HP OpenView Configuration Management Administrator

for the Windows operating system

Version: 5.00

Configuration Server Database Editor Guide

Document Release Date: June 2007

Software Release Date: April 2007

Legal Notices

Warranty

The only warranties for HP products and services are set forth in the express warranty statements accompanying such products and services. Nothing herein should be construed as constituting an additional warranty. HP shall not be liable for technical or editorial errors or omissions contained herein.

The information contained herein is subject to change without notice.

Restricted Rights Legend

Confidential computer software. Valid license from HP required for possession, use or copying. Consistent with FAR 12.211 and 12.212, Commercial Computer Software, Computer Software Documentation, and Technical Data for Commercial Items are licensed to the U.S. Government under vendor's standard commercial license.

Copyright Notices

Trademark Notices

Linux is a registered trademark of Linus Torvalds.

Microsoft®, Windows®, and Windows® XP are U.S. registered trademarks of Microsoft Corporation.

OpenLDAP is a registered trademark of the OpenLDAP Foundation.

DHTML Calendar

Documentation Updates

This guide's title page contains the following identifying information:

- Software Version number, which indicates the software version
- Document release date, which changes each time the document is updated
- Software release date, which indicates the release date of this version of the software

To check for recent updates or to verify that you are using the most recent edition, visit the following URL:

ovweb.external.hp.com/lpe/doc_serv/

You will also receive updated or new editions if you subscribe to the appropriate product support service. Contact your HP sales representative for details.

Table 1 below indicates changes made to this document for this version 5.00.

Table 2 on page 5 indicates changes made to this document in previous versions.

Table 1 Document Changes for Version 5.00

Chapter	Version	Changes
Removed	5.00	The previous Chapter 10, Support for the Inventory Manager, was removed from this guide. Refer to the *HP OVCM Inventory Manager Installation and Configuration Guide* for details on that product.
Removed	5.00	The following Appendixes were removed from this guide: • A: Using System Explorer Help. • B: System Explorer Shortcut Menus
Chapter 2	5.00	Page 29, The CM Admin CSDB Editor Toolbar; several icons on the CM Admin CSDB Editor window and toolbar have changed. Images throughout the guide have been updated to reflect the new icons.
Chapter 2	5.00	Page 44, Table 15 PRIMARY file default domains. Added the APPMGMT domain, which is used to deploy server applications using the Application Management Profiles component of the CM Server Management Agent.
Chapter 4	5.00	Page 69, Naming Instances, new topic gives guidelines on valid characters to use to name instances in the database.

Table 2 Document Changes to Previous Versions

Chapter	Version	Changes
Chapter 2	4.0.1	Page 29, Table 9: Help Menu Options: deleted the row for "Search For Help On". This option was removed from the Help menu.
Chapter 2	4.0	Page 30, Setting Preferences: The Options Dialog Box. Added Notify to the list of tabs on the View Options dialog box.
Chapter 2	4.0	Page 32, Table 11. Added a row for the new window resizing option, "Lock the splitter bar distance to the right side of the screen."
Chapter 2	4.0	Page 39, Notify Tab: this is a new section. The Notify tab on the Options dialog box enables customization of the UID and STARTDIR values in the command line generated for the Notify Subscribers. Select the Custom option to notify agents by machine name, or specify a starting directory.
Chapter 3	4.0.1	Page 44, The CM Configuration Server Database: modified the description of the PRIMARY File for the CM-CSDB versions 4.0 and 4.0.1: • Added Table 15 to summarize the PRIMARY File default domains. • Added default domains for CLIENT and PRDMAINT. • Added a note explaining the PRDMAINT Domain was previously named the NOVADIGM Domain, and an existing NOVADIGM Domain will remain for self-maintenance of agents at versions prior to 4.0. • Added Table 16 to describe the PRIMARY File optional domains, including the MACHINE, NOVADIGM, OS, USAGE and PATCHMGR Domains. • Page 46, added text: If you have installed HP OpenView Configuration Management for Servers you will encounter the following PRIMARY File domains: APPMGMT, BASELINE, and CFM. Refer to the CM Server Management documentation for more information on these domains and classes.

