

# **CodeWarrior™**

## **Development Tools**

### **mwClearCase Plug-in**

### **User Guide**

Revised 2002/10/29



Metrowerks, the Metrowerks logo, and CodeWarrior are registered trademarks of Metrowerks Corp. in the US and/or other countries. All other tradenames and trademarks are the property of their respective owners.

Copyright © Metrowerks Corporation. 2002. ALL RIGHTS RESERVED.

**The reproduction and use of this document and related materials are governed by a license agreement media, it may be printed for non-commercial personal use only, in accordance with the license agreement related to the product associated with the documentation. Consult that license agreement before use or reproduction of any portion of this document. If you do not have a copy of the license agreement, contact your Metrowerks representative or call 800-377-5416 (if outside the US call +1 512-997-4700). Subject to the foregoing non-commercial personal use, no portion of this documentation may be reproduced or transmitted in any form or by any means, electronic or mechanical, without prior written permission from Metrowerks.**

Metrowerks reserves the right to make changes to any product described or referred to in this document without further notice. Metrowerks makes no warranty, representation or guarantee regarding the merchantability or fitness of its products for any particular purpose, nor does Metrowerks assume any liability arising out of the application or use of any product described herein and specifically disclaims any and all liability. **Metrowerks software is not authorized for and has not been designed, tested, manufactured, or intended for use in developing applications where the failure, malfunction, or any inaccuracy of the application carries a risk of death, serious bodily injury, or damage to tangible property, including, but not limited to, use in factory control systems, medical devices or facilities, nuclear facilities, aircraft or automobile navigation or communication, emergency systems, or other applications with a similar degree of potential hazard.**

USE OF ALL SOFTWARE, DOCUMENTATION AND RELATED MATERIALS ARE SUBJECT TO THE METROWERKS END USER LICENSE AGREEMENT FOR SUCH PRODUCT.

## How to Contact Metrowerks

Corporate Headquarters	Metrowerks Corporation 9801 Metric Blvd. Austin, TX 78758 U.S.A.
World Wide Web	<a href="http://www.metrowerks.com">http://www.metrowerks.com</a>
Ordering & Technical Support	Voice: (800) 377-5416 Fax: (512) 997-4901

# Table of Contents

---

<b>1</b>	<b>Overview</b>	<b>5</b>
	About the mwClearCase Plug-in . . . . .	5
	System Requirements . . . . .	5
	Learning More About mwClearCase . . . . .	6
<b>2</b>	<b>Setting Up the mwClearCase Plug-in</b>	<b>7</b>
	Obtaining the mwClearCase Plug-in . . . . .	7
	Installing the mwClearCase Plug-in . . . . .	8
	Activating the mwClearCase Plug-in . . . . .	10
	Deactivating the mwClearCase Plug-in . . . . .	17
<b>3</b>	<b>Using the mwClearCase Plug-in</b>	<b>19</b>
	Preparation . . . . .	19
	The Project Window . . . . .	20
	The Project Window and the Plug-in . . . . .	20
	mwClearCase Plug-in Command Locations . . . . .	21
	The IDE's VCS Menu . . . . .	21
	The Project Window's Context Menu . . . . .	22
	The Editor Window's VCS Menu . . . . .	23
	mwClearCase Plug-in Command Reference . . . . .	24
<b>4</b>	<b>Tutorial</b>	<b>57</b>
	Creating a Default Plug-in Configuration . . . . .	57
	Checking out a File . . . . .	60
	Checking out a Project . . . . .	61
	Checking in a File . . . . .	61
	Checking in a Project . . . . .	63
	<b>Index</b>	<b>65</b>

**Table of Contents**

---

# Overview

---

This chapter provides an overview of the Metrowerks ClearCase plug-in for the CodeWarrior™ integrated development environment (IDE). This tool is commonly called the mwClearCase plug-in.

This chapter contains these topics:

- [About the mwClearCase Plug-in](#)
- [System Requirements](#)
- [Learning More About mwClearCase](#)

## About the mwClearCase Plug-in

The mwClearCase plug-in is a tool that makes working with CodeWarrior projects whose files are managed by the ClearCase software configuration management system more convenient. How?

The mwClearCase plug-in lets you:

- Execute common ClearCase commands on CodeWarrior files and projects from within the CodeWarrior IDE.
- View the ClearCase status of each file in a project (and of the project file itself) from within the IDE.

In short, the plug-in is a time saver because it lets you use ClearCase without leaving the CodeWarrior IDE. Without the plug-in, you must run a separate program to use ClearCase.

## System Requirements

To use the mwClearCase plug-in, your computer, operating system, and software must meet the requirements listed in [Table 1.1](#).

## Overview

*Learning More About mwClearCase*

---

**Table 1.1 System Requirements for the mwClearCase Plug-in**

Component	Requirement
Computer	IBM-compatible PC
Processor	Pentium-class processor
Memory	128 MB, minimum
Operating system	Windows 98, Windows ME, Windows NT, Windows 2000, or Windows XP
ClearCase	Rational ClearCase for Windows version 3.2 or later
CodeWarrior IDE	Version 4.0.1 or later (Windows-hosted version only)

## Learning More About mwClearCase

The *mwClearCase Plug-in User Guide* (this document) explains how to use the features of the plug-in. It does not cover project configuration management concepts or document the ClearCase product. For information of this type, refer to the ClearCase documentation.

# Setting Up the mwClearCase Plug-in

---

This chapter explains how to obtain, install, activate, configure, and deactivate the mwClearCase plug-in.

The chapter contains these topics:

- [Obtaining the mwClearCase Plug-in](#)
- [Installing the mwClearCase Plug-in](#)
- [Activating the mwClearCase Plug-in](#)
- [Deactivating the mwClearCase Plug-in](#)

## Obtaining the mwClearCase Plug-in

The Metrowerks ClearCase plug-in is free. To obtain the latest version of the plug-in, follow these steps:

- 1 If the CodeWarrior IDE is running, exit the program.
- 2 Go to the Metrowerks version control tool download page by clicking the URL below (or by typing it into your Internet browser):

<http://www.metrowerks.com/MW/Develop/Desktop/VersionControl.htm>

Your Internet browser starts and displays the Metrowerks version control tool download page.

- 3 In the Windows section of the download page, click the name of the .zip file that contains the mwClearCase plug-in.
- 4 Use the file download capabilities of your browser to obtain a local copy of the mwClearCase plug-in's .zip file.

- 5 Quit the Internet browser.

You now have a local copy of the .zip file that contains the latest version of the mwClearCase plug-in.

## Installing the mwClearCase Plug-in

This section explains how to install the mwClearCase plug-in.

---

**NOTE** If your CodeWarrior product includes the mwClearCase plug-in, that product's installation program automatically installs the plug-in.

Use the procedure in this section to install the mwClearCase plug-in once you have downloaded it from the Metrowerks website.

---

To install the mwClearCase plug-in, follow these steps:

- 1 Using Windows Explorer, create a temporary directory on your local hard disk. You can name this directory anything you want.
- 2 Using Windows Explorer, select the directory that contains the mwClearCase plug-in's .zip file.

For instructions that explain how to get this file, see [“Obtaining the mwClearCase Plug-in” on page 7](#).

- 3 Double-click the name of the plug-in's .zip file.  
WinZip (or another archive decompression program) runs and displays the contents of the .zip file.
- 4 Extract the contents of the .zip file to the temporary directory created earlier.

The decompression program creates these files in the temporary directory:

- mwClearCase\_UG.chm
- mwClearCase\_UG.pdf
- mwClearCase\_Release\_Notes\_win32i.txt
- VCSClearCase.dll



**NOTE** At the time of this writing, the installation steps that follow were correct. That said, before executing these steps, please check the release notes for changes to the installation procedure. The name of the release notes file is `mwClearCase_Release_Notes_win32i.txt`

---

**5** Copy the file `VCSClearCase.dll`

from the directory:

`{YourTempDir}`

to the directory:

`{CodeWarriorInstallDir}\Bin\Plugins\version control\`

where `{YourTempDir}` is a placeholder for the directory to which you extracted the mwClearCase plug-in .zip file and `{CodeWarriorInstallDir}` is a placeholder for the directory in which you installed your CodeWarrior product.

**6** Copy the file `mwClearCase_Release_Notes_win32i.txt`

from the directory:

`{YourTempDir}`

to the directory:

`{CodeWarriorInstallDir}\Release Notes  
\Version Control Notes\`

**7** Copy the file `ClearCase_UG.chm`

from the directory:

`{YourTempDir}\Documentation\`

to the directory:

`{CodeWarriorInstallDir}\CodeWarrior Manuals\HTML`

**8** Copy the file `ClearCase_UG.pdf`

from the directory:

`{YourTempDir}`

to the directory:

`{CodeWarriorInstallDir}\CodeWarrior Manuals\PDF`

**9** Delete the temporary directory.

The mwClearCase plug-in is now installed.

---

**NOTE** Read the release notes before using the mwClearCase plug-in. The name of the release notes file is `mwClearCase_Release_Notes_win32i.txt`

---

## Activating the mwClearCase Plug-in

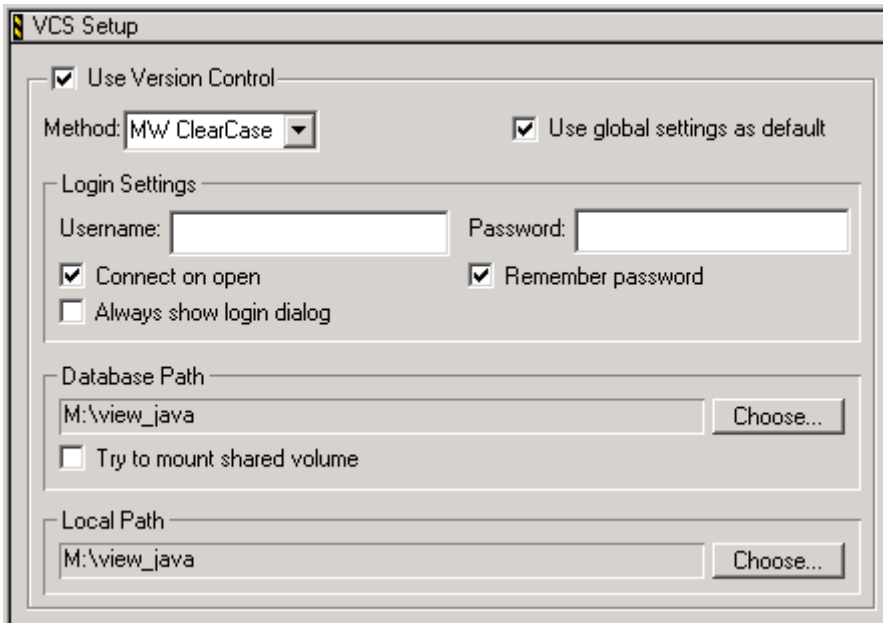
Before you can use the mwClearCase plug-in, you must first activate and configure it. To do this, you must make entries in the VCS Setup settings panel.

[Figure 2.1](#) shows the VCS Setup settings panel. [Table 2.1](#) lists each component in this panel and explains the purpose of each.

These procedures explain how to activate and configure the mwClearCase plug-in:

- [Creating a Default mwClearCase Plug-in Configuration](#)
- [Creating a Project-specific mwClearCase Plug-in Configuration](#)

**Figure 2.1** The VCS Setup Settings Panel of the VCS Settings Window



**Table 2.1 VCS Setup Settings Panel: Components**

Panel Component	Description
<b>Use Version Control</b> check box	Check to instruct the IDE to use a version control system plug-in. Clear to stop the IDE from using a version control system plug-in. <b>NOTE:</b> If you are overriding the default plug-in configuration, this item's label is <b>Use custom project</b> , not <b>Use Version Control</b> .
<b>Method</b> dropdown menu	Choose the version control system to activate. To activate the mwClearCase plug-in, choose MW ClearCase.
<b>Use global settings as default</b> check box	Check to create the default mwClearCase plug-in configuration. Clear to deactivate the default plug-in configuration. <b>NOTE:</b> This check box is present only if there were no projects open when you displayed the VCS Settings window.
<b>Login Settings</b> group box	Groups items that pertain to logging into a ClearCase view. The bulleted items below explain each item in this group. <b>NOTE:</b> For the plug-in to connect to ClearCase successfully, you must enter the login settings exactly as specified below.
• <b>Username</b> text field	Leave the Username field <i>blank</i> .
• <b>Password</b> text field	Leave the Password field <i>blank</i> .
• <b>Remember password</b> check box	<i>Always</i> check the <b>Remember password</b> check box.
• <b>Connect on open</b> check box	<i>Always</i> check the <b>Connect on open</b> check box.
• <b>Always show login dialog</b> check box	<i>Never</i> check the <b>Always show login dialog</b> check box.
<b>Database Path</b> group box	Groups items that pertain to the location of the ClearCase view that the plug-in uses. The bulleted items below explain each of these items.
• <b>Choose...</b> button	Displays the <b>Select an Access Path</b> dialog box. Use this dialog box to select the directory that contains the ClearCase view for the plug-in to use. You can specify an absolute, compiler relative, or system relative path. If a project is open, you can also specify a project relative path. <b>NOTE:</b> You must specify the same ClearCase view for <b>Database Path</b> and for <b>Local Path</b> (explained below).
• <b>Path</b> text field	Displays the name of the selected ClearCase view. This field is read-only.
• <b>Try to mount shared volume</b> check box	Reserved for future use. Leave unchecked.

