a later date. Saved configurations appear in **Saved Configurations** folder in the [Install Configuration](#) view.

To save this configuration, press:

Preview for a search item displays relevant configuration information about the query. Query Settings and Scope Settings can be changed in the Preview. The Search Now button shown in the Preview triggers the search. The search results are presented in the preview.



Like other views in the Workbench, the Preview view has its own toolbar.

## Titlebar

### Go Home

This command shows the Welcome to Eclipse Update page.

### Go Back

This command goes back in the chain of Preview pages.

### Go Forward

This command goes forward in the chain of Preview pages.

#### ● Related concepts

[Features](#)

#### ● Related tasks

[Updating Eclipse with the update manager](#)

■ Related reference

[Install Configuration View](#)

[Feature Updates View](#)

[External Preview](#)

© Copyright IBM Corporation and others 2000, 2003.

# External Preview

The External Preview view, which is part of the Install/Update perspective, is an embedded browser used to display html information. This view is available only on **Windows** systems.

In general, the External Preview is used when links are followed to html pages.



Like other views in the Workbench, the External Preview view has its own toolbar.

## Titlebar

### Go Back

This command goes back in the chain of Preview pages.

### Go Forward

This command goes forward in the chain of Preview pages.

## Go

This command navigates to the specified External Preview pages.

## Stop

This command stops loading of the External Preview page.

## Refresh

This command reloads the External Preview page.

### ■ Related concepts

[Features](#)

### ■ Related tasks

[Updating Eclipse with the update manager](#)

### ■ Related reference

[Install Configuration View](#)

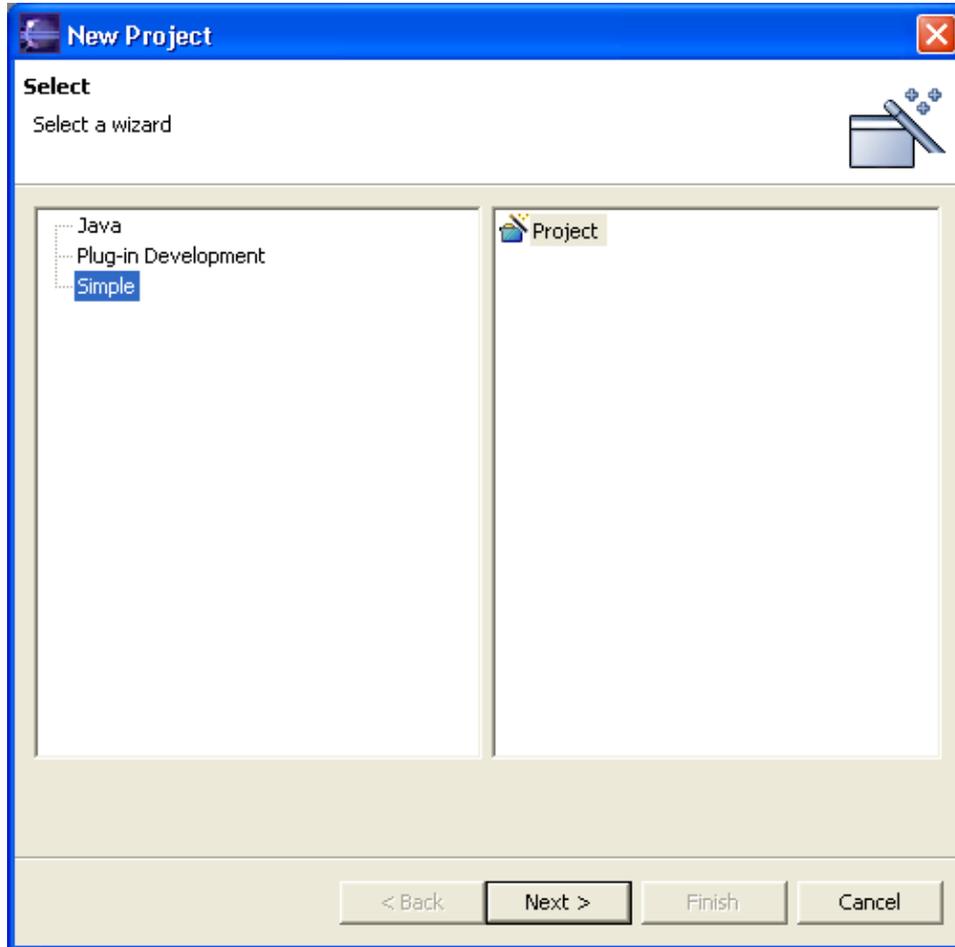
[Feature Updates View](#)

[Preview](#)

© Copyright IBM Corporation and others 2000, 2003.

# New Project wizard

This wizard helps you create a new project in the Workbench. When you first bring up the New Project wizard, you need to select the type of project you want to create. Select the Simple type if you want to create a generic project.



## Create a New Project Resource Page

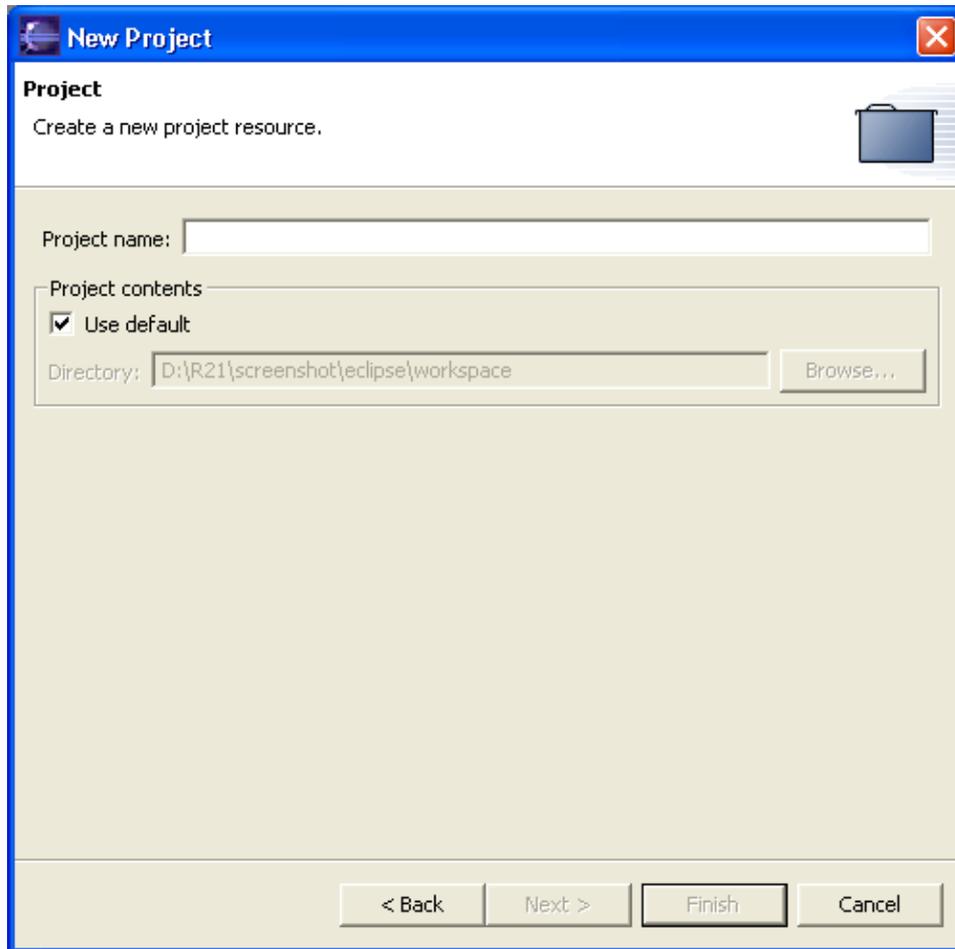
When you select Next, you will be presented with the New Project Resource Page, containing the following edit fields:

Field	Description	Default
Project Name	The name of the new project to be created.	<blank>
Location	The location in the file system where the project will be created. De-select "Use default location" to specify a location other than the default. You can type the new location or browse to select a file system location for the new project.	The workspace root directory

## Crash recovery

After you indicate a name and location for the project, you can either click Finish to create the project, or you can click Next to set up project dependencies on the Select Referenced Projects page.

Here is what the New Project Resource page looks like:

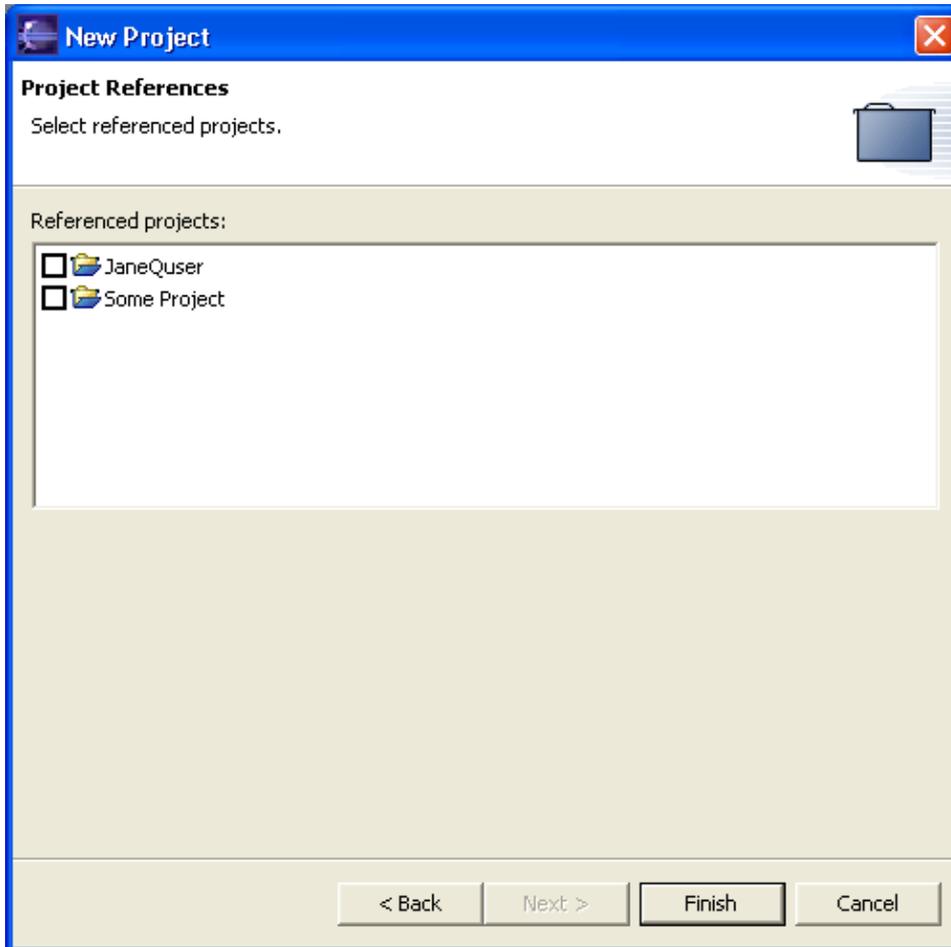


## Select Referenced Projects page

In the Referenced Projects list, you can set project dependencies for the new project. In the list of other projects in the Workbench, you can select one or more projects on which you want the new project to depend. Initially, no projects will be selected.

Click Finish when you are done to create the new project in the Workbench.

Here is what the Select Referenced Projects page looks like:



## New Project perspective options

On the Workbench preferences page (Window > Preferences > Workbench > Perspectives, New project options), you can change the way that new projects are initially displayed. For details on how to change this option see [Workbench](#).