Chapter	Version	Changes
Chapter 4	4.0.1	Page 81, Table 20: updated the list of POLICY classes in the CM-CSDB versions 4.0 and 4.0.1: • Added the optional classes used with the OS Manager product: Manufacturer (MANUFACT), Machine Models (MODEL), Machine Roles (ROLE), and Machine Subnets (SUBNET). • Added the Multicast (MULTCAST) class for use with a CM Multicast Server. Added the Mobile Device Confg (MBLCONFG) class for use with wireless PDAs being managed by the HP OpenView Mobility Server.
Chapter 5	4.0	Page 117, Defining MSI Basic Behaviors on the New Application Wizard: replaced Figure 29 showing the New Application Wizard with MSI Basic Behaviors section.
Chapter 5	4.0	Page 151, Set Notification Details. This dialog box now displays the values for UID and STARTDIR that will be used with the Notify command. These values can be changed on the Notify tab of the Options dialog box.
Chapter 9		Page 218, Table 33: added rows defining Delete Always, Delete Never and Delete (Default), which apply when you perform a Radreg32 Delete operation from the Registry File Editor.
Appendix A	4.0	Page 228, Editing a Component: Added a caution. The component editor is not aware of a component's target operating system. After an edit session, you may need to restore the original values of attributes such as ZRSCMMEM (PDS Member Name), ZRSCCRC (Resource CRC), ZRSCPADM (PDS AdminID), and ZPUBVER if the change is unintentional.

Support

You can visit the HP Software support web site at:

www.hp.com/managementsoftware/services

This web site provides contact information and details about the products, services, and support that HP Software offers.

HP Software online software support provides customer self-solve capabilities. It provides a fast and efficient way to access interactive technical support tools needed to manage your business. As a valued support customer, you can benefit by using the support site to:

- Search for knowledge documents of interest
- Submit and track support cases and enhancement requests
- Download software patches
- Manage support contracts
- Look up HP support contacts
- Review information about available services
- Enter into discussions with other software customers
- Research and register for software training

Most of the support areas require that you register as an HP Passport user and sign in. Many also require a support contract.

To find more information about access levels, go to:

www.hp.com/managementsoftware/access_level

To register for an HP Passport ID, go to:

www.managementsoftware.hp.com/passport-registration.html

Contents

1 Introduction

At the end of this chapter, you will:

- Be familiar with the HP OpenView Configuration Management Administrator Configuration Server Database Editor (CM Admin CSDB Editor) and its uses.
- Understand the Configuration Management (CM) desired state.
- Be familiar with CM terminology.

About this Guide

Who this Guide is for

The CM Admin CSDB Editor Guide was written for systems administrators who want to create and maintain a desired state for their enterprises.

What this Guide is about

This guide introduces the CM Admin CSDB Editor—the desktop-enterprise configuration and management tool of the HP Openview Configuration Management Administrator—and describes the concepts of maintaining a desired state in the CM-CSDB, and the actions you take when administering the CM environment.

Conventions

You should be aware of the following conventions used in this book.

Table 3 Usage

Element	Style	Example
Drives (system, mapped, CD)	Italicized placeholder	*SystemDrive*:\Program Files\HP might refer to C:\Program Files\HP on your computer.
Files (in the CM-CSDB)	All uppercase	PRIMARY
Domains (in the CM-CSDB)	All uppercase	PRIMARY.SOFTWARE Might also be referred to as the SOFTWARE Domain in the PRIMARY File.
Classes (in the CM-CSDB)	All uppercase	PRIMARY.SOFTWARE.ZSERVICE Might also be referred to as the ZSERVICE Class in the SOFTWARE Domain in the PRIMARY File.

The table below describes terms that could be used interchangeably throughout this book.

Table 4 Terminology

Term	Might also be called
Application	software, service
Agent	CM Application Manager and/or CM Application Self-service Manager
Computer	workstation, server
Configuration Server	Manager

About the CM Admin CSDB Editor

The CM Admin CSDB Editor is an interactive graphical tool for manipulating and inspecting the contents of the CM-CSDB.