**Table 2.1 VCS Setup Settings Panel: Components**

Panel Component	Description
<b>Local Path</b> group box	Groups items that pertain to the location of the ClearCase view that the plug-in uses. The bulleted items below explain each of these items. <b>NOTE:</b> You must specify the same ClearCase view for <b>Local Path</b> and for <b>Database Path</b> (explained above).
• <b>Choose...</b> button	Displays the <b>Select an Access Path</b> dialog box. Use this dialog box to select the same ClearCase view selected for Database Path (explained above). You can specify an absolute, compiler relative, or system relative path. Also, if a project is open, you can specify a project relative path.
• <b>Path</b> text field	Displays the selected local path. This field is read-only.

There are two kinds of plug-in activation:

- Global activation

To activate the mwClearCase plug-in globally, create a default plug-in configuration. In this case, the IDE activates the mwClearCase plug-in for each CodeWarrior project that has not been assigned a project-specific plug-in configuration.

- Project-specific activation

To activate the mwClearCase plug-in for a specific project, assign a plug-in configuration to just this CodeWarrior project. In this case, the IDE activates the mwClearCase plug-in only when this project is both open and selected.

The procedures in this section explain how to activate the plug-in each of these ways. These procedures are:

- [Creating a Default mwClearCase Plug-in Configuration](#)
- [Creating a Project-specific mwClearCase Plug-in Configuration](#)

---

## Creating a Default mwClearCase Plug-in Configuration

The CodeWarrior IDE attempts to use the default mwClearCase plug-in configuration for each project that does not have a custom configuration.

---

**NOTE** You do not have to create the default mwClearCase plug-in configuration. Instead, you can create a custom plug-in configuration for each of your projects. The default plug-in configuration is just a convenience.

---

To create the default mwClearCase plug-in configuration, follow these steps:

1 Run the CodeWarrior IDE.

2 Close any open projects.

3 Choose **Edit > Version Control Settings...**

The VCS Settings window appears.

4 Display the VCS Setup settings panel in the VCS Settings window.

To do this, select the item labeled **VCS Setup**. This item is in the pane labeled **VCS Settings Panels**. This pane is on the left side of the VCS Settings window.

[Figure 2.1 on page 10](#) shows the VCS Setup panel.

5 Check the **Use Version Control** check box.

The IDE enables the **Method** dropdown menu and the **Use global settings as default** check box.

6 Select the item labeled `MW ClearCase` from the **Method** dropdown menu.

The IDE enables the rest of the items in the VCS Setup panel.

7 Check **Use global settings as default** check box.

This setting makes the current plug-in configuration the default configuration. The IDE attempts to apply this configuration to all CodeWarrior projects that do not have a custom plug-in configuration.

---

**NOTE** Because no CodeWarrior project is open, the VCS Settings panel includes the **Use global settings as default** check box. If a project is open, the VCS Setting panel does not include this check box.

---

**8** Specify the path to the ClearCase view to make the default view.

To do this, follow these steps:

- a. Click the **Choose...** button in the **Database Path** group box.  
The **Select an Access Path** dialog box appears.
- b. Use this dialog box to navigate to the directory that contains the ClearCase view that you want to make the default view.
- c. Click **OK**  
The selected path appears in the read-only field to the left of the **Choose...** button.

---

**NOTE** If you are using a *dynamic* view, you must start the view before its directory appears in the **Select an Access Path** dialog box. See the ClearCase documentation for instructions that explain how to start a dynamic view

If you are using a snapshot view, its directory appears in the **Select an Access Path** dialog box automatically.

---

**9** Specify the local path to make the default local path.

To do this, follow these steps:

- a. Click the **Choose...** button in the **Local Path** group box.  
The **Select an Access Path** dialog box displays.
- b. Use this dialog box to select the same ClearCase view chosen in step 9.
- c. Click **OK**  
The selected path appears in the read-only field to the left of the **Choose...** button.

---

**NOTE** You must select the same view for **Local Path** and **Database Path** (discussed above).

---

**10** Click **OK**

The VCS Settings window closes. The IDE activates the mwClearCase plug-in globally and adds the VCS menu to its menu bar.

Refer to [“Using the mwClearCase Plug-in” on page 19](#) for instructions that explain how to use each feature of the plug-in.

---

## Creating a Project-specific mwClearCase Plug-in Configuration

A project-specific plug-in configuration is one that applies to just the CodeWarrior project that is open and selected when you create the configuration.

To create a project-specific plug-in configuration, follow these steps:

- 1 Run the CodeWarrior IDE.
- 2 Close any open projects.
- 3 Open the CodeWarrior project to which you want to assign a custom plug-in configuration.
- 4 Choose **Edit > Version Control Settings...**

The VCS Settings window appears.

- 5 Display the VCS Setup settings panel in the VCS Settings window.

To do this, select the item labeled **VCS Setup**. This item is in the pane labeled **VCS Settings Panels**. This pane is on the left side of the VCS Settings window.

---

**NOTE** Because a project is open, the VCS Settings panel does *not* include the **Use global settings as default** check box. This is because the IDE must assign a plug-in configuration to an open project if there is one.

---

- 6 Check the **Use Version Control** check box.

---

**NOTE** If the default plug-in configuration has been defined, this check box is labeled **Use custom project** instead of **Use Version Control**. In this case, the new plug-in configuration overrides the default plug-in configuration.

---

- 7 Select the `MW ClearCase` item from the **Method** dropdown menu.

The IDE enables the rest of the items in the VCS Setup panel.

- 8** Specify the path to the ClearCase view that contains the open project and its source files.

To do this, follow these steps:

- a.** Click the **Choose...** button in the **Database Path** group box.  
The **Select an Access Path** dialog box displays.
- b.** Use this dialog box to navigate to the directory that holds the ClearCase view that contains the open project.
- c.** Click **OK**

The selected path appears in the read-only text field to the left of the **Choose...** button.

---

**NOTE** If you are using a *dynamic* view, you must start the view before its directory appears in the **Select an Access Path** dialog box. See the ClearCase documentation for instructions that explain how to start a dynamic view

If you are using a snapshot view, its directory appears in the **Select an Access Path** dialog box automatically.

---

- 9** Specify the local path to make the default local path.

To do this, follow these steps:

- a.** Click the **Choose...** button in the **Local Path** group box.  
The **Select an Access Path** dialog box displays.
- b.** Use this dialog box to select the same ClearCase view chosen in step 9.
- c.** Click **OK**

The selected path appears in the read-only field to the left of the **Choose...** button.

- d.** Click **OK**

The selected path appears in the read-only text field to the left of the **Choose...** button.

- 10** Click **OK**

The VCS Settings window closes. The IDE activates the mwClearCase plug-in for the open project.



---

**NOTE** The VCS menu appears in the IDE's menu bar when you select a project for which the plug-in has been activated. This menu disappears when you select a project for which the plug-in has not been activated.

---

Refer to [“Using the mwClearCase Plug-in” on page 19](#) for instructions that explain how to use each feature of the mwClearCase plug-in.

## Deactivating the mwClearCase Plug-in

To deactivate the mwClearCase plug-in, follow these steps:

- 1 Run the CodeWarrior IDE.
- 2 If you are deactivating the default plug-in configuration, close all projects.  
If you are deactivating the plug-in configuration for a specific CodeWarrior project, open that project.
- 3 Choose **Edit > Version Control Settings...**  
The VCS Settings window appears.
- 4 Display the VCS Setup settings panel in the VCS Settings window.  
To do this, select the item labeled **VCS Setup**. This item is in the pane labeled **VCS Settings Panels**. This pane is on the left side of the VCS Settings window.  
[Figure 2.1 on page 10](#) shows the VCS Settings window with the VCS Setup panel displayed.
- 5 Select the item labeled **None** from the **Method** dropdown menu.
- 6 If you are deactivating the default plug-in configuration, clear the box labeled **Use global settings as default**.

---

**NOTE** The VCS Setup panel includes the **Use global settings as default** check box only if the IDE has no projects open. The VCS Setup panel does not include this check box if the IDE has one or more projects open.

---

- 7 Clear the **Use Version Control** check box.  
The IDE disables the rest of the items in the VCS Setup panel.

---

**NOTE** If the default plug-in configuration exists, this check box is labeled **Use custom project** instead of **Use Version Control**.

---

**8 Click OK**

The VCS Settings window closes. The IDE deactivates the plug-in and removes the VCS menu from its menu bar.

---

**NOTE** If no projects were open when you clicked **OK**, the IDE deactivates the default plug-in configuration.

If a project was open, the IDE deactivates the plug-in for just this project.

---

# Using the mwClearCase Plug-in

---

This chapter explains how to use the mwClearCase plug-in.

The chapter contains these topics:

- [Preparation](#)
- [The Project Window](#)
- [mwClearCase Plug-in Command Locations](#)
- [mwClearCase Plug-in Command Reference](#)

## Preparation

Before you can use the mwClearCase plug-in with a CodeWarrior™ project, you must first create a ClearCase view that contains your project. Further, if you are using a dynamic ClearCase view, you must first start this view you can use the plug-in. Refer to the ClearCase documentation for instructions that explain how to accomplish these tasks.

Once you have set up your ClearCase view, open a CodeWarrior project in the CodeWarrior IDE. If you created the default plug-in configuration or assigned a custom configuration to the project, you can now use the mwClearCase plug-in with this project.

See [“Creating a Default mwClearCase Plug-in Configuration” on page 12](#) and [“Creating a Project-specific mwClearCase Plug-in Configuration” on page 15](#) for instructions that explain how to configure the mwClearCase plug-in.

# The Project Window

To use the mwClearCase plug-in, you must understand how the plug-in interacts with the project window of a CodeWarrior project. This is because the project window:

- Shows you the ClearCase status of each file in the project and of the project itself.
- Lets you apply ClearCase commands to files in the project and to the project itself.
- Shows you the results of ClearCase commands.

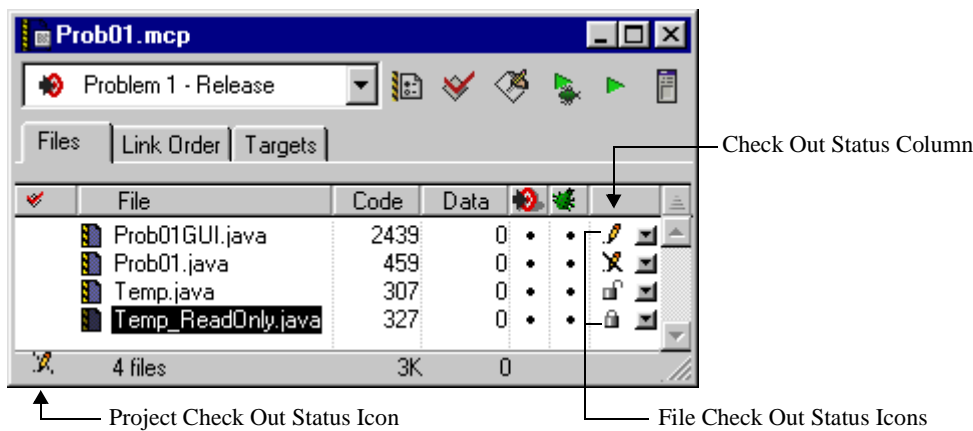
## The Project Window and the Plug-in

Once you have activated the mwClearCase plug-in, a new column appears in the project window of the project for which the plug-in was activated (or in all project windows if you activated the plug-in globally). This column contains icons that signify the check out status of each file in the project. In addition, an icon in the lower-left corner of the project window indicates the check out status of the project file itself (the .mcp file).






[Figure 3.1](#) shows the project window of a project for which the mwClearCase plug-in has been activated.

[Figure 3.2](#) explains how to interpret each check out status icon.

**Figure 3.1 A Project Window with the mwClearCase Plug-in Active**



**Figure 3.2 Check Out Status Icons: Interpretation**

Icon	Check Out Status
	File has not been added to ClearCase. File is writable.
	File has not been added to ClearCase. File is read-only.
	File has not been checked out of ClearCase. File is read-only.
	File has not been checked out of ClearCase. File is writable.
	File has been checked out of ClearCase. File is writable.

## mwClearCase Plug-in Command Locations

The mwClearCase plug-in lets you apply common ClearCase commands to a CodeWarrior project and to the files that belong to a project. To use the plug-in, however, you must know where to find these commands.