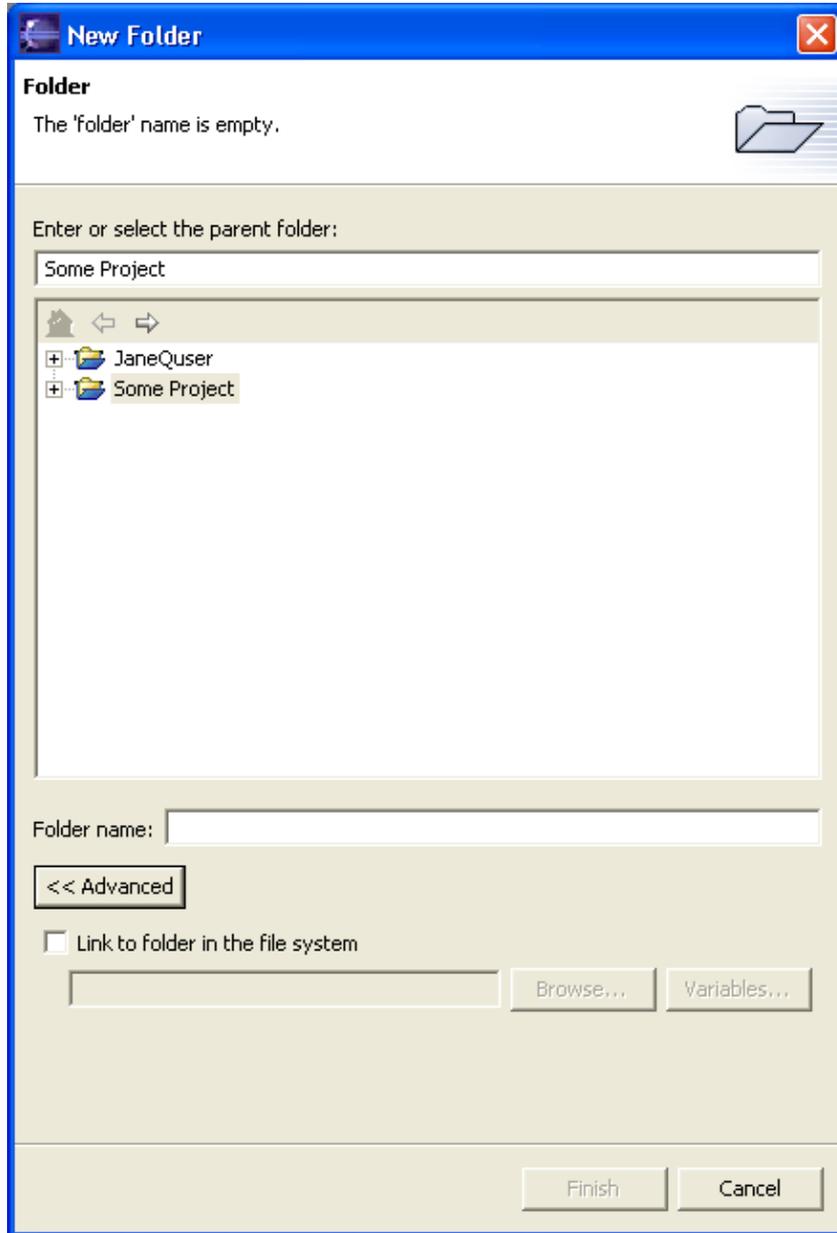
### ■ Related reference

[Navigator View](#)

# New Folder wizard

This wizard helps you create a new folder in the Workbench.

Here is what the New Folder wizard looks like:



## New Folder Fields

Field	Description	Default
Enter or select the parent folder	The resource in which the new folder will be created. Type or navigate the list to select the	The resource that was selected when you chose to create the new folder

	resource.	
Folder name	The name for the new folder.	<blank>

## Advanced

The **Advanced** button reveals or hides a section of the wizard used to create a linked folder. Check the **Link to folder in the file system** check box if you want the new folder to reference a folder in the file system. Use the field below the check box to enter a folder path or the name of a path variable. Use the **Browse...** button to browse for a folder in the file system. Use the **Variables...** button if you want to use a path variable to reference a file system folder.

### ■ Related concepts

[Linked resources](#)

[Path variables](#)

### ■ Related tasks

[Creating a folder](#)

[Creating linked resources](#)

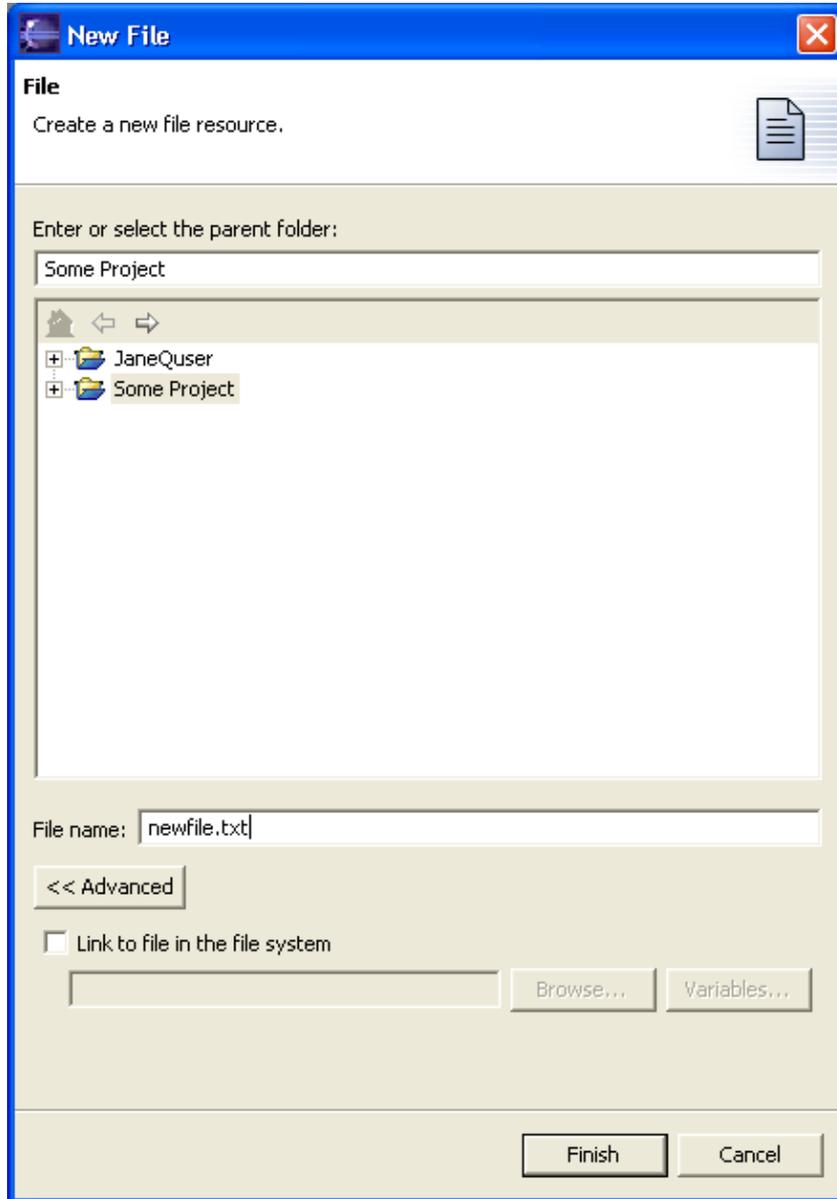
### ■ Related reference

[Navigator View](#)

# New File wizard

This wizard helps you create a new file in the Workbench.

Here is what the New File wizard looks like:



## New File Fields

Field	Description	Default
Enter or select the parent folder	The resource in which the new file will be created. Type or browse the list to select the resource.	The resource that was selected when you invoked the New File wizard.
File name		<blank>

	The name for the new file, including the file extension.	
--	--	--

## Advanced

The **Advanced** button reveals or hides a section of the wizard used to create a linked file. Check the **Link to file in the file system** check box if you want the new file to reference a file in the file system. Use the field below the check box to enter a file path or the name of a path variable. Use the **Browse...** button to browse for a file in the file system. Use the **Variables...** button if you want to use a path variable to reference a file system file.

### ■ Related concepts

[Linked resources](#)

[Path variables](#)

### ■ Related tasks

[Creating a file](#)

[Creating linked resources](#)

### ■ Related reference

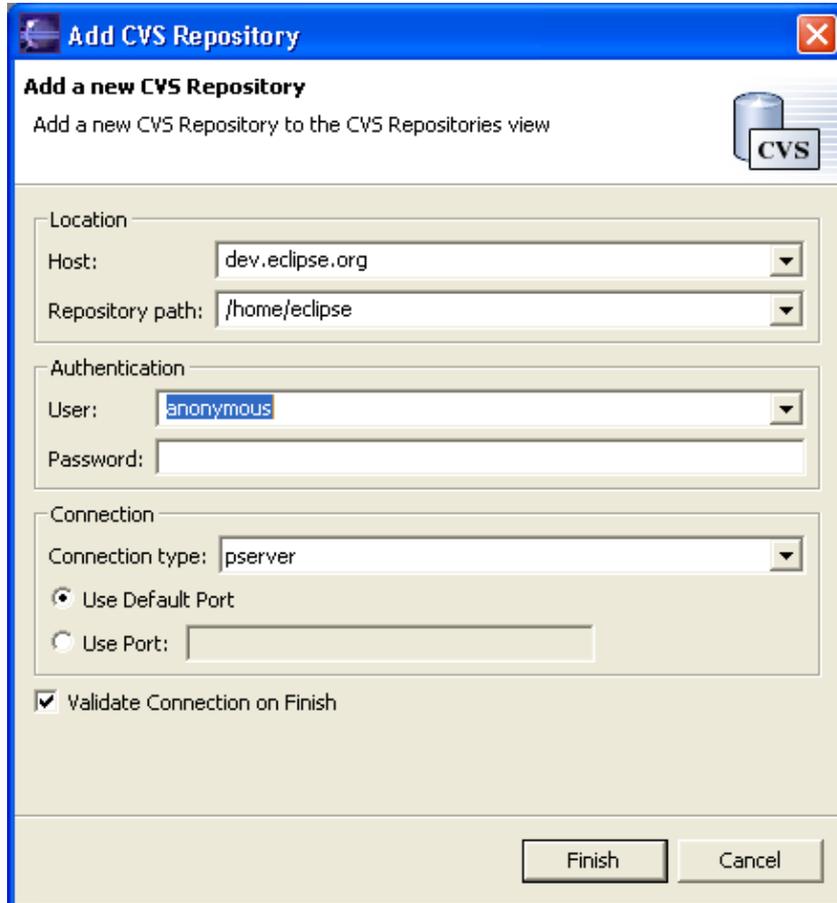
[Navigator View](#)

© Copyright IBM Corporation and others 2000, 2003.

# New CVS Repository Location wizard

This wizard helps you create a repository location.

Here is what the New CVS Repository Location wizard looks like:



The screenshot shows a dialog box titled "Add CVS Repository" with a close button in the top right corner. The main title is "Add a new CVS Repository" and the subtitle is "Add a new CVS Repository to the CVS Repositories view". There is a CVS logo icon in the top right. The dialog is divided into three sections: "Location", "Authentication", and "Connection".

**Location:**

- Host: dev.eclipse.org
- Repository path: /home/eclipse

**Authentication:**

- User: anonymous
- Password: (empty text field)

**Connection:**

- Connection type: pserver
- Use Default Port
- Use Port: (empty text field)

Validate Connection on Finish

Buttons: Finish, Cancel

# CVS repository location

The following are the editable fields for a new CVS repository location:

Option	Description	Default
Host	The address of the host (e.g., "hostmachine.com").	<blank>
Repository Path	The path to the repository on the host (e.g., "/x/y/z", "d:/myrepo").	<blank>
User Name	The user name under which you want to connect to the server.	<blank>
Password	The password required for the above user name to access the above host.	
Connection Type	The type of CVS connection for the repository, either "pserver", "extssh" or "ext".	pserver
Port	Option to use a custom port for the connection.	Use Default Port
Validate connection on finish	Check this option if you want to attempt a connection to the host upon completion of the wizard to validate that the entered information is correct. If this option is not enabled, you will not know whether the information entered is correct until you try to access the contents of a repository for the first time.	On

## ■ Related reference

[CVS Repositories View](#)

© Copyright IBM Corporation and others 2000, 2003.

# Merge wizard

This wizard helps you merge changes between two states of a project into your workspace. Often the Merge Wizard is used to move changes from one branch into another, for example after splitting a branch to work on a bug fix. The merge operation takes changes between two points in a branch, the **start point** and the **end point**, and merges them into your workspace. Typically the start point will be the root of a branch (version tag) and the end point can either be the tip (latest and best) of the branch or another version tag.