Use the CM Admin CSDB Editor to:

- Define users (that is, identify desktop computers to be managed by CM) in the CM-CSDB.
- Assign applications to individual users and workgroups.
- Define applications and their files.
- Define special file properties.
- Group applications based on your organization's policies.
- Grant users access to applications, or to groups of applications.
- Manage the configuration of users' desktops.

CM enables you to configure applications, connect users to applications, and define hardware and software auditing requirements for managed devices throughout your enterprise based on the policies that your organization sets for its desired configurations.

The Desired State

Your **desired state** records the identities and intended configurations of the computers whose relationships are managed by CM. It can be as sophisticated, or as simple, as you want. At a minimum, a CM desired state includes the following five elements:

- **Users**
  The identity of the computers that are being managed.
  (Example: a user's user ID)
- **Applications**
  The software that is being managed.
  (Example: Microsoft Office)
- **Application Files**
  The components that make up the application.
  (Example: Microsoft Office has approximately 800 files, DLLs, EXEs, HLPs, ICOs, as well as hundreds of registry updates.)
- **Deployment Source**
  The location where the application components, that will be deployed to users, are stored.
  (Examples: the CM-CSDB or a CM Proxy Server)
- **Deployment Destinations**
  The location to which the application and its files will be deployed on a computer or **LAN** (Local Area Network) device.
  (Example: `C:\MSOFFICE\, C:\WINDOWS\`)

The remaining chapters of this book explain how to use the CM Admin CSDB Editor to configure and maintain desired states for your applications.

As you gain experience with CM and become a more advanced user, you will probably want to include other elements in your desired states. These include distribution scheduling, error handling, security, and collecting audit information from computer devices. Refer to the CM library for descriptions of these concepts and capabilities.

Configuration Management Terminology

The following terms will be explained in detail later, but you should become familiar with these key concepts now.

The CM-CSDB records your desired state and information regarding its deployment. The CM-CSDB is hierarchically structured, and its components consist of **Files**, **Domains**, **Classes**, **Instances**, and **Attributes**. Refer to the Configuration Management publication library for more information.

Table 5 Configuration Management terminology

Term	Short Description
file	A logical partition in the CM-CSDB that groups similar domains. It is the highest level in the hierarchical structure of the database.
Example: PRIMARY File	The PRIMARY File is used to define and maintain the desired state. This is a pre-configured file distributed with CM and is installed when you first install CM.
Domain	A logical partition in the CM-CSDB that groups similar classes. It is the second level in the hierarchical structure of the database.
Example: SYSTEM Domain	The SYSTEM Domain contains the classes needed to deploy applications.
Class	A logical partition in the CM-CSDB that groups similar instances. It is the third level in the hierarchical structure of the database.
Example: USER Class	The USER Class defines users of CM-managed applications. The USER Class defines all the attributes necessary to identify a desktop (agent) computer.
Class Instance	A specific occurrence of a class. Each instance of a particular class inherits the attributes defined for that class.
Example: USER Class instance defining John Doe's computer	This USER instance is an object created from the USER Class, containing the information needed to identify John Doe's CM-managed computer.
Attribute Attribute Value	A property of a class that defines its type. There are four types of classes: expression, variable, connection, and method.

Term	Short Description
Example: USER Class instance defining John Doe's computer, NAME and USERID attributes	The NAME attribute of the USER Class contains the name of the user, and the USERID attribute contains the user ID, as specified by the CM administrator. In this example, the NAME attribute contains the value **John Doe**, and the USERID attribute contains the value **JDOE**.

Handling Your Unique Requirements

HP ships the CM-CSDB with a pre-configured set of components that you use to manage desktop computer configurations on an enterprise-wide scale. Use these components to build and maintain very sophisticated and complex desired states.

However, note that CM provides a highly customizable framework that you can modify and extend to meet your organization's requirements. For example, to extend the basic functionality of CM, you can add your own components to the CM-CSDB, extend the CM-supplied components, and integrate your own in-house developed or third-party-supplied executables.

In particular, this document describes how to use the CM Admin CSDB Editor with the CM-CSDB as it is configured when shipped from HP. If you customize CM, the sample screens shown in this document might differ from what you see in your live environment.

For purposes of maintainability, you should thoroughly document, keep up to date, and store in a central location any changes you make to CM. You might want to create a project folder that is accessible to all of your CM administrators.