The mwClearCase plug-in makes ClearCase commands available in these places:

- [The IDE's VCS Menu](#)
- [The Project Window's Context Menu](#)
- [The Editor Window's VCS Menu](#)

### The IDE's VCS Menu

The VCS menu is in the CodeWarrior IDE's menu bar.

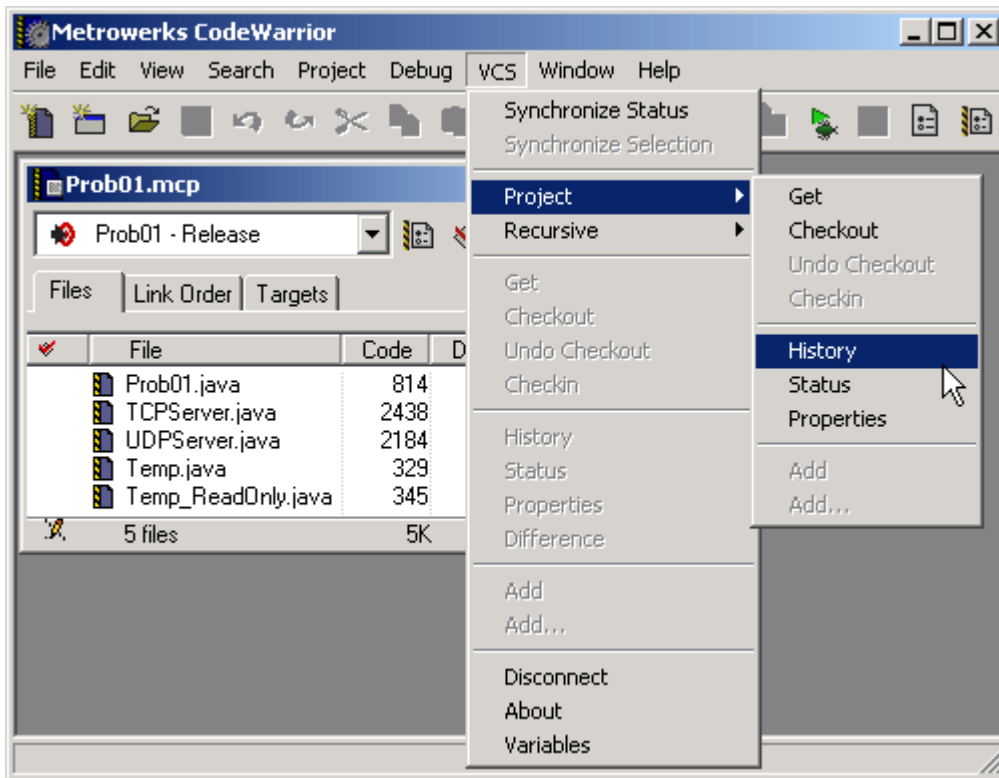
The VCS menu is present in the menu bar if at least one of these conditions is met:

- You created the default plug-in configuration.
- You assigned a plug-in configuration to a project, and this project is both open and selected.

The VCS menu includes every mwClearCase command supported by the plug-in. Other menus (discussed below) include just a subset of the supported ClearCase commands.

Individual items in the VCS menu are enabled or disabled depending upon the check out status of the current project or currently selected project file.

**Figure 3.3 The VCS Menu**



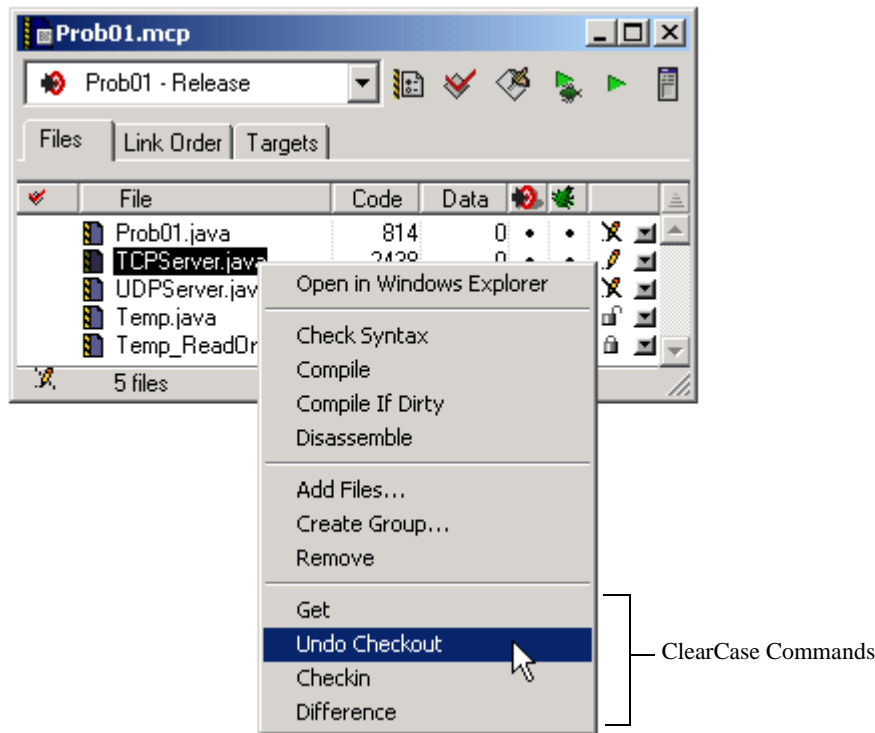
## The Project Window's Context Menu

To display a project window's context menu, right-click the mouse while its cursor is within the project window. [Figure 3.4](#) shows a project window context menu.

If the project window is for a project for which you have activated the mwClearCase plug-in, the project window context menu includes ClearCase commands at the bottom. Unlike the VCS menu, the context menu contains just a subset of the ClearCase commands the plug-in supports. Further, the particular commands included vary depending upon the check out status of the project file that is currently selected.

For example, because the file selected in [Figure 3.4](#) is checked out, the context menu contains just the [Get](#), [Undo Checkout](#), [Checkin](#), and [Difference](#) commands. This is because only these commands are pertinent for a file that is checked out.

Figure 3.4 The Project Window's Context Menu



## The Editor Window's VCS Menu

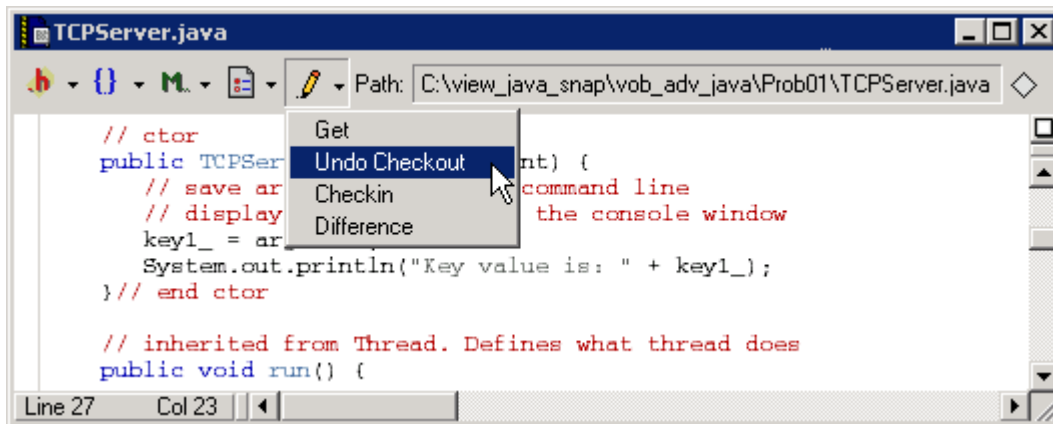
If the mwClearCase plug-in is active for a project, the editor window toolbar includes a VCS menu for each project file you open.

An editor window's VCS menu is identified by one of the checkout status icons. The particular icon displayed corresponds to the check out status of the file open in the editor window. [Figure 3.2 on page 21](#) explains how to interpret each check out status icon. [Figure 3.5](#) shows a CodeWarrior editor window.

If you click the check out status icon, a dropdown menu displays. This menu contains ClearCase commands. Unlike the IDE's VCS menu, the editor's VCS menu contains just those ClearCase commands that are pertinent based on the check out status of the open file.

For example, because the file in [Figure 3.5](#) is checked out, the dropdown menu contains just the [Get](#), [Undo Checkout](#), [Connect](#), and [Difference](#) commands. This is because these are the only commands you can apply to a file that is checked out.

Figure 3.5 The Editor Window's Menu of ClearCase Commands



## mwClearCase Plug-in Command Reference

This section explains the purpose and effect of each command supported by the mwClearCase plug-in.

The plug-in supports four categories of ClearCase commands:

- File Commands

File commands are ClearCase commands that the plug-in applies to the selected file(s) in the selected project.

You can issue file commands from the IDE's VCS menu, from a project's context menu, and from an editor window's VCS menu.

- Project Commands

File commands are ClearCase commands that the plug-in applies to the selected project.

You can issue project commands from just the Project submenu of the IDE's VCS menu.

- Recursive Commands

Recursive commands are ClearCase commands that the plug-in applies to each subdirectory and file contained in the plug-in's view directory.

You can issue recursive commands from just the Recursive submenu of the IDE's VCS menu.



- Miscellaneous Commands

Miscellaneous commands are *not* ClearCase commands. The plug-in's miscellaneous commands perform a variety of actions. See the reference entry for each miscellaneous command for an explanation of the purpose and effect of each of the commands in this category.

[Table 3.1](#) lists the commands the plug-in supports along with the location and type of each command.

**Table 3.1 Commands Supported by the mwClearCase Plug-in**

Command	Command Location(s)	Command Type(s)
<a href="#">About</a>	<ul style="list-style-type: none"> <li>• VCS Menu</li> </ul>	<ul style="list-style-type: none"> <li>• Miscellaneous</li> </ul>
<a href="#">Add</a>	<ul style="list-style-type: none"> <li>• VCS Menu</li> <li>• Project Window Context Menu</li> <li>• Editor Window Menu</li> </ul>	<ul style="list-style-type: none"> <li>• File</li> <li>• Project</li> <li>• Recursive</li> </ul>
<a href="#">Add...</a>	<ul style="list-style-type: none"> <li>• VCS Menu</li> <li>• Editor Window Menu</li> </ul>	<ul style="list-style-type: none"> <li>• File</li> <li>• Project</li> <li>• Recursive</li> </ul>
<a href="#">Checkin</a>	<ul style="list-style-type: none"> <li>• VCS Menu</li> <li>• Project Window Context Menu</li> <li>• Editor Window Menu</li> </ul>	<ul style="list-style-type: none"> <li>• File</li> <li>• Project</li> <li>• Recursive</li> </ul>
<a href="#">Checkout</a>	<ul style="list-style-type: none"> <li>• VCS Menu</li> <li>• Project Window Context Menu</li> <li>• Editor Window Menu</li> </ul>	<ul style="list-style-type: none"> <li>• File</li> <li>• Project</li> <li>• Recursive</li> </ul>
<a href="#">Connect</a>	<ul style="list-style-type: none"> <li>• VCS Menu</li> </ul>	<ul style="list-style-type: none"> <li>• Miscellaneous</li> </ul>
<a href="#">Difference</a>	<ul style="list-style-type: none"> <li>• VCS Menu</li> <li>• Project Window Context Menu</li> <li>• Editor Window Menu</li> </ul>	<ul style="list-style-type: none"> <li>• File</li> </ul>
<a href="#">Disconnect</a>	<ul style="list-style-type: none"> <li>• VCS Menu</li> </ul>	<ul style="list-style-type: none"> <li>• Miscellaneous</li> </ul>
<a href="#">Get</a>	<ul style="list-style-type: none"> <li>• VCS Menu</li> <li>• Project Window Context Menu</li> <li>• Editor Window Menu</li> </ul>	<ul style="list-style-type: none"> <li>• File</li> <li>• Project</li> <li>• Recursive</li> </ul>
<a href="#">History</a>	<ul style="list-style-type: none"> <li>• VCS Menu</li> </ul>	<ul style="list-style-type: none"> <li>• File</li> <li>• Project</li> </ul>
<a href="#">Properties</a>	<ul style="list-style-type: none"> <li>• VCS Menu</li> </ul>	<ul style="list-style-type: none"> <li>• File</li> <li>• Project</li> </ul>
<a href="#">Subtree Add</a>	<ul style="list-style-type: none"> <li>• VCS Menu</li> </ul>	<ul style="list-style-type: none"> <li>• Recursive</li> </ul>
<a href="#">Subtree Add...</a>	<ul style="list-style-type: none"> <li>• VCS Menu</li> </ul>	<ul style="list-style-type: none"> <li>• Recursive</li> </ul>

**Table 3.1** Commands Supported by the mwClearCase Plug-in

Command	Command Location(s)	Command Type(s)
<a href="#">Subtree Checkin</a>	<ul style="list-style-type: none"><li>• VCS Menu</li></ul>	<ul style="list-style-type: none"><li>• Recursive</li></ul>
<a href="#">Subtree Checkout</a>	<ul style="list-style-type: none"><li>• VCS Menu</li></ul>	<ul style="list-style-type: none"><li>• Recursive</li></ul>
<a href="#">Subtree Get</a>	<ul style="list-style-type: none"><li>• VCS Menu</li></ul>	<ul style="list-style-type: none"><li>• Recursive</li></ul>
<a href="#">Status</a>	<ul style="list-style-type: none"><li>• VCS Menu</li></ul>	<ul style="list-style-type: none"><li>• File</li><li>• Project</li></ul>
<a href="#">Synchronize Selection</a>	<ul style="list-style-type: none"><li>• VCS Menu</li></ul>	<ul style="list-style-type: none"><li>• Miscellaneous</li></ul>
<a href="#">Synchronize Status</a>	<ul style="list-style-type: none"><li>• VCS Menu</li></ul>	<ul style="list-style-type: none"><li>• Miscellaneous</li></ul>
<a href="#">Undo Checkout</a>	<ul style="list-style-type: none"><li>• VCS Menu</li><li>• Project Window Context Menu</li><li>• Editor Window Menu</li></ul>	<ul style="list-style-type: none"><li>• File</li><li>• Project</li></ul>
<a href="#">Variables</a>	<ul style="list-style-type: none"><li>• VCS Menu</li></ul>	<ul style="list-style-type: none"><li>• Miscellaneous</li></ul>

## About

The `About` command displays the mwClearCase plug-in’s “About” box.