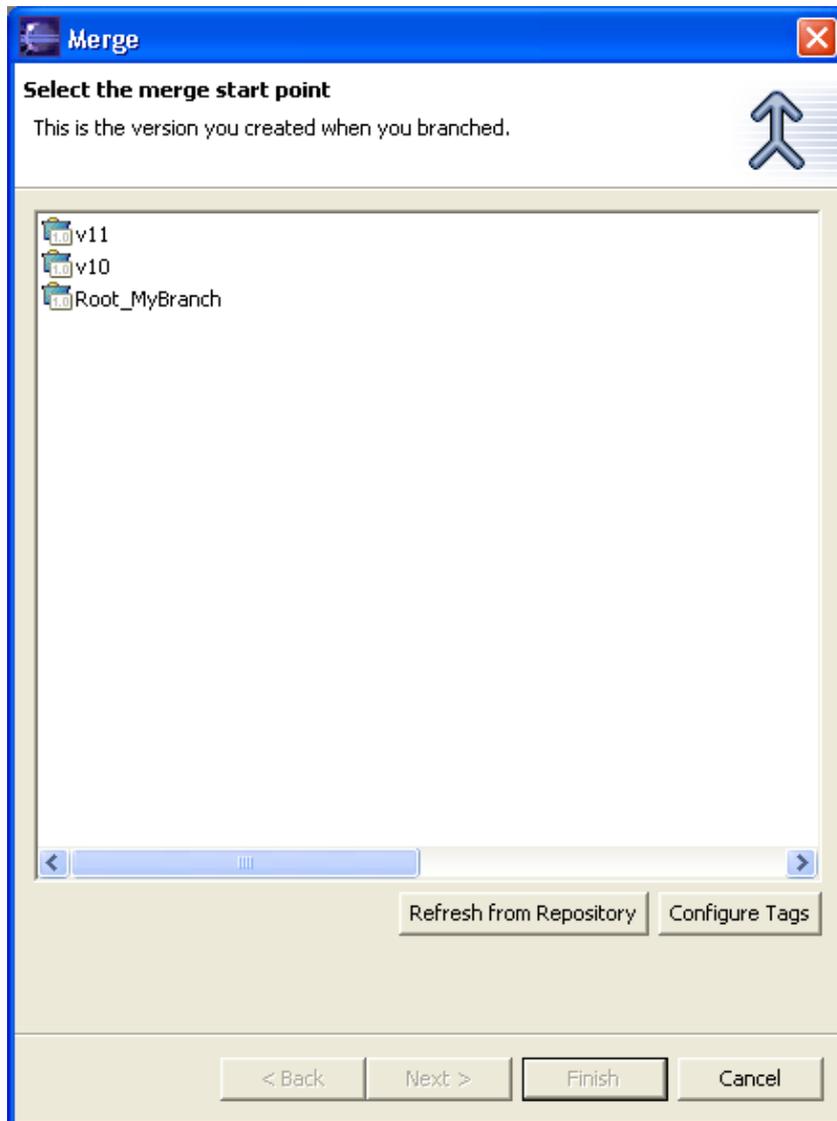
It is very important to understand that the destination of the merge is always the project in your workspace. After the merge has been completed you can test the changes locally and then commit them to the new branch (usually HEAD).

To start a merge, select a project (or one or more resources), and select **Team > Merge...** from the pop-up menu.

## Start Point

This is where you will select the starting point of the merge.

From the list, select a version of the project that will be the starting (base) point of the merge operation.



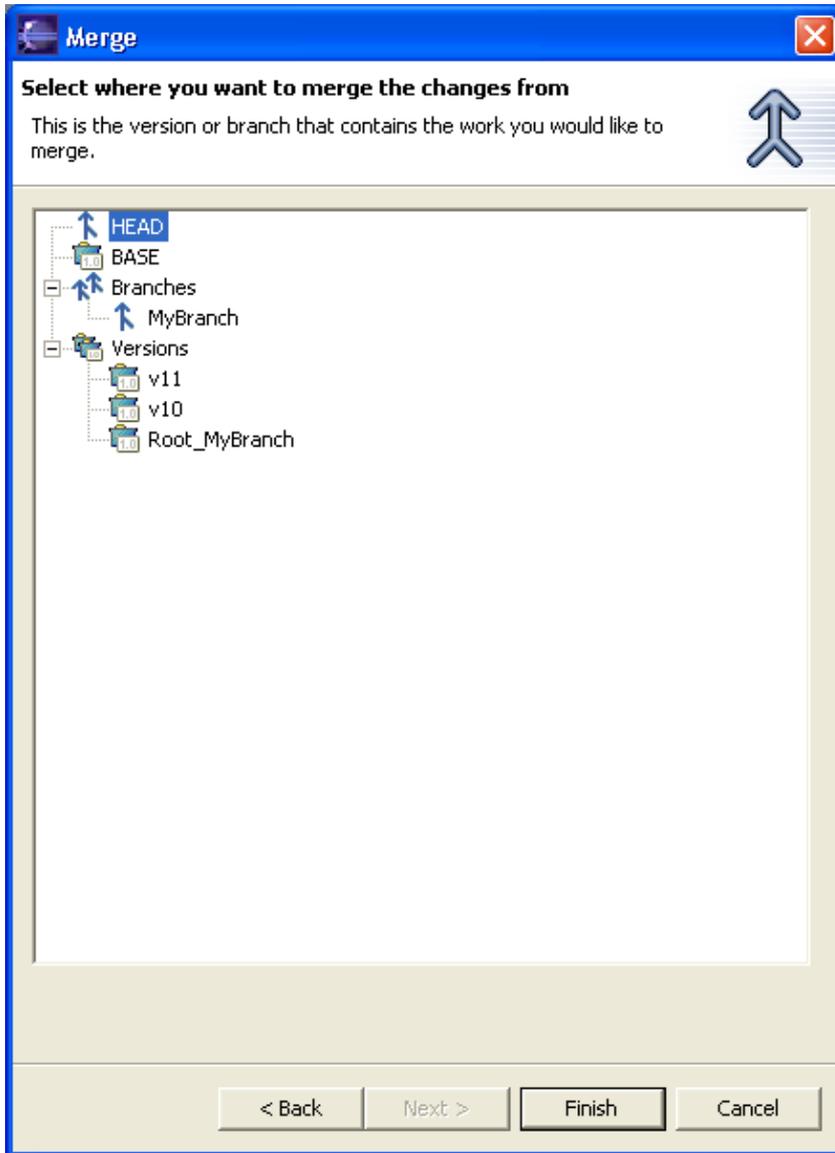
## End state

This is where you will choose the end state of the merge.

From the list, select either:

- A version of the project
- The branch

This will be the end point of the merge operation. Choose a version when you want to merge the differences between two versions of a project into your workspace. Choose a branch if you want to merge the changes made in the branch into your local workspace.



After the finish button is pressed the changes between the start point and end point are calculated and a merge editor is displayed. Depending on the size of the project you are merging, this may take some time. In the merge editor you can update or merge changes into your Workbench. You cannot commit from the merge view.

#### ■ Related concepts

[Team programming with CVS](#)

[Branches](#)

#### ■ Related tasks

[Branching](#)

[Merging from a branch](#)

#### ■ Related reference

[CVS](#)

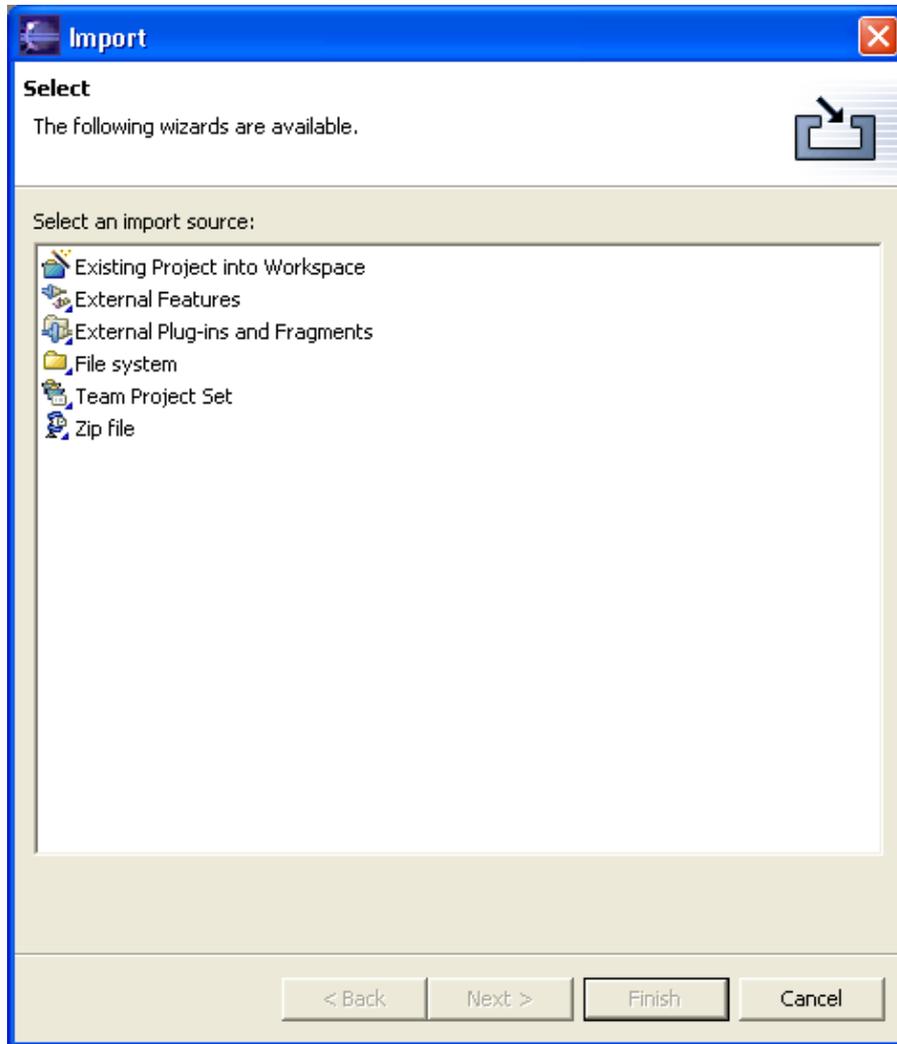
Compare editor

© Copyright IBM Corporation and others 2000, 2003.

# Import wizard

This wizard helps you import resources into the Workbench.

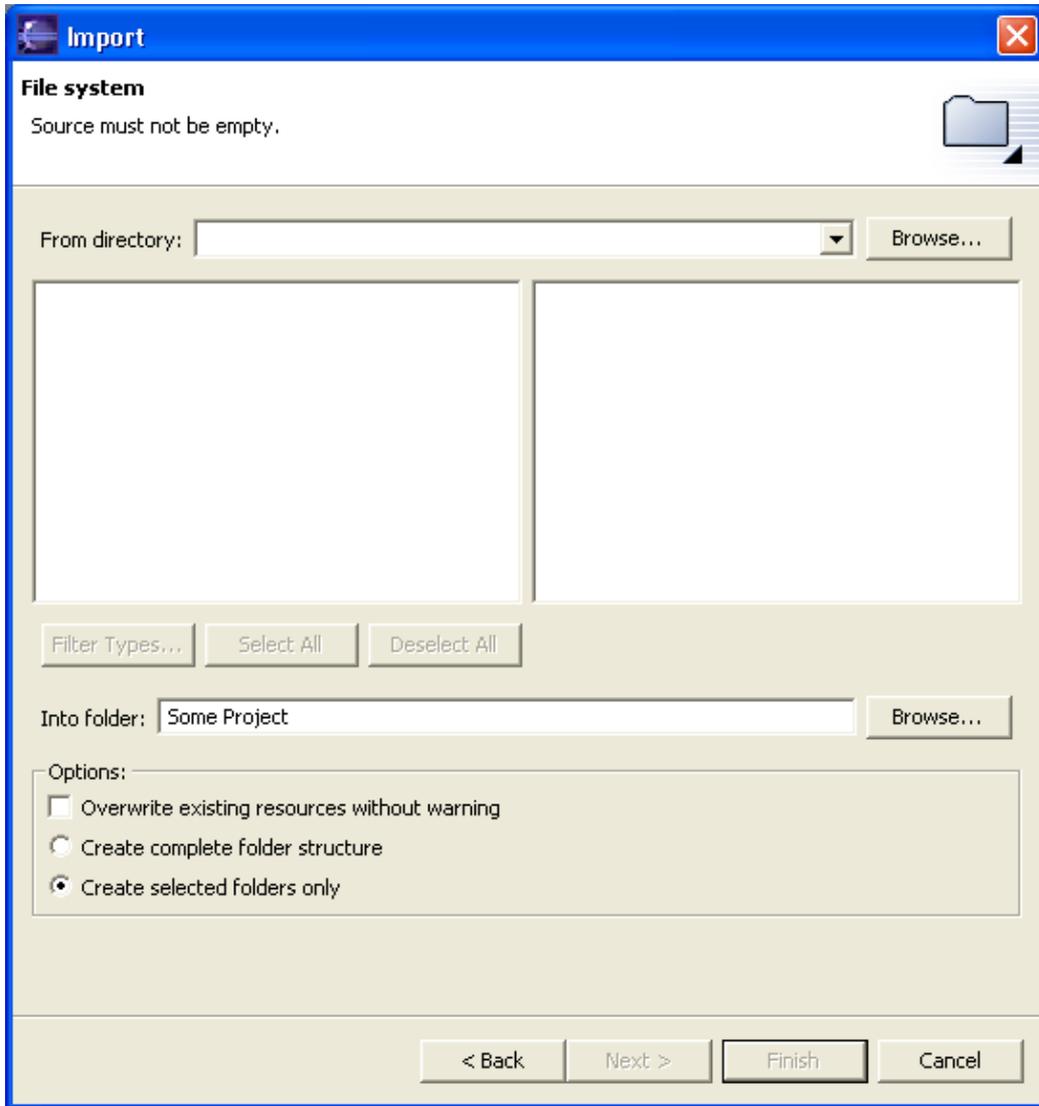
When the Import wizard first comes up, you must choose what type of import to do:



## File System

If you choose this option, you will import files from the file system.