Summary

- The CM Admin CSDB Editor is an interactive graphical tool for manipulating and inspecting the content of the CM-CSDB.
- The CM-CSDB records your desired state and information regarding its deployment.
- Your desired state records the identities, relationships, and intended configurations of the computers in your enterprise that are managed by CM.
- Although CM ships with a pre-configured database, the framework is highly customizable.

2 Getting Started

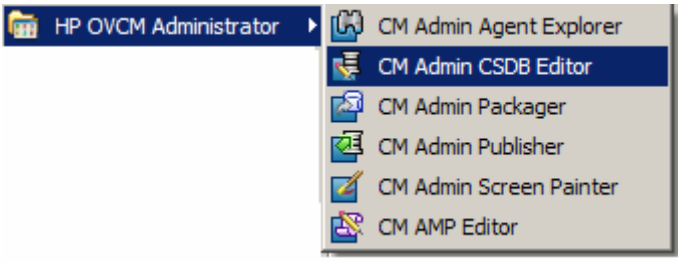
At the end of this chapter, you will:

- Know how to launch the CM Admin CSDB Editor.
- Manipulate the appearance of an open CM Admin CSDB Editor window.
- Know the CM Admin CSDB Editor menus.
- Know how to set your personal preferences.
- Know how to set the defaults for state files that are imported into the HP OpenView Configuration Management Knowledge Base Server (CM KB Server).

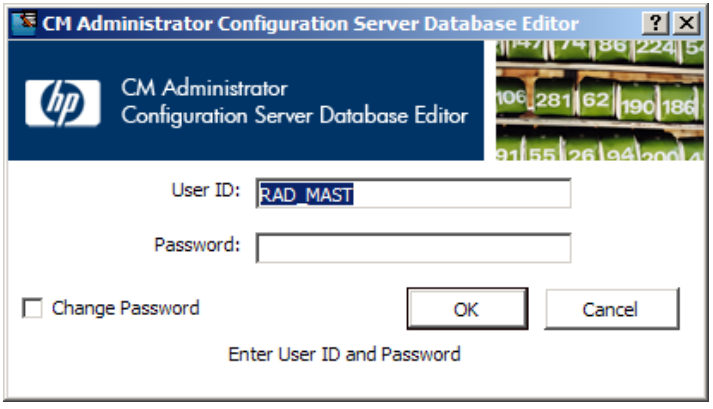
Opening the CM Admin CSDB Editor

To open the CM Admin CSDB Editor

1 Click **Start → Programs → HP OVCM Administrator → CM Admin CSDB Editor**.



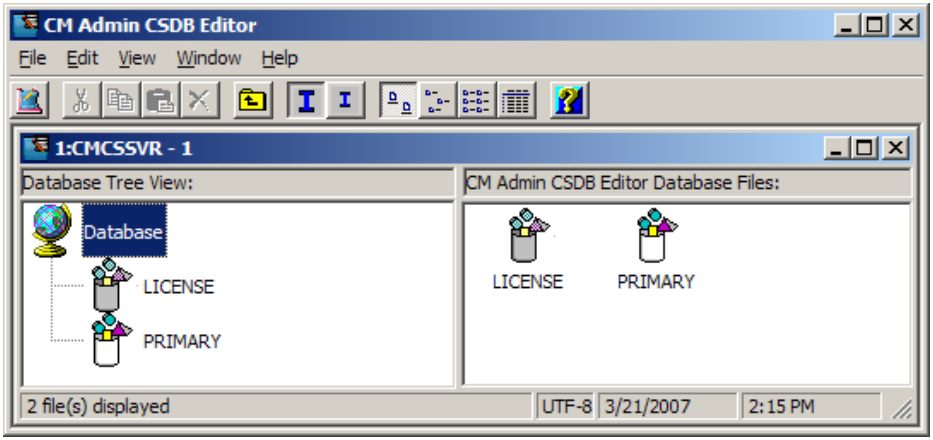
A logon dialog box opens.



The factory set user ID is RAD_MAST. No password is necessary. This might have been changed during installation.