The About box displays the version number of the plug-in along with other identification information.

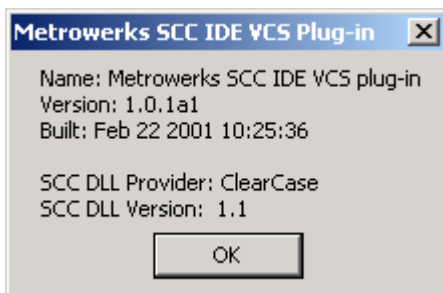
To issue the `About` command, select **VCS > About**. An About box like that shown in [Figure 3.6](#) displays.

---

**NOTE** The `About` command is available in just the IDE’s VCS menu.

---

**Figure 3.6** About Box for the mwClearCase Plug-in



## Add

You can apply the Add command to a file, to a project, or recursively.

If applied to a file, the Add command adds the selected file to ClearCase.

If applied to an open project, the Add command adds the project file (that is, the .mcp file) to ClearCase.

If applied recursively, the Add command adds all files and projects contained in the plug-in's ClearCase view to ClearCase.

### Adding a File

To add a file to ClearCase, follow these steps:

- 1 Run the CodeWarrior IDE.
- 2 Open the project that contains the file you want to add to ClearCase.  
The CodeWarrior IDE displays the project window for this project.
- 3 In the project window, click the name of the file you want to add.  
The IDE highlights this file name.

---

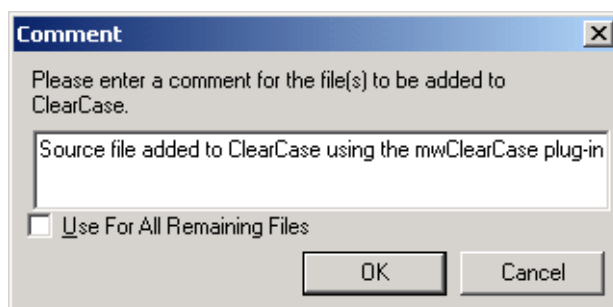
**TIP** To add several files at once, highlight the name of each file you want to add and then issue the Add command.

---

- 4 Select **VCS > Add**


The **Comment** dialog box appears. [Figure 3.7](#) shows this dialog box.

**Figure 3.7 The Comment Dialog Box**



5 Optionally, type a comment in the **Comment** dialog box.

6 Click **OK**

The **Comment** dialog box closes. The plug-in adds the selected file and comment (if any) to ClearCase and changes the file's check out status icon to: 

## Adding a Project

To add a project to ClearCase, follow these steps:

1 Run the CodeWarrior IDE.


2 Open the CodeWarrior project whose project file you want to add to ClearCase. The CodeWarrior IDE displays the project window for this project.

3 Select **VCS > Project > Add**

The **Comment** dialog box appears. [Figure 3.7 on page 27](#) shows this dialog box.

4 Optionally, type a comment in the **Comment** dialog box.

5 Click **OK**

The **Comment** dialog box closes. The plug-in adds the project file and comment (if any) to ClearCase and changes the project's check out status icon to: . This icon is in the lower-left corner of the project window.

## Adding Files and Projects Recursively

To add files and projects to ClearCase recursively, follow these steps:

1 Run the CodeWarrior IDE.

2 Select **VCS > Recursive > Add**

---

**NOTE** You do not have to open a project before issuing the recursive Add command because this form of the command does not affect a specific project. Instead, the command affects all subdirectories and files in the plug-in's view.

---

3 The **Comment** dialog box appears. [Figure 3.7 on page 27](#) shows this dialog box.

4 Optionally, enter a comment in the **Comment** dialog box.

5 Click **OK**

The **Comment** dialog box closes.

The plug-in adds all files and projects in its ClearCase view to ClearCase.

---

## Add...

You can apply the Add... command to a file, to a project, or recursively.

If applied to a file, the Add... command adds the selected file to ClearCase.

If applied to an open project, the Add... command adds the project file (that is, the project's .mcp file) to ClearCase.

If applied recursively, the Add... command adds all files and projects contained in the plug-in's ClearCase view to ClearCase.

---

**NOTE** The Add... command is different from the Add command. The Add... command gives you more control over how a file is added to ClearCase. See [“Add” on page 27](#) for an explanation of the Add command.

---

### Adding a File using the Add... Command

To add a file to ClearCase using the Add... command, follow these steps:

- 1 Run the CodeWarrior IDE.
- 2 Open the project that contains the file you want to add to ClearCase.  
The CodeWarrior IDE displays the project window for this project.
- 3 In the project window, click the name of the file you want to add.  
The IDE highlights this file name.

---

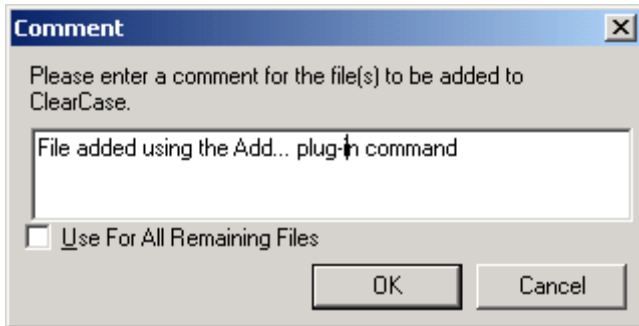
**TIP** To add several files at once, highlight the name of each file you want to add and then issue the Add... command.

---

4 Select **VCS > Add...**

The **Comment** dialog box appears. [Figure 3.8](#) shows this dialog box.

**Figure 3.8 The Comment Dialog Box**

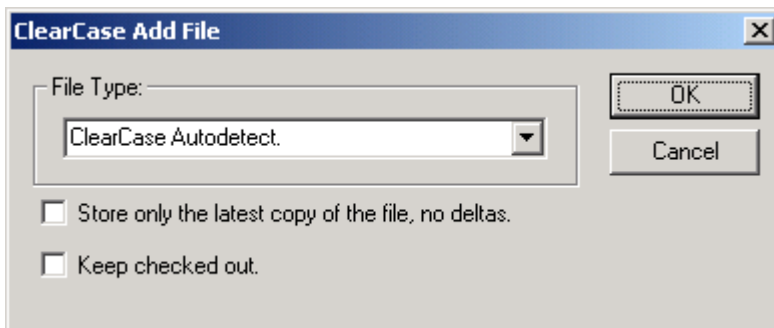


5 Optionally, type a comment in the **Comment** dialog box.

6 Click **OK**


The **Comment** dialog box closes. The **ClearCase Add File** dialog box displays. [Figure 3.9](#) shows this dialog box.

**Figure 3.9 The ClearCase Add File Dialog Box**



7 Use the **ClearCase Add File** dialog box to select the ClearCase file addition options you want.


8 Click **OK**

The **ClearCase Add File** dialog box closes. The plug-in adds the selected file and comment (if any) to ClearCase and changes the file's check out status icon to:  .

## Adding a Project using the Add... Command

To add a project to ClearCase using the Add . . . command, follow these steps:

- 1 Run the CodeWarrior IDE.
- 2 Open the CodeWarrior project whose project file you want to add to ClearCase. The CodeWarrior IDE displays the project window for this project.
- 3 Select **VCS > Project > Add...**  
The **Comment** dialog box appears. [Figure 3.7 on page 27](#) shows this dialog box.
- 4 Optionally, type a comment in the **Comment** dialog box.
- 5 Click **OK**  
The **Comment** dialog box closes. The **ClearCase Add File** dialog box displays. [Figure 3.9 on page 30](#) shows this dialog box.
- 6 Use the **ClearCase Add File** dialog box to select the ClearCase file addition options you want.
- 7 Click **OK**

The **ClearCase Add File** dialog box closes. The plug-in adds the project file and comment (if any) to ClearCase and changes the project's check out status icon to: . This icon is in the lower-left corner of the project window.

## Adding Files and Projects using the Recursive Add... Command

To add files and projects to ClearCase using the recursive Add . . . command, follow these steps:

- 1 Run the CodeWarrior IDE.
- 2 Select **VCS > Recursive > Add...**

---

**NOTE** You do not have to open a project before issuing the recursive Add . . . command because this form of the command does not affect a specific project. Instead, it affects all subdirectories and files in the plug-in's view.

---

- 3 The **Comment** dialog box appears. [Figure 3.7 on page 27](#) shows this dialog box.
- 4 Optionally, enter a comment in the **Comment** dialog box.
- 5 Click **OK**  
The **Comment** dialog box closes. The **ClearCase Add File** dialog box displays. [Figure 3.9 on page 30](#) shows this dialog box.
- 6 Use the **ClearCase Add File** dialog box to select the ClearCase file addition options you want.
- 7 Click **OK**  
The **ClearCase Add File** dialog box closes.  
The plug-in adds all files and projects in its ClearCase view to ClearCase.

---

## Checkin

You can apply the `Checkin` command to a file, to a project, or recursively.

If applied to a file, the `Checkin` command checks in the changes made to the selected file into ClearCase.

If applied to a project, the `Checkin` command checks the changes made to the project file (that is, to the `.mcp` file) into ClearCase.

If applied recursively, the `Checkin` command checks in each file and project in each subdirectory of the plug-in's ClearCase view that is currently checked out.

### Checking in a File

To check in a file, follow these steps:

- 1 Open the project that contains the file you want to check in.  
The CodeWarrior IDE displays the project window for this project.
- 2 In the project window, click the name of the file you want to check in.  
The IDE highlights this file name.

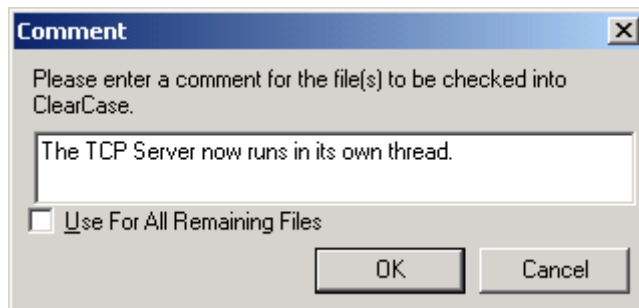


**TIP** To check in several files at once, highlight the name of each file and then issue the `Checkin` command.


---

- 3 Select **VCS > Checkin**
- 4 The **Comment** dialog box appears. [Figure 3.10](#) shows this dialog box.

**Figure 3.10 The Comment Dialog Box**



- 5 Optionally, enter a comment in the **Comment** dialog box.
- 6 Click **OK**

The **Comment** dialog box closes. The plug-in checks in the selected file and changes its check out status icon to: .

---

**NOTE** If you try to check in a file that has not changed, the check in fails. In this case, apply the [Undo Checkout](#) command to the file.

---


## Checking in a Project

To check a project into ClearCase, follow these steps:

- 1 Run the CodeWarrior IDE.
- 2 Open the CodeWarrior project whose project file you want to check in.  
The CodeWarrior IDE displays the project window for this project.
- 3 Select **VCS > Project > Checkin**  
The **Comment** dialog box appears. [Figure 3.10 on page 33](#) shows this dialog box.

4 Optionally, type a comment in the **Comment** dialog box.

5 Click **OK**

The **Comment** dialog box closes. The plug-in checks the project file and comment (if any) into ClearCase and changes the project's check out status icon to: . This icon is in the lower-left corner of the project window.