## Crash recovery



### Import – File System Options

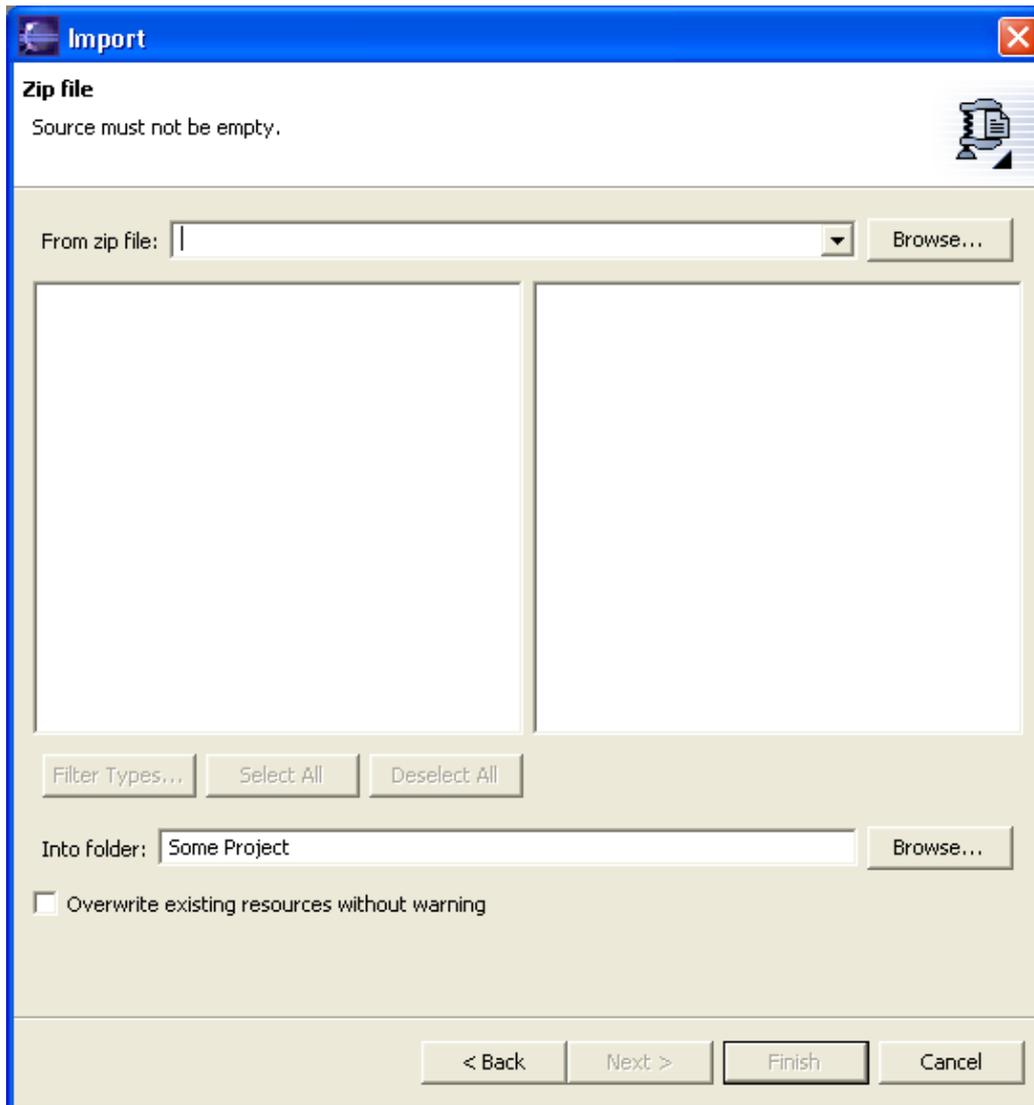
Option	Description	Default
Directory	The directory from which to import files. Select a previous path from the drop down combo or Browse to select the path in the file system.	<blank>
Filter Types	Dialog to select which file types to import. Use this to restrict the import to only certain file types.	N/A
Select All	Check off all files and folders for import.	N/A
Deselect All	Uncheck all resources.	N/A
Folder	The folder into which the resources will be imported. Type the path or Browse to select a path in the Workbench.	The folder holding the selected resource
Overwrite existing resources without	Determines whether importing a resource should silently overwrite a resource which already exists in the Workbench. If this option is off, you	Off

## Crash recovery

warning	will be prompted before a given resource is overwritten, in which case you can either overwrite the resource, skip it, or cancel the import.	
Create complete folder structure	Create hierarchy (folder) structure in the Workbench to accommodate the resources being imported, and all parent folders of those resources in the file system.	Off
Create selected folders only	Create hierarchy (folder) structure in the Workbench to accommodate the resources being imported.	On

## ZIP File

If you choose this option, you will import files from an archive (ZIP) file.



Import – Zip File Options

Option	Description	Default
--------	-------------	---------

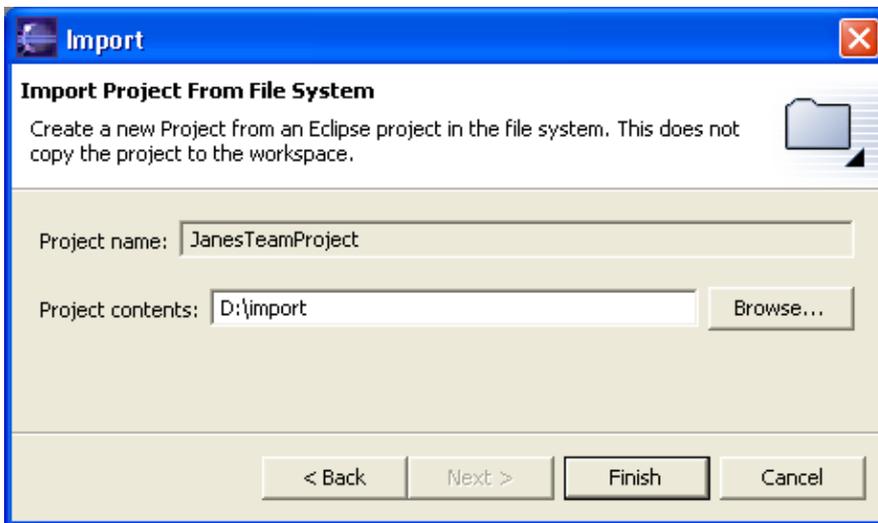
ZIP File

## Crash recovery

ZIP File	The file from which to import. Type in the full path or Browse to select the path on the file system.	<blank>
Filter Types...	Dialog to select which file types to import. Use this to restrict the import to only certain file types.	N/A
Select All	Check off all resources for import	N/A
Deselect All	Uncheck all resources.	N/A
Folder	The folder into which the resources will be imported. Type the path or Browse to select a path in the Workbench.	The folder holding the selected resource
Overwrite existing resources without warning	Determines whether importing a resource should silently overwrite a resource which already exists in the Workbench. If this option is off, you will be prompted before a given resource is overwritten, in which case you can either overwrite the resource, skip it, or cancel the import.	Off

## Existing Project

Imports a project into this workspace that was previously located in this workspace, or that currently exists in another workspace.



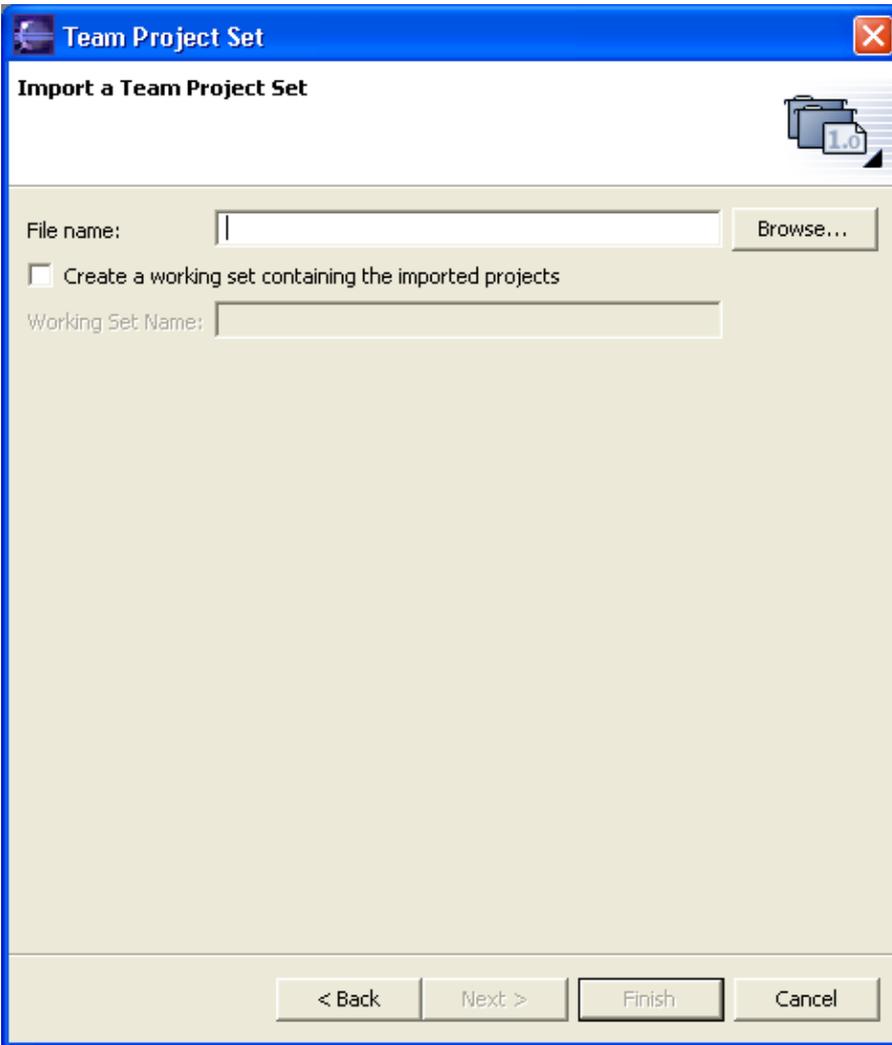
Import – Existing Project Options

Option	Description	Default
Project Contents	The location in the file system where the existing project resides.	<blank>

## Team Project Set

Imports a description of the repository and version control information for a set of projects. This allows you to synchronize correctly with the appropriate project state in your repository.