If your administrator has configured the CSDB Editor to require passwords, after a successful logon using the assigned password, you can change a password by selecting the Change Password check box and typing the new password in the New Password and Verify New Password text boxes.

2 Click **OK**. The initial CM Admin CSDB Editor window opens.



The CM Admin CSDB Editor Window

Customizing the CM Admin CSDB Editor Window

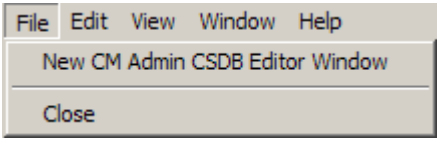
You can customize the look of the CM Admin CSDB Editor and how it functions by using options from the shortcut menus. The most frequently used functions are available from the CM Admin CSDB Editor Toolbar. See The CM Admin CSDB Editor Toolbar on page 29 for additional information about these functions.

CM Admin CSDB Editor Window Menus

The File Menu

Use these options to open additional CM Admin CSDB Editor windows or to close the current one. Multiple windows can be open at the same time. You can drag items between multiple windows.

Figure 1 File menu



The Edit Menu

Use these options to edit the CM-CSDB. The first six choices apply to the selected items in the active pane or dialog box.

Figure 2 Edit menu

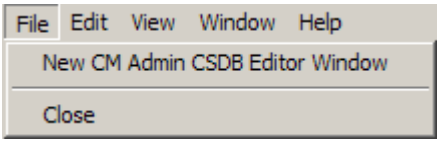


Table 6 Edit Menu options

Operation	Description
New	Creates a new occurrence of the selected item such as domain, class, or instance.
Cut	Removes the selected text and places a copy of it on the Windows clipboard.
Copy	Copies the selected text to the Windows clipboard.
Paste	Copies the current content of the Windows clipboard to the current cursor location.
Rename	Changes the friendly name and/or internal name of the selected item.
Delete	Removes the currently selected items from the CM-CSDB.
Select All	Selects all of the instances in the list view of a CM Admin CSDB Editor window.
Invert Selection	Toggles the selection status of the instances in the list view. Those that are selected are cleared and those that are cleared are selected.
Find	Searches the Name and Instance Name columns in the selected class for instances that contain the text you specify. The results of the search appear highlighted and sorted at the top of the list in the list view.

The View Menu

These options control the appearance of the CM-CSDB. Show or hide the toolbar and status bar, control the size and display of the icons, refresh the window, or set the display and operational options from within this menu.

Figure 3 View menu

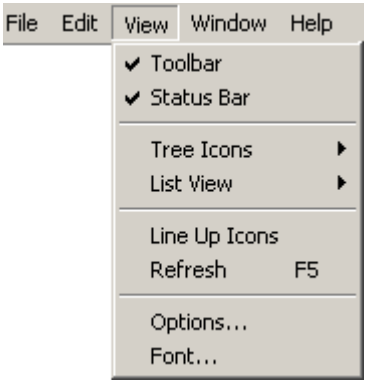


Table 7 View Menu options

Option	Description
Toolbar	Click to toggle the toolbar display on and off.
Status Bar	Click to toggle the Status Bar display on and off.
Tree Icons	Toggles the display between large and small icons.
List View	Choose from among four different displays for your list view (list view)—Large Icons, Small Icons, List, and Details.
Line Up Icons	Redraws the list view arranging the icons either vertically or horizontally.
Refresh	Redraws the list view of the selected CM Admin CSDB Editor window, lining up all of the elements for easy reading.
Options	Opens the CM Admin CSDB Editor Options dialog box. See Setting Preferences: The Options Dialog Box on page 30 for more information.
Font	Opens a dialog box that allows you to change the font used to display text in the CM Admin CSDB Editor windows.

The Window Menu

Use these options when you have more than one CM Admin CSDB Editor window open.