---

**NOTE** If you try to check in a project file that has not changed, the check in fails. In this case, apply the [Undo Checkout](#) command to the project file.

---

### Checking in Files and Projects Recursively

To check files and projects into ClearCase recursively, follow these steps:

- 1 Run the CodeWarrior IDE.
- 2 Select **VCS > Recursive > Checkin**

The **Comment** dialog box appears. [Figure 3.10 on page 33](#) shows this dialog box.

---

**NOTE** You do not have to open a project before issuing the recursive **Checkin** command because this form of the command does not affect a specific project. Instead, it affects all files and projects in the plug-in's view.

---

3 Optionally, enter a comment in the **Comment** dialog box.

4 Click **OK**

The **Comment** dialog box closes.

The plug-in checks in all files and projects in its ClearCase view.

---

**NOTE** The recursive checkin command fails for each checked out file that has not changed. For these files, issue the [Undo Checkout](#) command.

---

## Checkout

You can apply the `Checkout` command to a file, to a project, or recursively.

If applied to a file, the `Checkout` command creates a modifiable copy of the file in the plug-in's ClearCase view.

If applied to a project, the `Checkout` command creates a modifiable version of the project file (that is, of the `.mcp` file) in the plug-in's ClearCase view.

If applied recursively, the `Checkout` command checks out each file and project in each subdirectory of the plug-in's ClearCase view.

See [“Checkin” on page 32](#) for instructions that explain how to check in a file that you have checked out.

### Checking out a File

To check out a file, follow these steps:


- 1 Open the project that contains the file you want to check out.  
The CodeWarrior IDE displays the project window for this project.
- 2 In the project window, click the name of the file to check out.  
The IDE highlights this file name.

---

**TIP** To check out multiple files, highlight the name of each file and then issue the `Checkout` command.

---

- 3 Select **VCS > Checkout**


The plug-in checks out the selected file and changes its check out status icon to:  .

### Checking out a Project

To check out a project, follow these steps:

- 1 Open the project that you want to check out.  
The CodeWarrior IDE displays the project window for this project.

2 Select **VCS > Project > Checkout**

The plug-in checks out the project (that is, the project's .mcp file) and changes the project's check out status icon to: . This icon appears in the lower-left corner of the project window.

### Checking out Files and Projects Recursively

To check out files and projects recursively, follow these steps:

- 1 Run the CodeWarrior IDE.
- 2 Select **VCS > Recursive > Checkout**

---

**NOTE** You do not have to open a project before issuing the recursive `Checkout` command because this form the command does not affect a specific project. Instead, the command affects all files and projects in the plug-in's view.

---

3 Click **OK**

The plug-in checks out all files and projects in its ClearCase view.

---

## Connect

Although a `Connect` command appears in the IDE's VCS menu, you do not ever have to issue it. This is because the plug-in automatically connects to the ClearCase view specified in its configuration when necessary.

---

## Difference

You can apply the `Difference` command to a single file only. You cannot apply it to multiple files, to a project, or recursively.

The `Difference` command displays the differences between the copy of the selected file in the plug-in's ClearCase view and the latest version of this file in ClearCase.

To use the `Difference` command, follow these steps:

- 1 Open the project that contains the file to which you want to apply the `Difference` command.

The CodeWarrior IDE displays the project window for this project.

- 2 In the project window, click the name of the file to which to apply the `Difference` command.

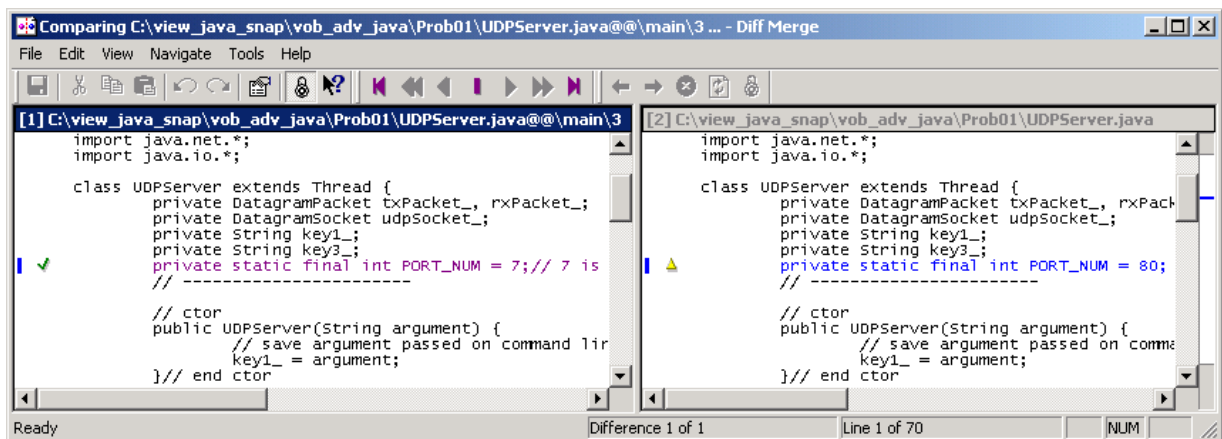
The IDE highlights this file name.

- 3 Select **VCS > Difference**

The plug-in compares the local copy of the selected file to the latest version of this file in ClearCase. The plug-in then displays the results in the ClearCase **Diff Merge** window.

[Figure 3.11](#) shows the ClearCase **Diff Merge** window. Refer to the ClearCase documentation for instructions that explain how to use this window.

**Figure 3.11 The ClearCase Diff Merge Window**



## Disconnect

Although a `Disconnect` command appears in the IDE's VCS menu, you do not ever have to issue it. This is because the plug-in automatically disconnects from the ClearCase view specified in its configuration when necessary.

## Get

You can apply the `Get` command to a file, to a project, or recursively.

If applied to a file, the `Get` command copies the latest version of the file from ClearCase into the plug-in's ClearCase view.

If applied to a project, the `Get` command copies the latest version of the project file (that is, of the `.mcp` file) from ClearCase into the plug-in's ClearCase view.

If applied to recursively, the `Get` command copies the latest version of each file and project from ClearCase into the plug-in's ClearCase view.

### Getting a File

To get the latest version of a file, follow these steps:

- 1 Open the project that contains the file for which you want to get the latest version.  
The CodeWarrior IDE displays the project window for this project.
- 2 In the project window, click the name of the file for which to get the latest version.  
The IDE highlights this file name.

---

**TIP** To get the latest version of several files, highlight the name of each file and then issue the `Get` command.

---

- 3 Select **VCS > Get**

The plug-in copies the latest version of the selected file into the plug-in's ClearCase view.

### Getting a Project

To get the latest version of a project, follow these steps:

- 1 Open the project for which you want to get the latest version.  
The CodeWarrior IDE displays the project window for this project.

**2 Select VCS > Project > Get**

The plug-in copies the latest version of the project (that is, of the project's .mcp file) from ClearCase into the plug-in's ClearCase view.

**Getting Files and Projects Recursively**

To get the latest version of each file and project from the plug-in's ClearCase view recursively, follow these steps:

- 1 Run the CodeWarrior IDE.**
- 2 Select VCS > Recursive > Get**

---

**NOTE** You do not have to open a project before issuing the recursive `Get` command because this form of the command does not affect a specific project. Instead, it affects all files and projects in the plug-in's ClearCase view.

---

**3 Click OK**

The plug-in copies the latest version of each file and project from ClearCase into the plug-in's ClearCase view

---

## History

You can apply the `History` command to a file or to a project. You cannot apply it recursively.

If applied to a file, the `History` command lists each revision and check-in comment made to the file since it was first added to ClearCase.

If applied to a project, the `History` command lists each revision and check-in comment made to the project's .mcp file since it was first added to ClearCase.

## Displaying the Revision History of a File

To display the revision history of a file, follow these steps:

- 1 Open the project that contains the file for which you want to obtain a revision history.

The CodeWarrior IDE displays the project window for this project.

- 2 In the project window, click the name of the file for which to you want to obtain a revision history.

The IDE highlights this file name.

---

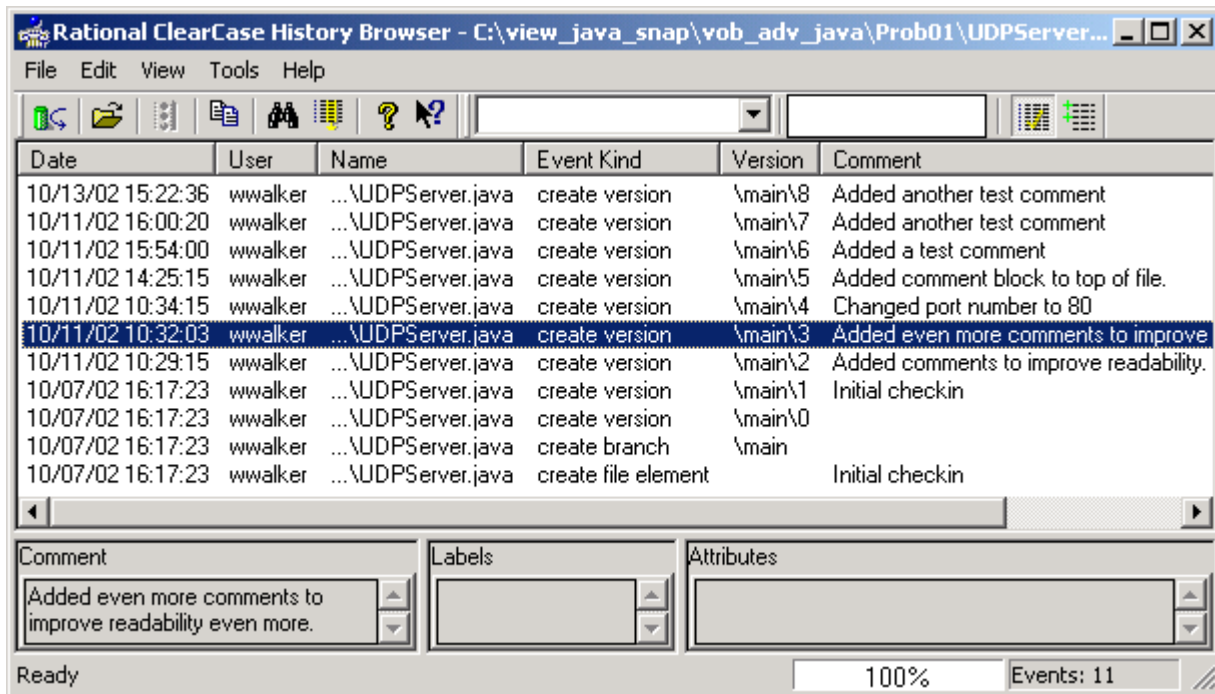
**TIP** To get the revision history of several files, highlight the name of each file and then issue the `History` command.

---

- 3 Select **VCS > History**

The plug-in displays the file's revision history in the **ClearCase History Browser** window. [Figure 3.12](#) shows this window.

**Figure 3.12 The ClearCase History Browser Window**





## Displaying the Revision History of a Project

To display the revision history of a project, follow these steps:

- 1 Open the project for which you want to obtain a revision history.  
The CodeWarrior IDE displays the project window for this project.
- 2 Select **VCS > History**  
The plug-in displays the revision history of the project's .mcp file in the **ClearCase History Browser** window. [Figure 3.12 on page 40](#) shows this window.

---

## Subtree Add

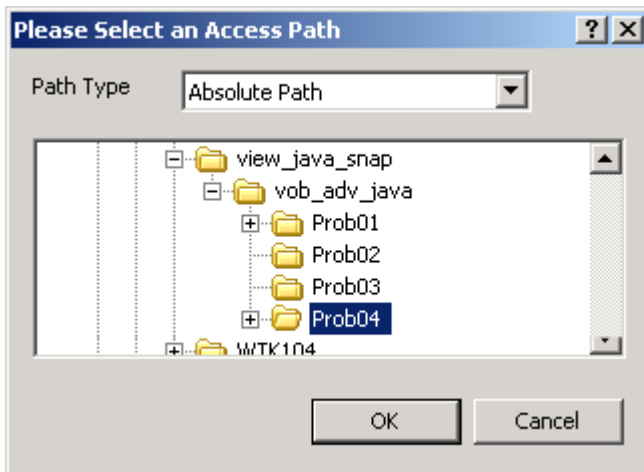
The `Subtree Add` command is a recursive command. You can apply it to any subdirectory of the plug-in's ClearCase view. You cannot apply the command to an individual file or to a project.

The `Subtree Add` command first lets you select a subdirectory of the plug-in's ClearCase view. Next, the command recurses into each subdirectory of the selected "root" directory and adds each file, project, and subdirectory not already under source code control to ClearCase.