## Crash recovery



### Import – Team Project Options

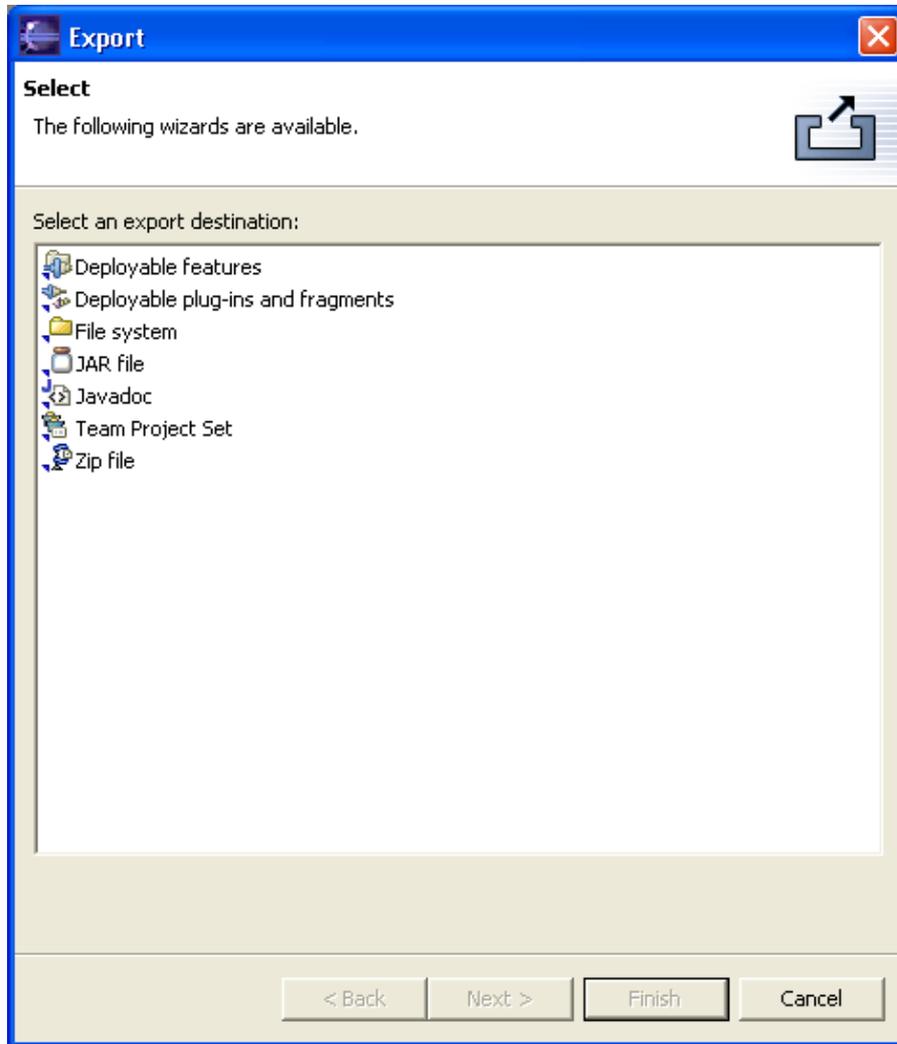
Option	Description	Default
File name	The name of the team project set export file	<blank>

© Copyright IBM Corporation and others 2000, 2003.

# Export wizard

This wizard help you export resources from the Workbench.

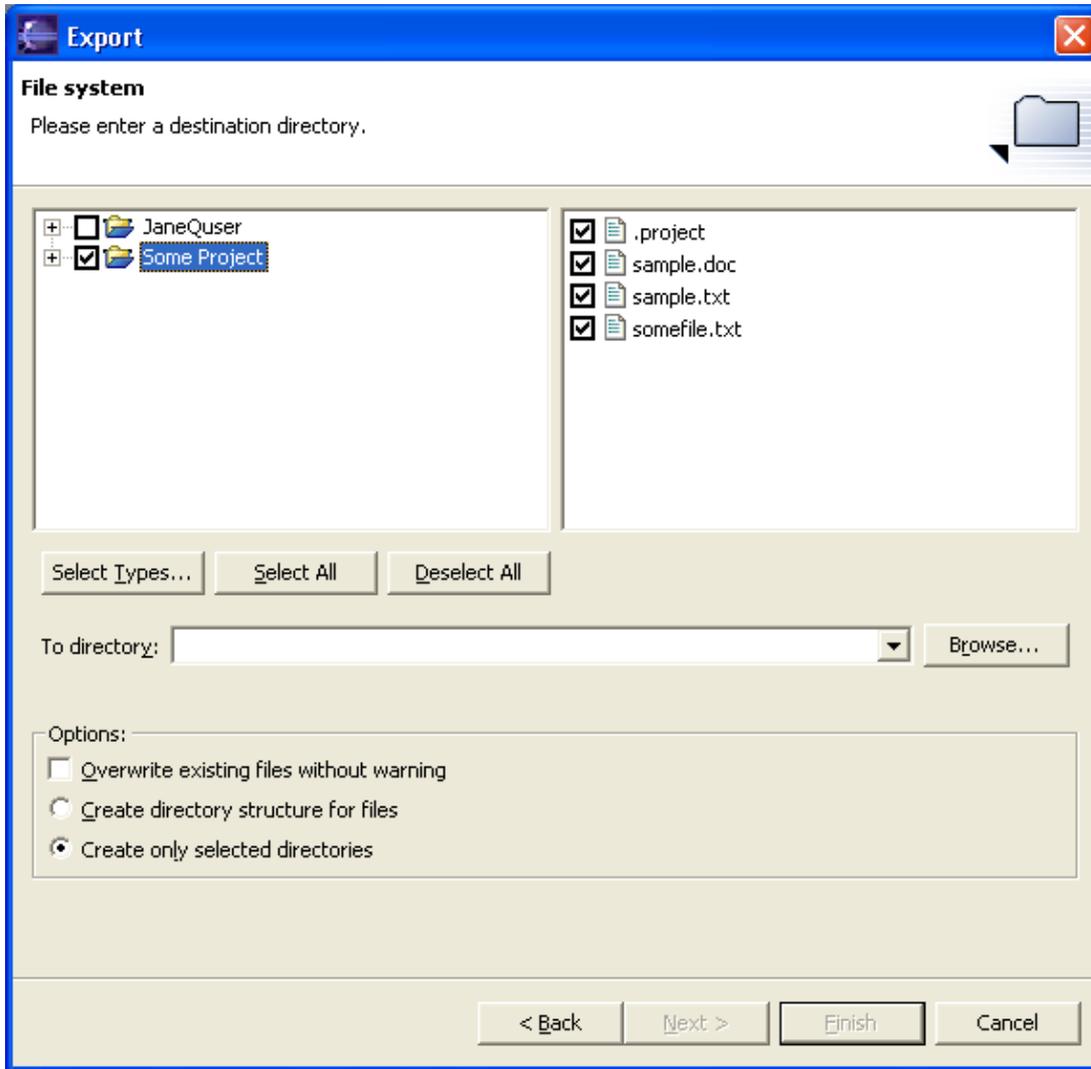
When the Export wizard first comes up, you must choose what type of export to do:



## File System

If you choose this option, you will export files to the file system.

## Crash recovery



### Export – File System Options

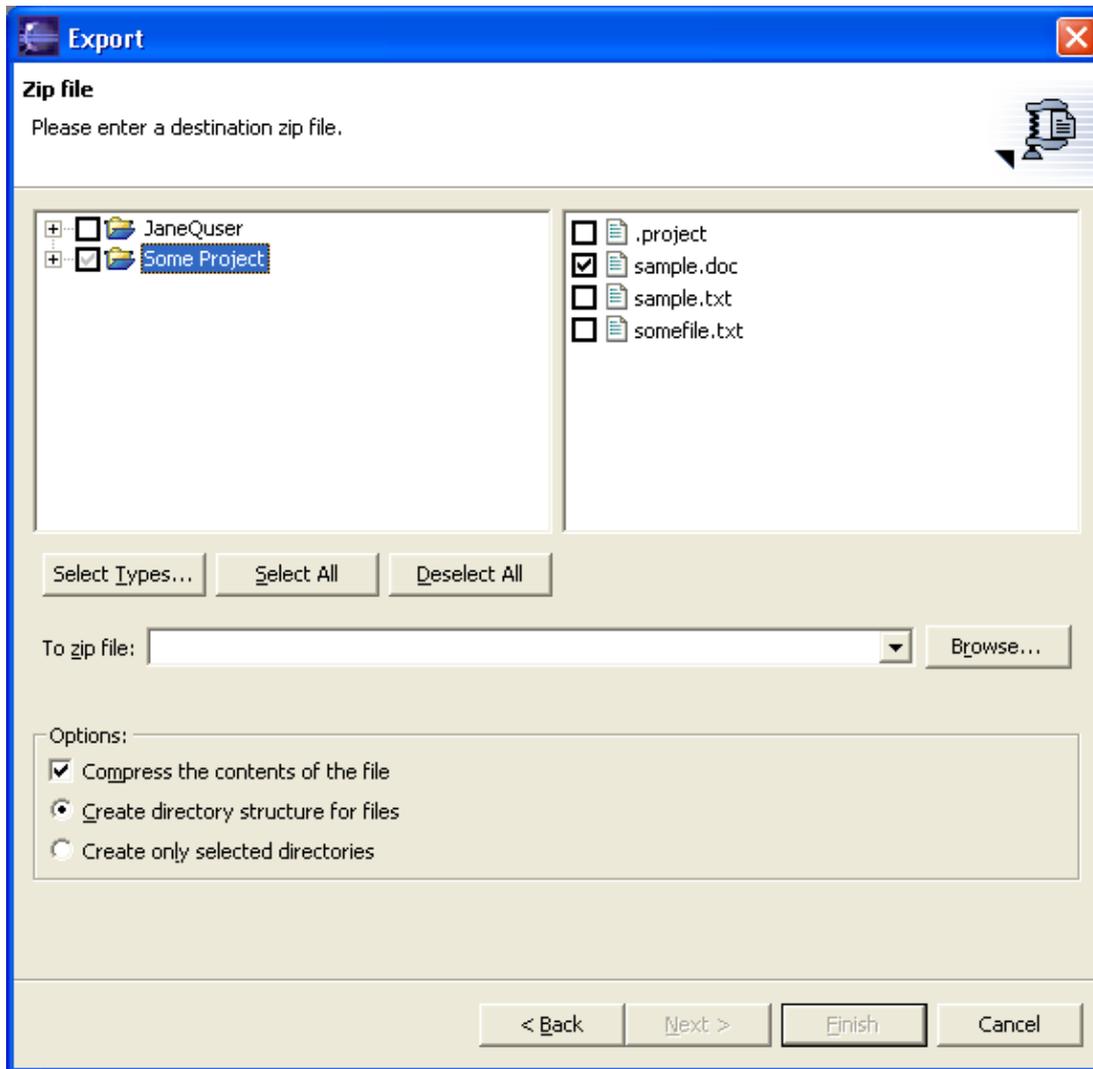
Option	Description	Default
Select resources to export	The project (and resources within that project) to export to the file system.	The project holding the selected resource
Select Types...	Dialog to select which file types to export. Use this to restrict the import to only certain file types.	N/A
Select All	Checks off all resources for export.	N/A
Deselect All	Uncheck all resources.	N/A
Directory	The directory on the file system into which the resources will be exported. Type the path, select a previous export path from the drop down list, or Browse to select a path.	The directory of the last export, or <blank>
Overwrite existing files without warning	Determines whether exporting a resource should silently overwrite a resource which already exists in the file system. If this option is off, you will be prompted before a given file is overwritten, in which case you	Off

## Crash recovery

	can either overwrite the file, skip it, or cancel the export.	
Create directory structure for files	Create hierarchy (folder) structure in the file system as it exists in the Workbench.	Off
Create only selected directories	Create hierarchy (folder) structure in the file system only for selected folders.	On

## ZIP File

If you choose this option, you will export files to an archive (ZIP) file.