Figure 4 Window menu

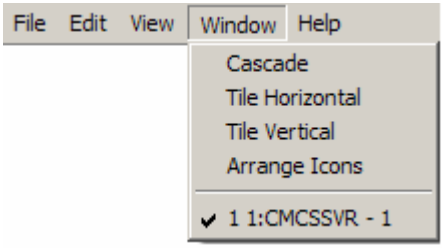


Table 8 Window Menu options

Option	Description
Cascade	Organizes and overlaps the windows starting at the top, left corner of the CM Admin CSDB Editor interface.
Tile Horizontal	Automatically sizes open windows to the width of the CM Admin CSDB Editor user interface. It allocates the available height within the CM Admin CSDB Editor interface equally, stacking the windows one on top of the other.
Tile Vertical	Automatically sizes open windows to the height of the CM Admin CSDB Editor interface. It allocates the available width equally, stacking the windows next to each other.
Arrange Icons	Docks minimized CM Admin CSDB Editor windows in the bottom left corner of the interface.
1 1:*CSname* – 1	Multiple View allows the administrator to toggle between open CM Admin CSDB Editor sessions. A check next to a session indicates which window is currently active.

The Help Menu

Get details about the CM Admin CSDB Editor or link to the HP OpenView web site.

Figure 5 Help menu

Table 9 Help Menu options

Option	Description
Contents	Opens a traditional Windows-style help interface.
Hewlett-Packard on the Web	Opens a sub-menu with links to the Hewlett-Packard web site pages for Technical Support and Home Page. Selecting one of these links opens your default Web browser and connects to the selected page over the Internet.
About CS Database Editor	Displays information about the current CM Admin CSDB Editor—version number, CM Configuration Server information, and so forth. Click **System Info** to display a standard Microsoft System Information window. This application analyzes your machine and shows the results in an easy to navigate format.

The CM Admin CSDB Editor Toolbar

The CM Admin CSDB Editor toolbar provides easy access to frequently used functions:

Figure 6 CM Admin CSDB Editor toolbar

The following table describes the functions of the toolbar buttons.

Table 10 CM Admin CSDB Editor toolbar buttons

Button	Function
	Open a new CM Admin CSDB Editor window.
	Not available.
	Copy the selected item.
	Not available.
	Delete the selected item.
	Display the Database level above.
	Display large icons in the tree view. Provides easy selection from a small list.

Button	Function
	Display small icons in the tree view. Maximizes the number of items that can be seen in the list view of the CM Admin CSDB Editor at one time. Provides easy selection from larger lists.
	Display large icons in the list view. Provides easy selection from a small list.
	Display small icons in the list view. Maximizes the number of items that can be seen in the list view of the CM Admin CSDB Editor window at one time. It is useful for selecting from large lists.
	Display the list view vertically. Maximizes the number of items that can be seen in the list view of the CM Admin CSDB Editor window at one time.
	Display the list view vertically. Adds a column for TYPE (such as Class and Instance). Maximizes the number of items and provides detailed description of the list view.
	Get help. The cursor will change shape to indicate you are seeking help. Click this cursor on a control or screen area for which you are seeking help.

Setting Preferences: The Options Dialog Box

The CM Admin CSDB Editor provides an Options dialog box to allow you to control some of the features of the CM Admin CSDB Editor.

- To open the Options dialog box, click **View → Options.**

  There are five tabs on the Options dialog box:

  — General

  — Instance Options

  — Advanced

  — KB AutoImport

  — Notify

- Click a tab to access its configuration options.