To add a subtree of the plug-in's ClearCase view to ClearCase, follow these steps:

- 1 Run the CodeWarrior IDE.
- 2 Select **VCS > Recursive > Subtree Add**  
The **Please Select an Access Path** dialog box displays. [Figure 3.13](#) shows this dialog box.

**Figure 3.13 The Please Select an Access Path Dialog Box**



---

**NOTE** You do not have to open a project before issuing the `Subtree Add` command because this command does not affect a specific project. Instead, it affects the selected subtree of the plug-in's ClearCase view.

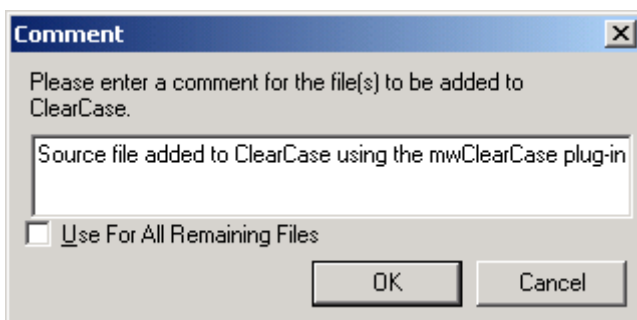
---

3 Use the **Please Select and Access Path** dialog box to select a subtree of the plug-in's ClearCase view to use as the starting point of a recursive add.

4 Click **OK**

The **Please Select and Access Path** dialog box closes. The **Comment** dialog box appears. [Figure 3.14](#) shows this dialog box.

**Figure 3.14 The Comment Dialog Box**



5 Optionally, type a comment in the **Comment** dialog box.

**6 Click OK**

The **Comment** dialog box closes.

The plug-in adds each file, project file, and subdirectory in the selected subtree to the plug-in's ClearCase view.

---

## Subtree Add...

The `Subtree Add...` command is a recursive command. You can apply it to any subdirectory of the plug-in's ClearCase view. You cannot apply the command to an individual file or to a project.

The `Subtree Add...` command first lets you select a subdirectory of the plug-in's ClearCase view. Next, the command recurses into each subdirectory of the selected "root" directory and adds each file, project, and subdirectory not already under source code control to ClearCase.

---

**NOTE** The `Subtree Add...` command is different from the [Subtree Add](#) command in that the `Subtree Add...` command gives you more control over how each file is added to ClearCase.

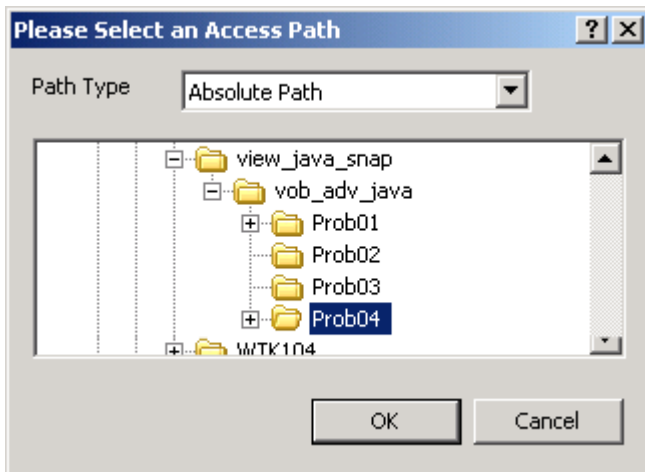
---

To add a subtree of the plug-in's ClearCase view to ClearCase, follow these steps:

- 1 Run the CodeWarrior IDE.
- 2 Select **VCS > Recursive > Subtree Add...**

The **Please Select an Access Path** dialog box displays. [Figure 3.13](#) shows this dialog box.

**Figure 3.15 The Please Select an Access Path Dialog Box**



---

**NOTE** You do not have to open a project before issuing the `Subtree Add...` command because this command does not affect a specific project. Instead, it affects the selected subtree of the plug-in's ClearCase view.

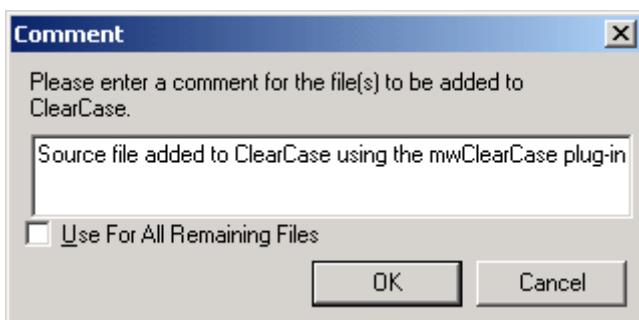
---

**3** Use the **Please Select and Access Path** dialog box to select a subtree of the plug-in's ClearCase view to use as the starting point for a recursive add.

**4** Click **OK**

The **Please Select and Access Path** dialog box closes. The **Comment** dialog box appears. [Figure 3.16](#) shows this dialog box.

**Figure 3.16 The Comment Dialog Box**

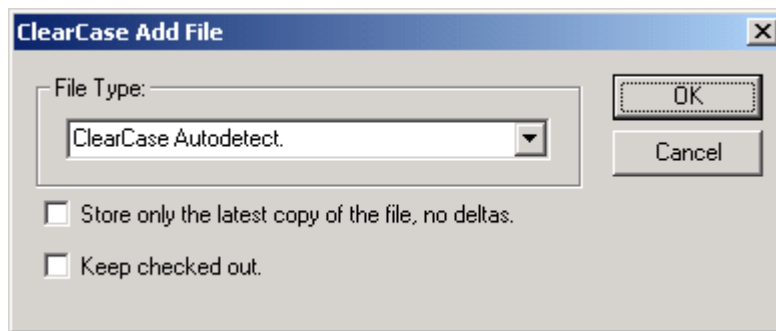


**5** Optionally, type a comment in the **Comment** dialog box.

**6 Click OK**

The **Comment** dialog box closes. The **ClearCase Add File** dialog box displays. [Figure 3.17](#) shows this dialog box.

**Figure 3.17 The ClearCase Add File Dialog Box**



**7** Use the **ClearCase Add File** dialog box to select the ClearCase file addition options you want.

**8 Click OK**

The **ClearCase Add File** dialog box closes.

The plug-in adds each file, project file, and subdirectory of the selected subtree to the plug-in's ClearCase view.

---

## Subtree Checkin

The `Subtree Checkin` command is a recursive command. You can apply it to any subdirectory of the plug-in's ClearCase view. You cannot apply the command to an individual file or to a project.

The `Subtree Checkin` command first lets you select a subdirectory of the plug-in's ClearCase view. Next, the command recurses into each subdirectory of the selected "root" directory and checks in each file and project currently checked out.

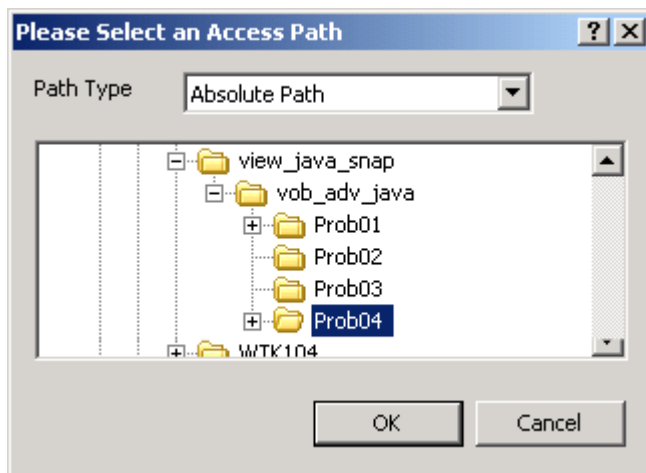
To check in a subtree of the plug-in's ClearCase view, follow these steps:

**1** Run the CodeWarrior IDE.

2 Select **VCS > Recursive > Subtree Checkin**

The **Please Select an Access Path** dialog box displays. [Figure 3.13](#) shows this dialog box.

**Figure 3.18 The Please Select an Access Path Dialog Box**



---

**NOTE** You do not have to open a project before issuing the `Subtree Checkin` command because this command does not affect a specific project. Instead, it affects the selected subtree of the plug-in's ClearCase view.

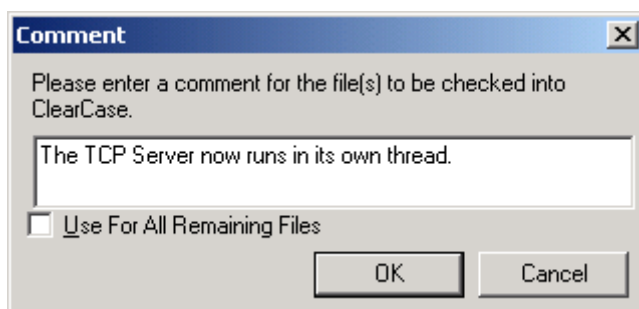
---

3 Use the **Please Select and Access Path** dialog box to select a subtree of the plug-in's ClearCase view to use as the starting point for a recursive check in.

4 Click **OK**

The **Please Select and Access Path** dialog box closes. The **Comment** dialog box appears. [Figure 3.19](#) shows this dialog box.

**Figure 3.19 The Comment Dialog Box**



5 Optionally, type a comment in the **Comment** dialog box.

6 Click **OK**

The **Comment** dialog box closes.

The plug-in recurses into each subdirectory of the selected “root” directory and checks in each file, project file, and subdirectory that is checked out.

---

**NOTE** The `Subtree Checkin` command fails for each checked out file that has not been changed. For such files, use the [Undo Checkout](#) command.

---

---

## Subtree Checkout

The `Subtree Checkout` command is a recursive command. You can apply it to any subdirectory of the plug-in’s ClearCase view. You cannot apply the command to an individual file or to a project.

The `Subtree Checkout` command first lets you select a subdirectory of the plug-in’s ClearCase view. Next, the command recurses into each subdirectory of the selected “root” directory and checks out each file, project, and subdirectory that is not already checked out.

To check out a subtree of the plug-in’s ClearCase view, follow these steps:

- 1 Run the CodeWarrior IDE.
- 2 Select **VCS > Recursive > Subtree Checkout**

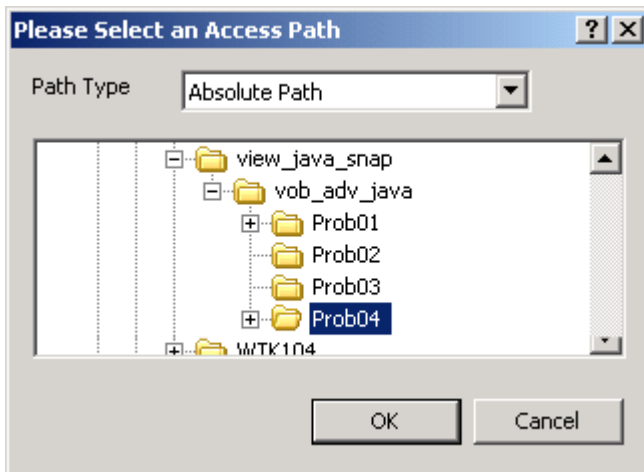
The plug-in displays the **Please Select an Access Path** dialog box. [Figure 3.20](#) shows this dialog box.

---

**NOTE** You do not have to open a project before issuing the `Subtree Checkout` command because this command does not affect a specific project. Instead, it affects the selected subtree of the plug-in’s ClearCase view.

---

**Figure 3.20 The Please Select an Access Path Dialog Box**



**3** Use the **Please Select and Access Path** dialog box to select a subtree of the plug-in’s ClearCase view to use as the starting point for a recursive check out.

**4** Click **OK**

The **Please Select and Access Path** dialog box closes.

The plug-in recurses into each subdirectory of the selected “root” directory and checks out each file, project file, and subdirectory not already checked out.

---

## Subtree Get

The `Subtree Get` command is a recursive command. You can apply it to any subdirectory of the plug-in’s ClearCase view. You cannot apply the command to an individual file or to a project.

The `Subtree Get` command first lets you select a subdirectory of the plug-in’s ClearCase view. Next, the command recurses into each subdirectory of the selected “root” directory and gets the latest version of each file and project.

To get the latest version of each file in a subtree of the plug-in’s ClearCase view, follow these steps:

**1** Run the CodeWarrior IDE.



2 Select **VCS > Recursive > Subtree Get**

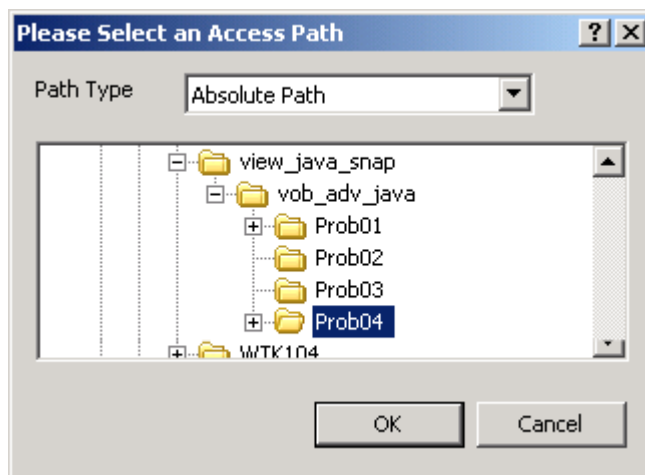
The plug-in displays the **Please Select an Access Path** dialog box. [Figure 3.21](#) shows this dialog box.