### Export – ZIP File Options

Option	Description	Default
Select resources to export	The project (and resources within that project) to export to an archive.	The project holding the

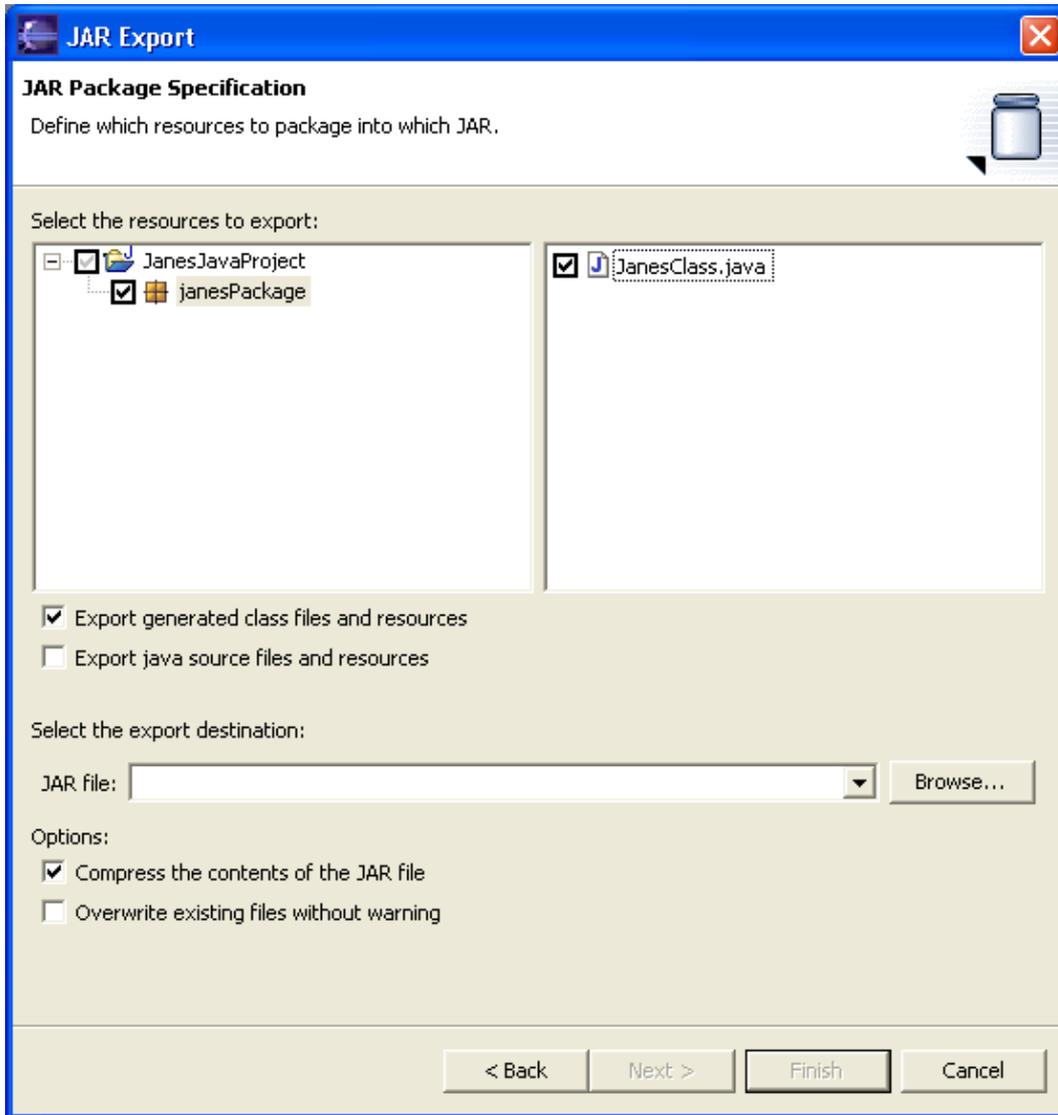
## Crash recovery

		selected resource
Select Types...	Dialog to select which file types to export. Use this to restrict the import to only certain file types.	N/A
Select All	Check off all resources for export.	N/A
Deselect All	Uncheck all resources.	N/A
ZIP File	The path and name of an archive file into which the resources will be exported. Type the path, select a previous path from the drop down list, or Browse to select a path and file name on the file system.	The ZIP file of the previous export, or <blank>.
Compress the contents of the file	Compresses the contents (resources selected to be exported) in the archive that is created.	On
Overwrite existing file without warning	If the specified archive already exists in the file system, you will be prompted to overwrite the file. If you do not want to be prompted turn this option on.	Off
Create directory structure for files	Create hierarchy (folder) structure in the file system as it exists in the Workbench.	Off
Create only selected directories	Create hierarchy (folder) structure in the file system only for selected folders.	On

## Jar File

If you choose this option, you will export files to an JAR file.

## Crash recovery



### Export – JAR File Options

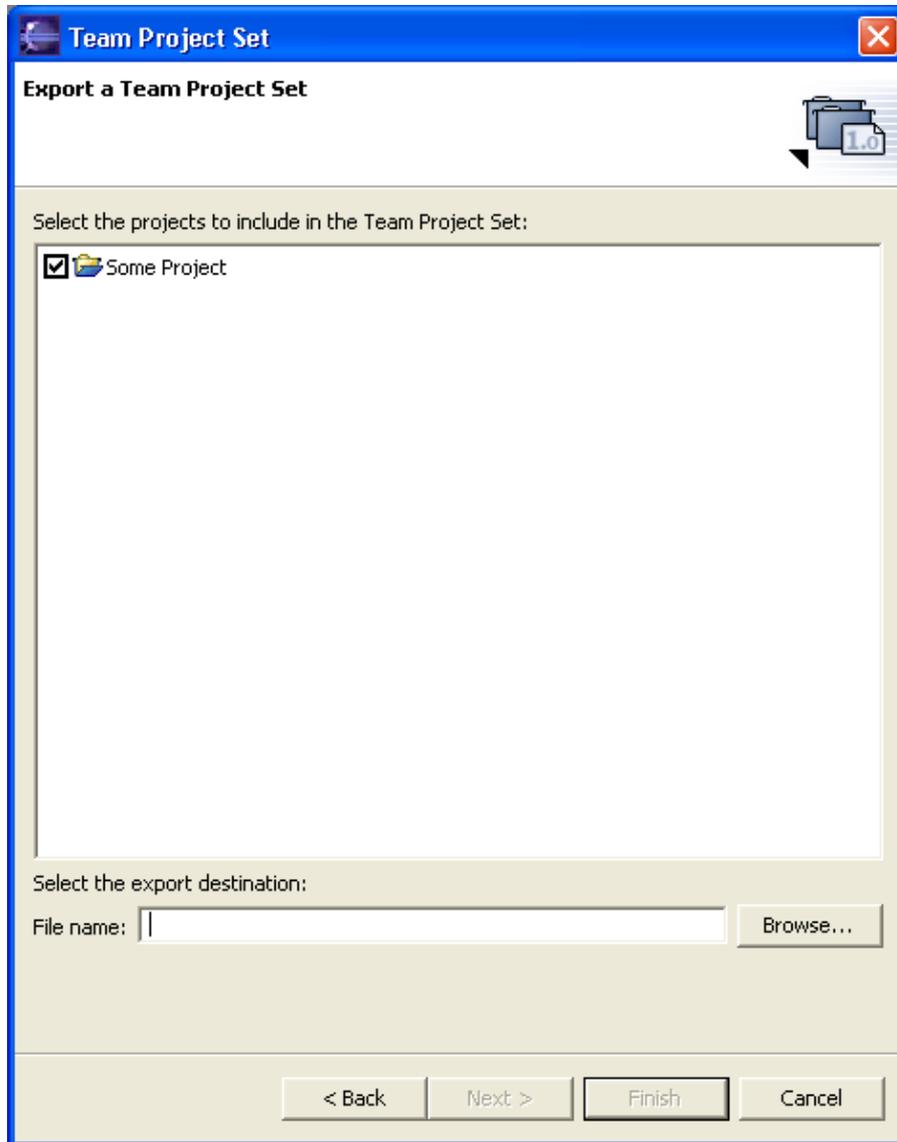
Option	Description	Default
Select packages to export	The project (and packages within that project) to export to a JAR file.	The project holding the selected resource
Export generated class files and resources	Export the generated bytecodes (.class files) and resource files contained in the selected packages and projects.	On
Export java source files and resources	Export the java source files (.java) and resource files contained in the selected packages and projects.	Off
Jar File	The path and name of a JAR file into which the resources will be exported. Type the path or Browse to select a path and file name on the file system.	The JAR file of the previous export, or <blank>.
Compress the contents of the JAR	Compresses the contents (resources selected to be exported) in the JAR file that is created.	On

## Crash recovery

file		
Overwrite existing files without warning	If the specified JAR file already exists on the file system, you will be prompted to overwrite the file. If you do not want to be prompted turn this option on.	Off

## Team Project Set

Exports a description of the repository and version control information for a set of projects. This allows you to synchronize those projects correctly in a different workspace.



### Export – Team Project Set Options

Option	Description	Default
Select projects	The project(s) to export to the team project set file.	The selected projects

## Crash recovery

File name	The path and name of the file to export to.	<blank>
-----------	---	---------

© Copyright IBM Corporation and others 2000, 2003.

# Workbench User Guide

The Help view displays help related to using the Workbench. See [Help View](#) on how to navigate through the contents of the Help view.

If you select Workbench User Guide from the drop-down menu of the Help view, you will see help topics related to using the Workbench in the Contents tab. The Workbench User Guide is broken down into four main sections, described below.

## Getting started

This section contains tutorials that will help you when you start using the Workbench.

## Concepts

Concepts are high level descriptions of the schema and functions of the Workbench. This section helps provide a general understanding of how the Workbench functions. For example, the Concepts section includes a discussion of what perspectives and views are and how they relate to one another.

## Tasks

Task descriptions are step by step instructions for performing specific actions and tasks in the Workbench. For example, the Tasks section contains step by step instructions for creating a repository location, and for importing a file from the file system into the Workbench.

## Reference

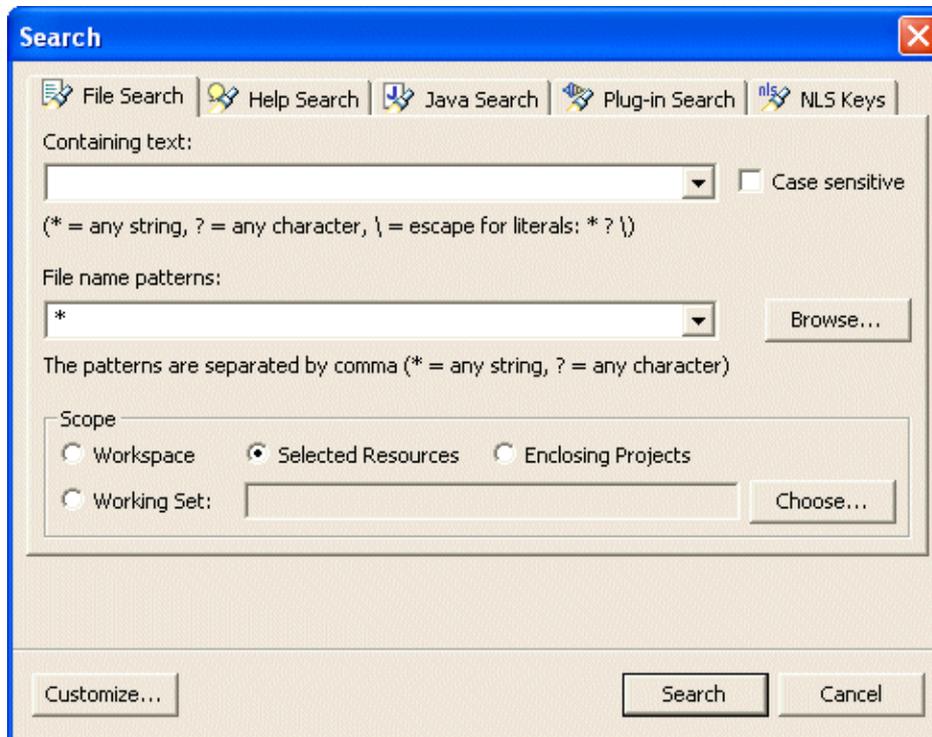
Reference materials are helpful resources that will assist you while you are using the Workbench. This includes descriptions of various wizards, dialogs, and fields as well as Workbench resources such as specific views and perspectives.

The Reference section also includes a glossary of terms that you might find useful while using the Workbench.

© Copyright IBM Corporation and others 2000, 2003.

# File search

In the Search Dialog, the File Search tab allows you to search for files or text in the Workbench. You can bring up the Search Dialog by clicking on the Search toolbar button.



## Containing text

Type the expression for which you wish to do the text search. Leave this field empty to search for files.

From the drop-down menu, you can choose to repeat or modify a recent search.

## Wildcards

The available wildcards for search expressions are displayed in the search dialog:

- "\*" matches any set of characters, including the empty string
- "?" matches for any character
- "\" is the escape for a literal; if you want to search for an asterisk, question mark, or backslash character, type a backslash before it to indicate that you are not using these characters as wildcards (e.g., "\\\*", "\\?", or "\\")

## File name patterns

In this field, enter all the file name patterns for the files to find or search through for the specified expression.

## Wildcards

The available wildcards for file name patterns are displayed in the search dialog:

- "\*" matches any set of characters, including the empty string
- "?" matches for any character

## Case sensitive

Turn this option on if you want the text search to be case sensitive.

## Scope

Choose the scope of your search. You can either search the whole workspace, pre-defined working sets, previously selected resources or projects enclosing the selected resources.