General Tab

Figure 7 CM Admin CSDB Editor Options General tab

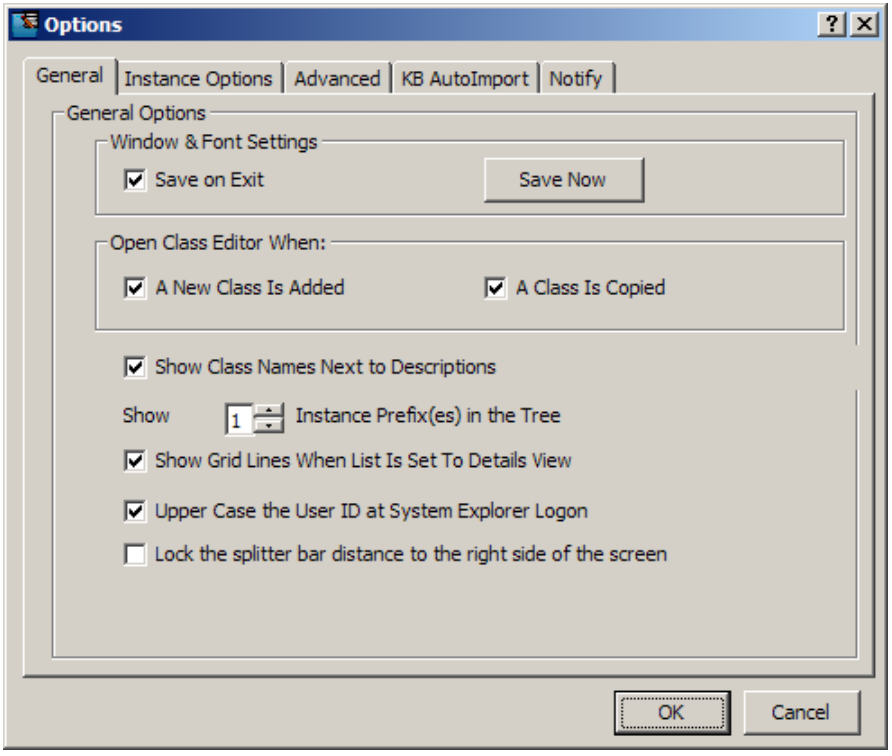


Table 11 describes the General tab options.

Table 11 Functions of the Options General tab

Option	Function and Use
Window & Font Settings	Select **Save on Exit** to track certain window settings and the font to use for displaying text when you exit from the CM Admin CSDB Editor session. The window settings include the view setting (i.e., large icons, small icons, list, or details) for the tree and list views of the CM Admin CSDB Editor window, and the relative width of each pane. Window settings are saved independently for each level of the CM-CSDB. Click **Save Now** to save the current window settings and font in use. To change the font, choose a font from the dialog box that opens when you click **View** → **Fonts** from the menus.
Open Class Editor When	Select the **A New Class is Added** check box to open the Editing Class dialog box when you add a class. Select the **A Class is Copied** check box to open the Editing Class dialog box when you copy a class. Since you will probably want to edit the newly copied or newly added class, HP recommends that you have these two check boxes selected.
Show Class Names Next to Descriptions	Select this check box to show the CM internal name for classes next to their friendly names in the tree view and in the list view. The internal name will appear in parentheses next to the friendly name. This is useful if you usually do not use the Details View for the list view. The Details View in the list view window always shows the internal name for classes in the Type column.
Show *n* Instance Prefix(es) in the Tree	This option can be set to 0, 1, 2, or 3 to control how many levels of instance name prefixes (delimited by the underscore character) will be compressible and expandable in the tree view. This is described in Navigating the CM-CSDB on page 48.
Show Grid Lines When List Is Set To Details View	Select this check box to display grid lines in the list view when its view option is set to Details. Clearing this check box removes the grid lines. Which setting results in a more legible display is a matter of personal preference.
Uppercase the User ID at CM Admin CSDB Editor Logon	Select this check box to automatically convert the user ID into all uppercase.

Option	Function and Use
Lock the splitter bar distance to the right side of the screen	Select this check box to lock the width of the right side of the CM Admin CSDB Editor window during resizing. This is useful if you are focusing on information on the right side of the window. The left side will increase or decrease in size as you resize the window. When this check box is not selected and the window is resized, the width of the left side of the window remains fixed, and the width of the right side of the window varies.