---

**NOTE** You do not have to open a project before issuing the `Subtree Get` command because this command does not affect a specific project. Instead, the command affects the selected subtree of the plug-in's ClearCase view.

---

**Figure 3.21 The Please Select an Access Path Dialog Box**



3 Use the **Please Select and Access Path** dialog box to select a subtree of the plug-in's ClearCase view to use as the starting point for a recursive get.

4 Click **OK**

The **Please Select and Access Path** dialog box closes.

The plug-in recurses into each subdirectory of the selected "root" directory and gets the latest version of each file and project.

---

## Properties

You can apply the `Properties` command to a single file or to a project. You cannot apply it to multiple files or recursively.

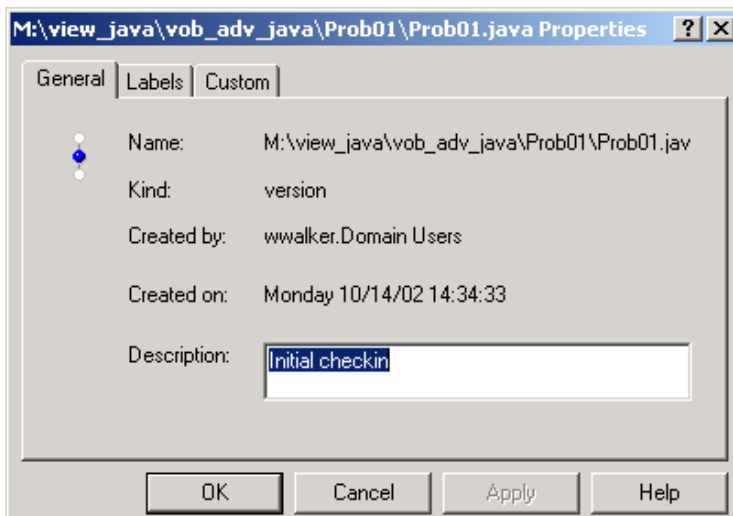
The `Properties` command displays the ClearCase property dialog box for the selected file or project.

### Displaying the Properties of a File

To display a file's properties, follow these steps:

- 1 Run the CodeWarrior IDE.
- 2 Open the project that contains the file whose properties you want to display.  
The CodeWarrior IDE displays the project window for this project.
- 3 In the project window, click the name of the file whose properties you want to display.  
The IDE highlights this file name.
- 4 Select **VCS > Properties**  
The plug-in displays the file's properties in the **ClearCase Properties** dialog box. [Figure 3.22](#) shows this dialog box.

**Figure 3.22 The ClearCase Properties Dialog Box**



---

**NOTE** Refer to the ClearCase documentation for instructions that explain how to use the **ClearCase Properties Dialog Box**.

---

## Displaying the Properties of a Project

To display a project's properties, follow these steps:

- 1 Run the CodeWarrior IDE.
- 2 Open the project whose properties you want to display.  
The CodeWarrior IDE displays the project window for this project.
- 3 Select **VCS > Project > Properties**

The plug-in displays the project's properties (that is, the properties of the project's .mcp file) in the **ClearCase Properties** dialog box. [Figure 3.22 on page 50](#) shows this dialog box.

---

**NOTE** Refer to the ClearCase documentation for instructions that explain how to use the **ClearCase Properties Dialog Box**.

---

---

## Status

You can apply the `Status` command to a file or to a project. You cannot apply it recursively.

The `Status` command reports the ClearCase status of the selected file or project.

### Displaying the Status of a File

To display the status of a file, follow these steps:

- 1 Run the CodeWarrior IDE.
- 2 Open the project that contains the file for which you want to obtain status.  
The CodeWarrior IDE displays the project window for this project.
- 3 In the project window, click the name of the file for which to you want to obtain status.

The IDE highlights this file name.

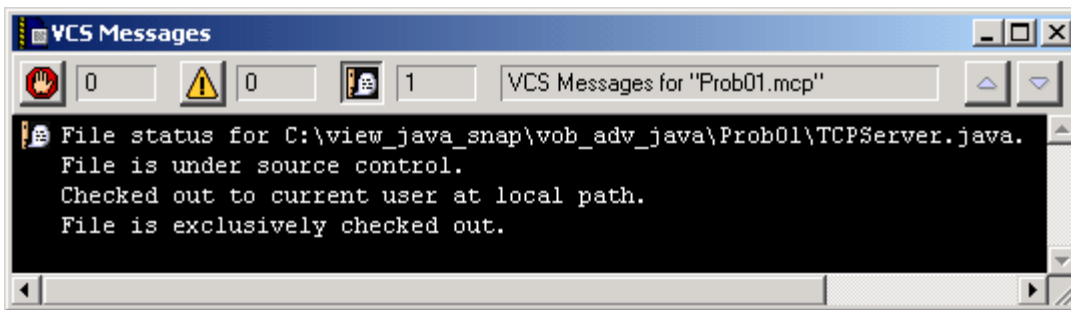
**TIP** To get the status of several files, highlight the name of each file and then issue the `Status` command.

---

#### 4 Select **VCS > Status**

The plug-in displays the file's status in the **VCS Messages** window. [Figure 3.12](#) shows this window.

**Figure 3.23 The Status Window**



### Displaying the Status of a Project

To display the status of a project, follow these steps:

- 1 Run the CodeWarrior IDE.
- 2 Open the project for which you want to obtain status.

The CodeWarrior IDE displays the project window for this project.

- 3 Select **VCS > Project > Status**

The plug-in displays the status of the project (that is, of the project's `.mcp` file) in the **VCS Messages** window. [Figure 3.23 on page 52](#) shows this window.

---

## Synchronize Selection

You can apply the `Synchronize Selection` command to a file or files in a project. You cannot apply the command to a project or recursively.

The command verifies that the check out status icon displayed in the project window for the selected file matches the status of that file in ClearCase. If there is a discrepancy, the plug-in displays the correct check out status icon.

The mwClearCase plug-in can display the wrong check out status icon if you change the check out status of a file outside of the CodeWarrior IDE (for example, by using the Rational ClearCase Explorer program). You can use the `Synchronize Selection` command to rectify this situation.

To issue the `Synchronize Selection` command, follow these steps:

- 1 Run the CodeWarrior IDE.
- 2 Open the project that contains the file for which to synchronize status.  
The CodeWarrior IDE displays the project window for this project.
- 3 In the project window, click the name of the file for which to synchronize status.  
The IDE highlights this file name.

---

**NOTE** To synchronize the status of several files at once, highlight the name of each file and then issue the `Synchronize Selection` command.

---

#### 4 Select **VCS > Synchronize Selection**

The plug-in verifies that the check out status icon displayed for the selected file is correct. If not, the plug-in displays the correct check out status icon.

---

## Synchronize Status

The `Synchronize Status` command is applied to all files in a project and to the project itself. It cannot be applied to an individual file or recursively.

The command updates the check out status icon of each file in the project and of the project file itself (that is, of the project's `.mcp` file).

The mwClearCase plug-in can sometimes display incorrect check out status icons. For example, if you change the check out status of a file or project outside of the CodeWarrior IDE (for example, by using the Rational ClearCase Explorer program), the plug-in might display incorrect status icons. Use the `Synchronize Status` command to rectify this situation.

To issue the `Synchronize Status` command, follow these steps:

- 1 Run the CodeWarrior IDE.

- 2 Open the project whose status you want to synchronize.

The CodeWarrior IDE displays the project window for this project.

- 3 Select **VCS > Synchronize Status**

The plug-in updates the check out status icon for each file for which it finds a discrepancy.

---

## Undo Checkout

You can apply the `Undo Checkout` command to a file or project. You cannot apply the command recursively.

If applied to a checked out file, the `Undo Checkout` command changes the ClearCase status of the file to “not checked out.” All changes made to the file are lost.

If applied to a checked out project, the `Undo Checkout` command changes the ClearCase status of the project file (that is, of the project’s `.mcp` file) to “not checked out.” All changes made the project file are lost.

### Undoing a File Check Out

To undo a file check out, follow these steps:


- 1 Run the CodeWarrior IDE.
- 2 Open the project that contains the file for which to undo the check out.  
The CodeWarrior IDE displays the project window for this project.
- 3 In the project window, click the name of the file for which to undo the check out.  
The IDE highlights this file name.

---

**NOTE** To undo the check out for several files at once, highlight the name of each file and then issue the `Undo Checkout` command.

---

- 4 Select **VCS > Undo Checkout**

The plug-in undoes the check out of the selected file and changes its check out status icon to: .


## Undoing a Project Check Out

To undo a project check out, follow these steps:

- 1 Run the CodeWarrior IDE.
- 2 Open the project for which to undo the check out.

The CodeWarrior IDE displays the project window for this project.

- 3 Select **VCS > Project > Undo Checkout**

The plug-in undoes the check out of the selected project and changes its check out status icon to: . This icon appears in the lower-left corner of the project window.

---

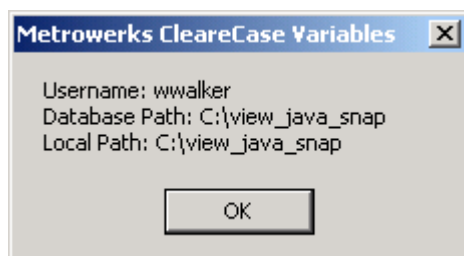
## Variables

The `Variables` command displays the username, database path, and local path assigned to the mwClearCase plug-in.

To issue the `Variables` command, select **VCS > Variables**. The **Metrowerks ClearCase Variables** dialog box displays.

[Figure 3.24](#) shows this dialog box.

**Figure 3.24 Variables Dialog Box for the mwClearCase Plug-in**







# Tutorial

---

This chapter consists of a tutorial. The tutorial shows you how to:

- Create a default configuration for the mwClearCase plug-in.
- Use the plug-in to perform common ClearCase tasks from within the CodeWarrior™ IDE.

---

**NOTE** To perform the lessons in this tutorial, you must have a ClearCase view available. Further, if you use a dynamic ClearCase view, you must start this view before beginning the tutorial.

---

The tutorial contains these lessons:

- [Creating a Default Plug-in Configuration](#)
- [Checking out a File](#)
- [Checking out a Project](#)
- [Checking in a File](#)
- [Checking in a Project](#)

## Creating a Default Plug-in Configuration

Although you can create a custom plug-in configuration for each every CodeWarrior project, it is often easier to create a default plug-in configuration. This approach is easier because the default plug-in configuration works for all of your CodeWarrior projects.

To create a default plug-in configuration, follow these steps:

- 1 Run the CodeWarrior IDE.
- 2 Close any open projects.

**3 Select **Edit > VCS Settings...****

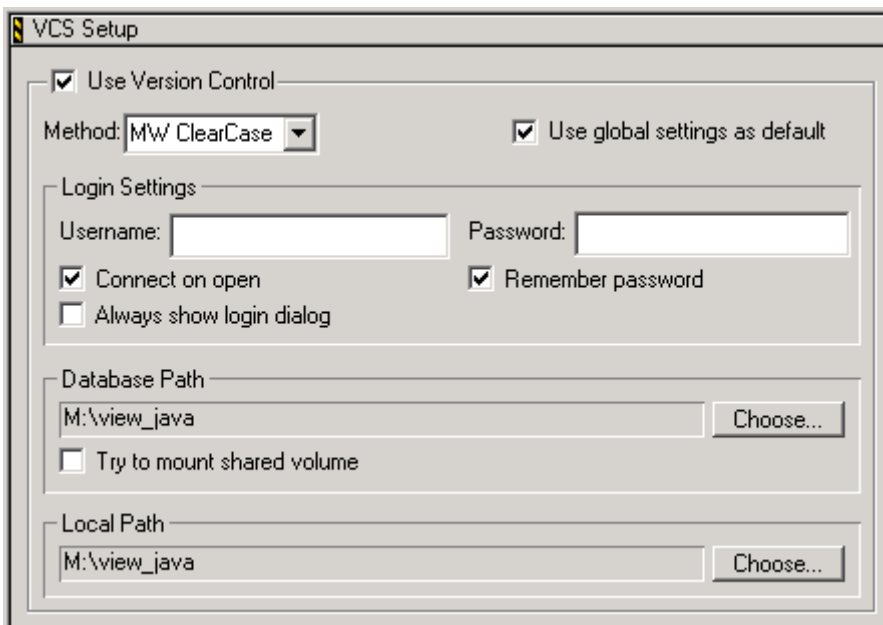
The VCS Settings window appears.