### Related concepts

[Search view](#)

### Related tasks

[Searching for files](#)

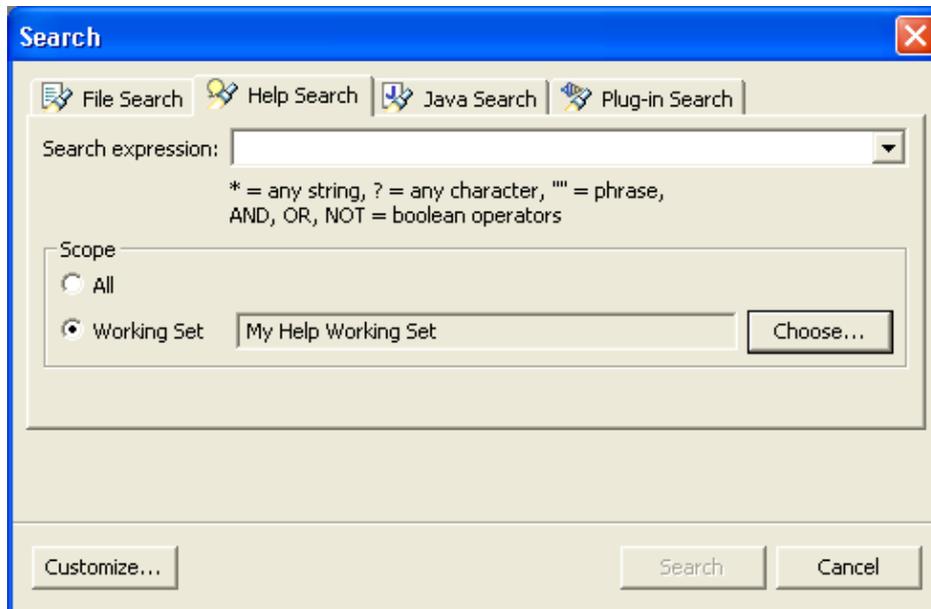
[Searching for text within a file](#)

### Related reference

[Search view](#)

# Help search

In the Search Dialog, the Help Search tab allows you to search on–line documentation. You can bring up the Search Dialog by choosing **Search > Help** from the menu.



## Search Expression

Type the expression for which you wish to do the help search.

From the drop–down menu, you can choose to repeat or modify a recent search.

## Advanced Search options

Use following expression rules when creating and advanced query:

- "\*" matches any set of characters, including the empty string
- "?" matches for any character
- Double quotation marks around terms indicate terms should be treated as a phrase
- AND operator requires terms on both sides of the operator to exist in the document; If no operator is specified between terms, there is an implicit AND between them
- OR operator requires term on either of the operator to exist in the document
- NOT operator in front of a term requires the term not to exist in the document

## Scope

Choose the scope of your search. You can either search all topics, or pre–defined working set.

### ■ Related concepts

[Search view](#)

[Working sets](#)

## Crash recovery

- Related tasks

[Searching online help](#)

- Related reference

[Search view](#)

© Copyright IBM Corporation and others 2000, 2003.

# File menu

This menu allows you to create, save, close, print, import, and export Workbench resources and exit the Workbench itself.

## New

This command creates new resources.

## Close

This command closes the active editor.

## Close All

This command closes all open editors.

## Save

This command allows you to save the contents of the active editor.

## Save As

This command allows you to save the contents of the active editor under another file name or location.

## Save All

This command saves the contents of all open editors.

## Revert

This command replaces the contents of the active editor with the previously saved contents.

## Move

This command moves the currently selected resources to a different location.

## Rename

This command changes the name of the currently selected resource.

## Refresh

This command refreshes the resource with the contents in the filesystem.

## **Print**

This command prints the contents of the active editor.

## **Import**

This option launches the import wizard, which allows you to add resources to the Workbench.

## **Export**

This option launches the export wizard, which allows you to export resources from the Workbench.

## **Properties**

This command opens the properties dialog for the currently selected resource.

## **Recent file list**

A list of the most recently accessed files in the Workbench is maintained at the bottom of the File menu. Any of these files can be opened from the File menu by simply selecting the file name.

## **Exit**

This command closes and exits the Workbench.

© Copyright IBM Corporation and others 2000, 2003.

# Edit menu

This menu helps you manipulate resources in the editor area.

## Undo

This command reverses your most recent editing action.

## Redo

This command re-applies the editing action that has most recently been reversed by the Undo action.

## Cut

This command removes the selection and places it on the clipboard.

## Copy

This command places a copy of the selection on the clipboard.

## Paste

This command places the text or object on the clipboard at the current cursor location in the currently active view or editor.

## Delete

This command removes the current selection.

## Select All

This command selects all text or objects in the currently active view or editor.

## Find/Replace

This command allows you to search for an expression in the active editor, and optionally replace the expression with a new expression.

## Find Next

This command allows you to search for the next occurrence of the current selection, or for the next occurrence of the most recent expression found using the Find/Replace action.

## Find Previous

This command allows you to search for the previous occurrence of the current selection, or for the previous occurrence of the most recent expression found using the Find/Replace action.

## Incremental Find Next

This command allows you to search for expressions in the active editor. As you type the search expression, it will incrementally jump to the next exact match in the active editor. While in this mode, the up and down cursor keys can be used to navigate between matches, and the search can be cancelled by pressing left or right cursor keys, the enter key, or the escape key.

## Incremental Find Previous

This command allows you to search for expressions in the active editor. As you type the search expression, it will incrementally jump to the previous exact match in the active editor. While in this mode, the up and down cursor keys can be used to navigate between matches, and the search can be cancelled by pressing left or right cursor keys, the enter key, or the escape key.

## Add Bookmark

This command adds a bookmark in the active file on the line where the cursor is currently displayed.

## Add Task

This command adds a task in the active file on the line where the cursor is currently displayed.

## Convert Line Delimiters

This submenu allows you to convert the line delimiters used in the active editor to a different format.

## Encoding

This submenu allows you to change the file encoding used to read and write the file in the active editor. These actions are only available when the current editor contents are saved.

# Navigate menu

This menu allows you to locate and navigate through resources and other artifacts displayed in the Workbench.

## Go Into

This command refocuses the active view so that the current selection is at the root. This allows web browser style navigation within hierarchies of artifacts.

## Go To

- **Back:** This command displays the hierarchy that was displayed immediately prior to the current display. For example, if you Go Into a resource, then the Back command in the resulting display returns the view to the same hierarchy from which you activated the Go Into command. This command is similar to the Back button in an HTML browser.
- **Forward:** This command displays the hierarchy that was displayed immediately after the current display. For example, if you've just selected the Back command, then selecting the Forward command in the resulting display returns the view to the same hierarchy from which you activated the Back command. This command is similar to the Forward button in an HTML browser.
- **Up one level:** This command displays the hierarchy of the parent of the current highest-level resource.
- **Resource:** This command allows you to navigate quickly to a resource. For more information see the links to related tasks below.

## Open Resource

This command displays a dialog that lets you select any resource in the workspace to open it in an editor. For more information see the links to related tasks below.

## Show In

This sub-menu is used to find and select the currently selected resource in another view. If an editor is active, these commands are used to select the resource currently being edited in another view.

## Next

This command navigates to the next item in a list or table in the active view. For example, when the search results view is active, this navigates to the next search result.

## Previous

This command navigates to the previous item in a list or table in the active view. For example, when the search results view is active, this navigates to the previous search result.

## Go to Line

This command allows you to jump to a specific line in the active editor.

## Back

This command navigates to the previous resource that was viewed in an editor. Analogous to the **Back** button on a web browser.

## Forward

This command navigates to undo the effect of the previous **Back** command. Analogous to the **Forward** button on a web browser.

### ■ Related tasks

[Finding a resource quickly](#)

© Copyright IBM Corporation and others 2000, 2003.

# Project menu

The Project menu allows you to perform actions (builds or compilations) on projects in the Workbench.

## Open Project

This command opens the currently selected project or projects. The selected projects must currently be closed for this command to be available.

## Close Project

This command closes the currently selected project or projects. The selected projects must be currently open for this command to be available. Closing a project will remove all of that project's state from memory, but the contents on disk are left untouched.

## Build Project

This command performs an incremental build on the currently selected project. That is, it builds (compiles) all resources in the project that are affected by any resource changes since the last build. This command is only available if auto-build is turned off. Auto-build is turned off via **Window > Preferences > Workbench > Perform build automatically on resource modification**.

## Rebuild Project

This command performs a full build of the selected project. That is, it builds (compiles) all resources in the project from scratch.

## Build All

This command performs an incremental build on all projects in the Workbench. That is, it builds (compiles) all resources in the Workbench that are affected by any resource changes since the last incremental build. This command is only available if auto-build is turned off. Auto-build is turned off via **Window > Preferences > Workbench > Perform build automatically on resource modification**.

## Rebuild All

This command performs a full build. That is, it builds (compiles) all resources in the Workbench from scratch.

## Properties

This command opens a dialog showing the properties of the selected project or of the project that contains the selected resource.

© Copyright IBM Corporation and others 2000, 2003.

# Window menu

This menu allows you to display, hide, and otherwise manipulate the various views, perspectives, and actions in the Workbench.

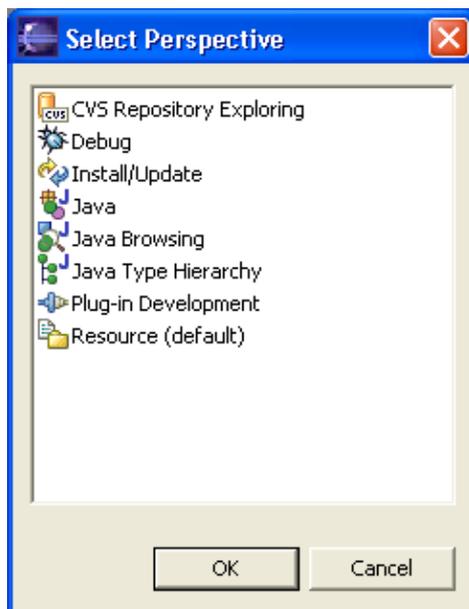
## New Window

This command opens a new Workbench window with the same perspective as the current perspective.

## Open Perspective

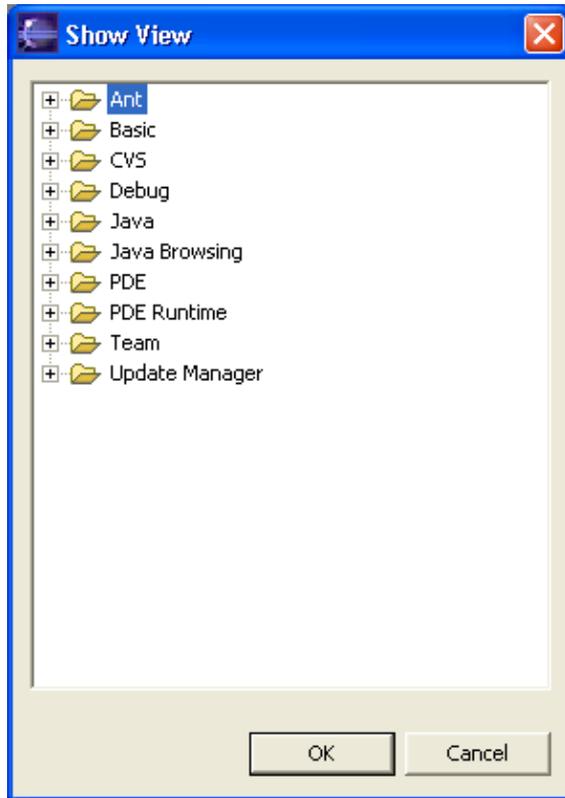
This command opens a new perspective in this Workbench window. This preference can be changed in the **Window > Preferences > Workbench > Perspectives** page. All of the perspectives that are open within the Workbench window are shown on the shortcut bar.

The perspectives you will likely want to open are listed first. This list is dependent on the current perspective. From the **Other...** submenu you can open any perspective.



## Show View

This command displays the selected view in the current perspective. You can configure how views are opened in the **Window > Preferences > Workbench > Perspectives** page. Views you are likely to want to open are listed first. This list is dependent on the current perspective. From the **Other...** submenu you can open any view. The views are sorted into categories in the Show View dialog.