**4 Display the **VCS Setup** settings panel.**

To do this, select the item labeled **VCS Setup**. This item is in the pane labeled **VCS Settings Panels**. This pane is on the left side of the VCS Settings window.

[Figure 4.1](#) shows the VCS Setup settings panel.

**Figure 4.1 The VCS Setup Panel of the VCS Settings Window**



**5 Check the box labeled **Use Version Control****

The **Method** dropdown menu and the **Use global settings as default** check box become active.

**6 Select the item labeled `MW ClearCase` from the **Method** dropdown menu.**

The IDE activates the rest of the items in the **VCS Setup** panel.

**7 Check the box labeled **Use global settings as default**.**

This setting makes this plug-in configuration the default configuration. The IDE uses this configuration for each CodeWarrior project you open that does not have a custom plug-in configuration.

- 8 Leave the **Username** field empty.
- 9 Check the **Connect on open** check box.
- 10 Check **Remember password** check box.  
The plug-in enables the **Password** field.
- 11 Leave the **Password** field empty.
- 12 Leave the **Always show login dialog** check box unchecked.
- 13 Assign a ClearCase view to the default plug-in configuration.  
To do this, follow these steps:
  - a. Click the **Choose...** button in the **Database Path** group box.  
The **Select an Access Path** dialog box appears.
  - b. Use this dialog box to select the ClearCase view that you want to use for the rest of this tutorial.
  - c. Leave the **Try to mount shared volume** check box unchecked.
  - d. Click **OK**  
The full path to the selected ClearCase view appears in the read-only field to the left of the **Choose...** button.
- 14 Assign a local path to the default plug-in configuration.  
To do this, follow these steps:
  - a. Click the **Choose...** button in the **Local Path** group box.  
The **Select an Access Path** dialog box appears.
  - b. Use this dialog box to select the same ClearCase view that you selected for Database Path.
  - c. Click **OK**  
The full path to the selected ClearCase view appears in the read-only field to the left of the **Choose...** button.

---

**NOTE** You must select the same ClearCase view for **Database Path** and for **Local Path**. Otherwise, the mwClearCase plug-in will not work.

---

**15 Click OK**

The **VCS Settings** window closes. The IDE activates the mwClearCase plug-in globally and adds the VCS menu to the IDE's menu bar.

You have finished creating the default mwClearCase plug-in configuration.

## Checking out a File

Now that you have created a default plug-in configuration, you can use the plug-in to check out files from a CodeWarrior project while within the IDE.

To check out a file, follow these steps:

- 1 Run the CodeWarrior IDE.
- 2 Open one of the CodeWarrior projects in the ClearCase view selected in the last lesson.

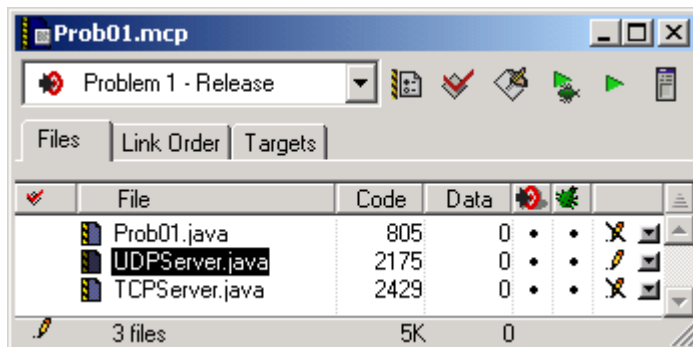
The IDE displays the project window of the selected project.

**3 Select VCS > Synchronize Status**

The project window displays the check out status icon for each file in the project and for the project itself.

[Figure 4.2](#) shows the project window for a CodeWarrior project. [Figure 3.2 on page 21](#) explains how to interpret each of the check out status icons.


**Figure 4.2 A CodeWarrior Project Window**



- 4 In the project window, click the name of one of the files in the project.  
The project window highlights this file name.

## 5 Select **VCS > Checkout**

The plug-in does these things:

- a. Copies the latest version of the selected file into its ClearCase view.
- b. Makes the file writable and marks it “checked out.”
- c. Changes the file’s check out status icon to:  .

The selected file is checked out. You can now use the CodeWarrior editor to change the file.

# Checking out a Project

To check out a project (that is, a project’s .mcp file), follow these steps:


1 Run the CodeWarrior IDE.

2 Open the project that you want to check out.

The IDE displays the project window for this project.

3 Select **VCS > Project > Checkout**

The plug-in does these things:

- a. Copies the latest version of the project’s .mcp file into its ClearCase view.
- b. Makes the .mcp file writable and marks it “checked out.”
- c. Changes the project’s check out status icon to:  . This icon is in the lower-left corner of the project window.

The project is checked out. You can now change the project’s configuration as necessary.

# Checking in a File

Once you finish changing a file that you have checked out, you can check it back in.

To check a file into ClearCase, follow these steps:

1 Run the CodeWarrior IDE.

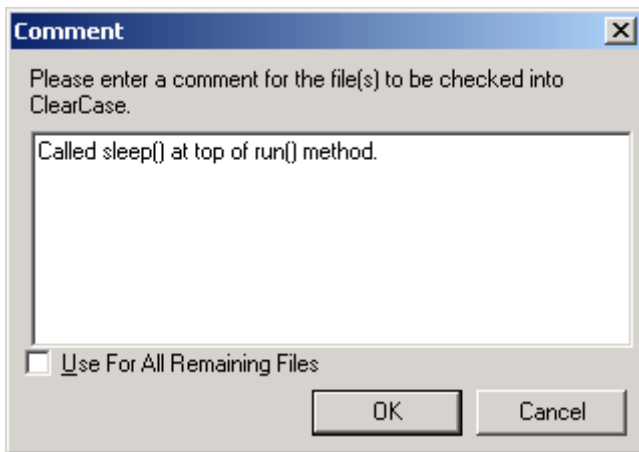
## Tutorial


### Checking in a File

---

- 2 Open the CodeWarrior project that contains the file you want to check in.  
The IDE displays the project window of the selected project.
- 3 In the project window, click the name of the file you want to check in.  
The project window highlights this file name.
- 4 Select **VCS > Checkin**  
The plug-in displays the **Comment** dialog box. [Figure 4.3](#) shows this dialog box.

**Figure 4.3 The Comment Dialog Box**



- 5 Optionally, enter a check in comment.
- 6 Click **OK**  
The plug-in instructs ClearCase to do these things:
  - a. Create a new version of the file being checked in.
  - b. Mark the file “not checked out.”The plug-in then changes the file’s check out status icon to: .

---

**NOTE** If you check out a file, do not change it, and then check it back in, the `check in` command fails. Use the [Undo Checkout](#) command in this case.

---

# Checking in a Project

Once you finish changing a project's configuration, you can check it back in.

To check a project into ClearCase, follow these steps:

1 Run the CodeWarrior IDE.

2 Open the CodeWarrior project that you want to check in.

The IDE displays the project window of the selected project.

3 Select **VCS > Project > Checkin**


The plug-in displays the Comment dialog box. [Figure 4.3 on page 62](#) shows this dialog box.

4 Optionally, enter a checkin comment.

5 Click **OK**

The plug-in instructs ClearCase to do these things:

- a. Create a new version of the project file being checked in.
- b. Mark the project file "not checked out."

The plug-in then changes the projects's check out status icon to: . This icon is in the lower-left corner of the project window

---

**NOTE** If you check out a project, do not change it, and then check it back in, the `check in` command fails. Use the [Undo Checkout](#) command in this case.

---





# Index

---

## A

- about
  - ClearCase 6
  - the mwClearCase plug-in 5
- about command 26
- activating the mwClearCase plug-in 10–17
- add command 27
- add... command 29
- Always show login dialog configuration option 11

## C

- check out status icons
  - in a project window 20
  - interpretation 21
- checkin command 32
- checkout command 35
- ClearCase
  - learning more about 6
- command locations 21–24
  - editor window 23
  - project window context menu 22
  - VCS menu 21
- commands
  - about 26
  - add 27
  - add... 29
  - checkin 32
  - checkout 35
  - connect 36
  - difference 36
  - disconnect 37
  - get 38
  - history 39
  - project window context menu 22
  - properties 49
  - status 51
  - subtree add 41
  - subtree add... 43
  - subtree checkin 45
  - subtree checkout 47
  - subtree get 48
  - synchronize selection 52
  - synchronize status 53
  - undo checkout 54

- variables 55
- commands supported by the plug-in 24–55
- configuration options
  - Always show login dialog 11
  - Connect on open 11
  - Database Path 11
  - Local Path 12
  - Login Settings 11
  - Method 11
  - Password 11
  - Remember password 11
  - Try to mount shared volume 11
  - Use global settings as default 11
  - Use Version Control 11
  - Username 11
- connect command 36
- Connect on open configuration option 11
- creating a project-specific plug-in configuration 15–17
- creating the default plug-in configuration 12–15

## D

- Database Path configuration option 11
- deactivating the mwClearCase plug-in 17–18
- default plug-in configuration, creating 12–15
- difference command 36
- disconnect command 37

## G

- general ClearCase information 6
- get command 38

## H

- history command 39
- how to
  - activate the mwClearCase plug-in 10–17
    - check in a file 61–63
    - check in a project 63
    - check out a file 60–61
    - check out a project 61
  - create a default plug-in configuration 12–15, 57–60
  - create a project-specific plug-in configuration 15–17
  - deactivate the mwClearCase plug-in 17–18
  - install the mwClearCase plug-in 8–10

---

interpret the check out status icons 21  
obtain the mwClearCase plug-in 7–8

## I

information about ClearCase 6  
installing the mwClearCase plug-in 8–10

## L

learning about ClearCase 6  
Local Path configuration option 12  
location of commands 21–24  
Login Settings configuration option 11

## M

meaning of check out status icons 21  
menu, VCS 21  
Method configuration option 11  
more information about ClearCase 6  
mwClearCase plug-in  
  about 5  
  activating 10–17  
    for a specific project 15–17  
    globally 12–15  
  command locations 21–24  
    editor window 23  
    project window context menu 22  
    VCS menu 21  
  command reference 24–55  
  commands  
    about 26  
    add 27  
    add... 29  
    checkin 32  
    checkout 35  
    connect 36  
    difference 36  
    disconnect 37  
    get 38  
    history 39  
    properties 49  
    status 51  
    subtree add 41  
    subtree add... 43  
    subtree checkin 45  
    subtree checkout 47  
    subtree get 48  
    synchronize selection 52

  synchronize status 53  
  undo checkout 54  
  variables 55  
  deactivating 17–18  
  installing 8–10  
  obtaining 7–8  
  overview 5–6  
  project window  
    check out status icons 20  
  project window context menu commands 22  
  setting up 7–18  
  system requirements 5  
  tutorial 57–63  
    checking in a file 61–63  
    checking in a project 63  
    checking out a file 60–61  
    checking out a project 61  
    default plug-in configuration, creating 57–60  
    using 19–55

## O

obtaining the mwClearCase plug-in 7–8  
overview of the mwClearCase plug-in 5–6

## P

Password configuration option 11  
plug-in tutorial 57–63  
project specific plug-in configuration, creating 15–17  
project window  
  check out status icons 20  
  context menu 22  
  context menu and ClearCase commands 22  
project window context menu 22  
properties command 49

## R

Remember password option 11  
requirements for using the mwClearCase plug-in 5

## S

setting up the mwClearCase plug-in 7–18  
settings panels  
  VCS Setup 10  
status command 51  
subtree add command 41  
subtree add... command 43

---

subtree checkin command 45  
subtree checkout command 47  
subtree get command 48  
synchronize selection command 52  
synchronize status command 53  
system requirements 5

Use global settings as default 11  
Use Version Control 11  
Username 11

## T

### tasks

- activating the mwClearCase plug-in 10–17
- creating a default plug-in configuration 12–15
- creating a project-specific plug-in configuration 15–17
- deactivating the mwClearCase plug-in 17–18
- installing the mwClearCase plug-in 8–10
- obtaining the mwClearCase plug-in 7–8

Try to mount shared volume configuration option 11

tutorial

- checking in a file 61–63
- checking in a project 63
- checking out a file 60–61
- checking out a project 61
- default plug-in configuration, creating 57–60

## U

undo checkout command 54

Use global settings as default configuration option 11

Use Version Control configuration option 11

Username configuration option 11

using the mwClearCase plug-in 19–55

## V

variables command 55

VCS menu 21

VCS Setup settings panel

- explanation of options 11–12
- figure 10
- options
  - Always show login dialog 11
  - Connect on open 11
  - Database Path 11
  - Local Path 12
  - Login Settings 11
  - Method 11
  - Password 11
  - Remember password 11
  - Try to mount shared volume 11