## Show/Hide Editors

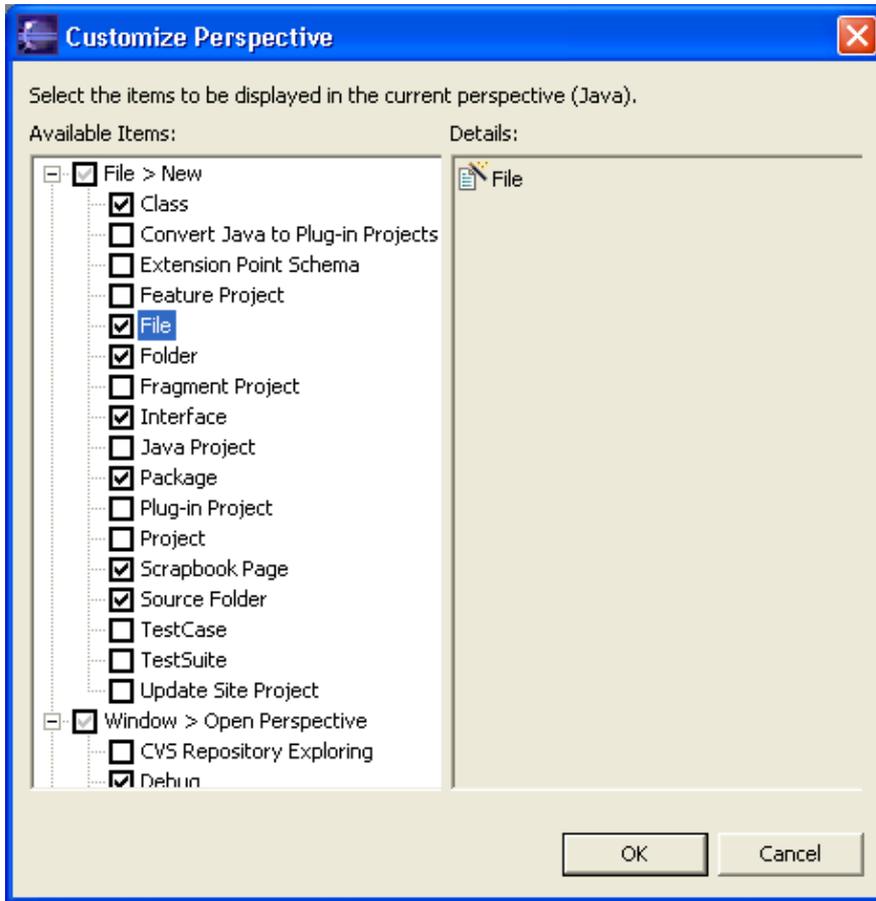
This command can be toggled to show or hide the editor area in the active perspective.

## Lock the Toolbars

This command toggles whether or not the Workbench tool bar can be manually reorganized.

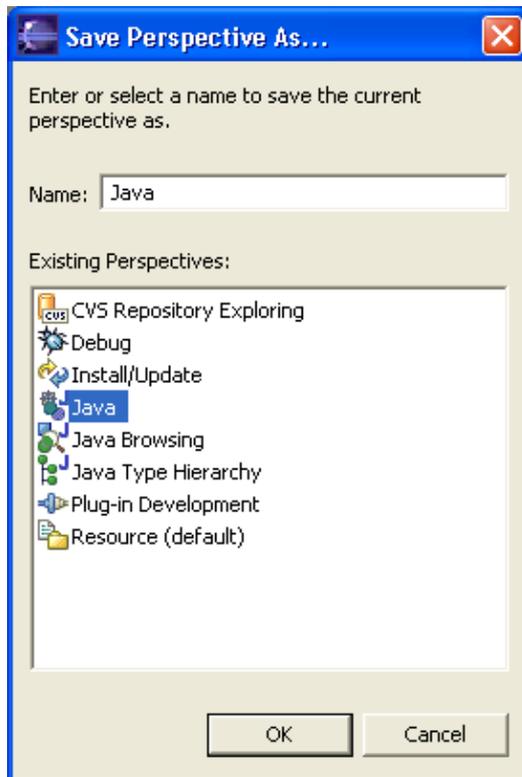
## Customize Perspective

Each perspective includes a predefined set of actions that are accessible from the menu bar and Workbench toolbar. Related actions are grouped into action sets. This command allows you to customize the current perspective by showing or hiding various action sets. The first three (**File > New**, **Window > Open Perspective**, **Window > Show View**) control which actions appear as top level items in their respective menus. The last category (**Other**) controls which action sets are visible in the perspective.



## Save Perspective As

This command allows you to save the current perspective, creating your own custom perspective. You can open more perspectives of this type using the **Window > Open Perspective > Other** menu item once you have saved a perspective.



## Reset Perspective

This command changes the layout of the current perspective to its original configuration.

## Close Perspective

This command closes the active perspective.

## Close All Perspectives

This command closes all open perspectives in the Workbench window.

## Keyboard Shortcuts

This submenu contains shortcut keys for navigating between the views, perspectives, and editors in the Workbench window.

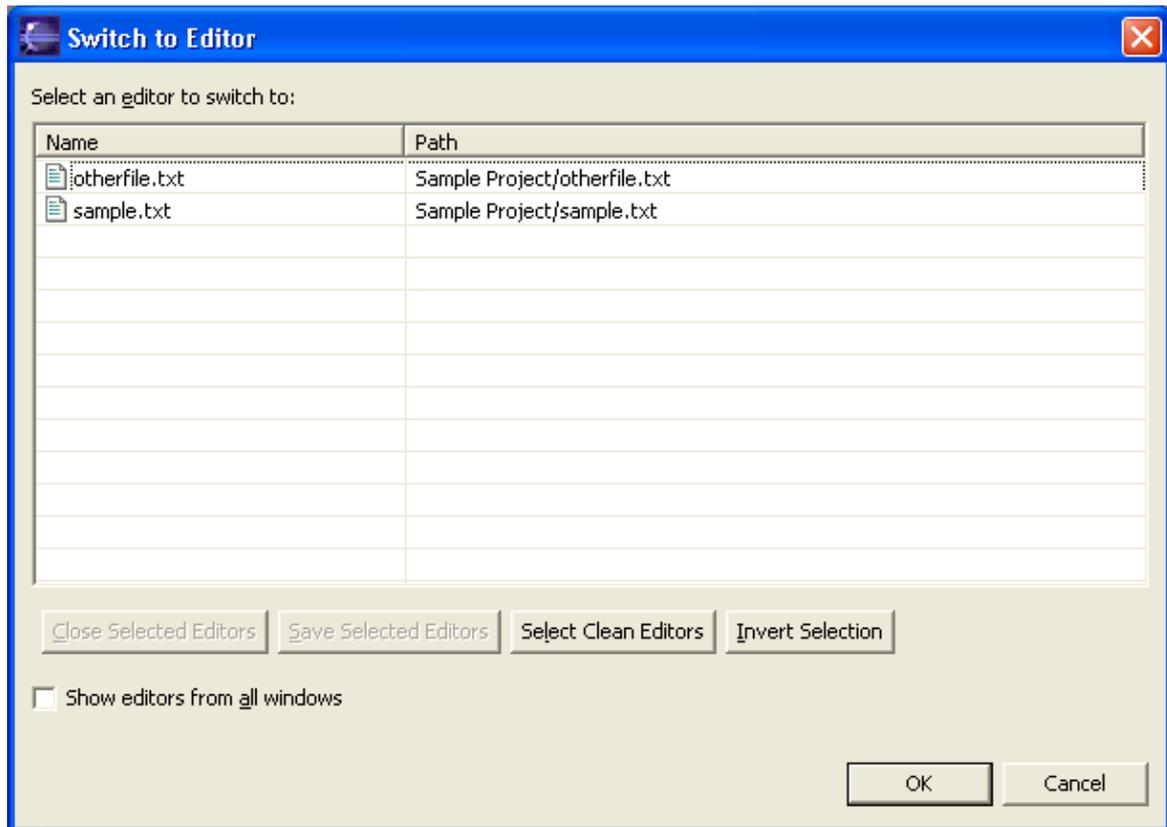
- **Show System Menu:** Shows the menu that is used for resizing, closing or pinning the current view or editor.
- **Show View Menu:** Shows the drop down menu that is available in the toolbar of the active view.
- **Activate Editor:** Makes the current editor active.
- **Next Editor:** Activates the next open editor in the list of most recently used editors.
- **Previous Editor:** Activates the previous open editor in the list of most recently used editors.
- **Next View:** Activates the next open view in the list of most recently used views.
- **Previous View:** Activates the previous open editor in the list of most recently used editors.
- **Next Perspective:** Activates the next open perspective in the list of most recently used perspectives.

## Crash recovery

- **Previous Perspective:** Activates the previous open perspective in the list of most recently used perspectives.

## Switch to Editor

Opens a dialog that allows you to manage all editors that are open, and rapidly switch to other open editors. Here is what the switch editors dialog looks like:



The following actions are available from the switch editors dialog:

- **Switch to Editor:** By selecting an editor from the list and hitting OK, that editor will be made active. If the chosen editor is in a different window or perspective, that window and perspective will be made active
- **Close Selected Editors:** Closes all editors selected in the list.
- **Save Selected Editors:** Saves the contents of all editors selected in the list.
- **Select Clean Editors:** Selects the editors in the list that have not been modified since their last save.
- **Invert Selection:** Reverses the selection in the list of editors.

## Preferences

This command allows you to indicate your preferences for using the Workbench. There are a wide variety of preferences for configuring the appearance of the Workbench and its views, and for customizing the behavior of all tools that are installed in the Workbench.

## Crash recovery

© Copyright IBM Corporation and others 2000, 2003.

# Help menu

This menu provides help on using the Workbench.

## Welcome

This command is used to open a Welcome editor containing a brief tutorial intended to introduce new users to some facet of Eclipse.

## Tips and Tricks

This command will open a list of interesting productivity features that you may not have discovered.

## Help Contents

This command displays the help view. The help view contains help books, topics, and information related to the Workbench and installed features.

## Software Updates

This group of commands allows you to update your product and to download and install new features.

## About

This command displays information about the product, installed features, and available plug-ins.

© Copyright IBM Corporation and others 2000, 2003.

# Navigator view icons

The following icons can appear in the Navigator view:

Navigator view icons

Icon	Description
	Project (open)
	Folder (open)
	Project (closed)
	Generic File

© Copyright IBM Corporation and others 2000, 2003.

# Editor area marker bar

The following markers can appear in the marker bar (to the left of the editor area):

## Marker Bar Markers

Icon	Description
	Bookmark
	Breakpoint
	Task marker
	Search result
	Error marker
	Warning marker
	Information marker

© Copyright IBM Corporation and others 2000, 2003.

# Tasks view

The following markers can appear in the Tasks view:

## Tasks View Icons

Icon	Description	Icon	Description
	Task (normal priority)		Information task
	High priority task		Warning task
	Low priority task		Error task
	Completed task		Problem task

© Copyright IBM Corporation and others 2000, 2003.

# Toolbar buttons

The following buttons may appear in the Workbench toolbar, toolbars for views, and the shortcut bar:

Button	Description	Button	Description
	Open a new perspective		Save the active editor contents
	Save the contents of all editors		Save editor contents under a new name or location
	Opens the search dialog		Print editor contents
	Open a resource creation wizard		Open a file creation wizard
	Open a folder creation wizard		Open a project creation wizard
	Open the import wizard		Open the export wizard
	Run incremental build		Run a program
	Debug a program		Run an external tool or Ant
	Cut selection to clipboard		Copy selection to clipboard
	Paste selection from clipboard		Undo most recent edit
	Redo most recent undone edit		Navigate to next item in a list
	Navigate to previous item in a list		Navigate forwards
	Navigate backwards		Navigate up one level
	Navigate back to home location		Open a view's drop down menu
	Close view or editor		Pin editor to prevent automatic reuse
	Pin view (make it no longer a fast view)		Add bookmark or task
	Filter tasks or properties		Go to a task, problem, or bookmark in the editor
	Restore default properties		Show items as a tree
	Refresh view contents		Sort list in alphabetical order
	Cancel a long running operation		Delete selected item or content

## ■ Related concepts

### Toolbars

© Copyright IBM Corporation and others 2000, 2003.

# External Tools and Ant icons

## Objects

	Ant buildfiles
	Ant target containing an error
	External tools project builder
	Invalid project builder
	Default target
	Public Ant target (target with description)
	Ant internal target (target with no description)
	Jar file
	Ant property
	Ant task
	Ant type

## Launch configurations

	Launch external tool
	Ant launch configuration
	Program launch configuration
	Main tab
	Refresh tab
	Targets tab
	Properties tab
	Classpath tab

## Ant View

	Ant view
	Add buildfile
	Add buildfiles with search
	Run selected buildfile or selected target
	Remove selected buildfiles
	Remove all buildfiles
	Properties

## Crash recovery

© Copyright IBM Corporation and others 2000, 2003.