Instance Options Tab

Figure 8 CM Admin CSDB Editor Options Instance Options tab

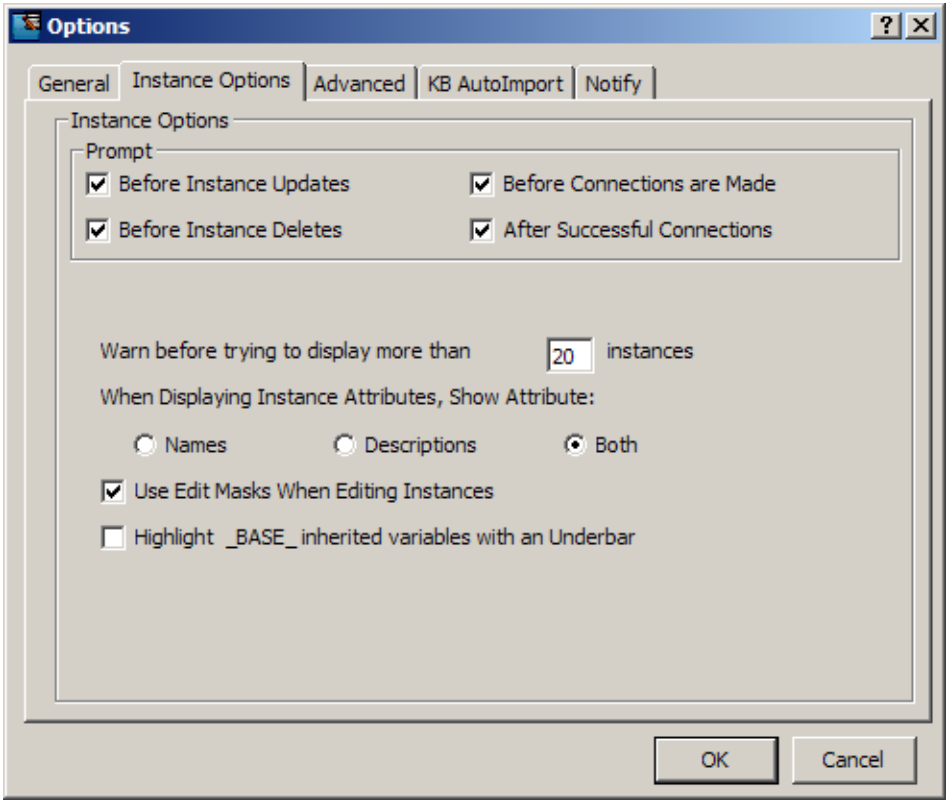
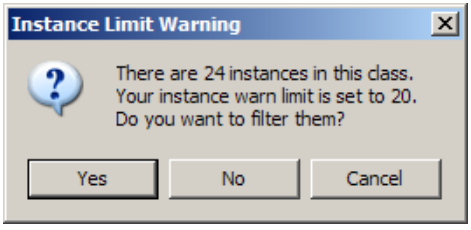


Table 12 describes the **Instance Options** tab options.

Table 12 Functions of the Options Instance Options tab

Option	Function and Use
Prompt	These four check boxes control whether CM Admin CSDB Editor prompts you before completing the actions indicated by their descriptions. The prompt consists of a message asking you to confirm whether you want to complete the action before anything is permanently changed in the CM-CSDB.
Warn before trying to display more than *n* instances	In large desired states, certain classes such as FILE or USER can contain many instances. If you attempt to view the instances of a class that contains more instances than this option setting, a warning similar to the following opens:  This gives you the opportunity to filter the class instances, before they are displayed. This is a performance issue. If you try to display too many instances, you will wait while the CM Admin CSDB Editor retrieves them from the CM-CSDB, and you will have to scroll through a large list to find the instances you want to work with. • Click **Yes** to open the Filter Class dialog box, and enter an appropriate filter. • Click **No** to display all of the instances. • Click **Cancel** to close the warning without displaying any instances.
When Displaying Instance Attributes, Show Attribute	This setting determines which columns to display (Name, Description, or both) in the Details view on the list view of the window, when you open an instance by double-clicking its name in the tree view.

Option	Function and Use
Use Edit Masks When Editing Instances	This setting determines whether the Edit Instance dialog box will be sensitive to the attribute type in the data entry area of the dialog box. If you select this check box, attributes that hold logical (i.e., Yes/No) data have check boxes in the data entry area. Attributes that have multiple choices have drop-down lists in the data entry area. A flag set will appear as a series of check boxes in the data entry area. You can set the mask by entering the valid choices for an attribute within square brackets at the end of the attribute's description in the class definition. If you do not select this check box, all data entry in the Edit Instance dialog box will be plain text. This facility is described in detail in Optional Attribute Editing Controls on page 65.
Highlight _BASE_ inherited variables with an Underscore 	Use this setting if you want to see which values in an instance are inherited from the _BASE_INSTANCE_ of the class. The inherited instances will show with a red underscore.

Advanced Tab

Figure 9 CM Admin CSDB Editor Options Advanced tab

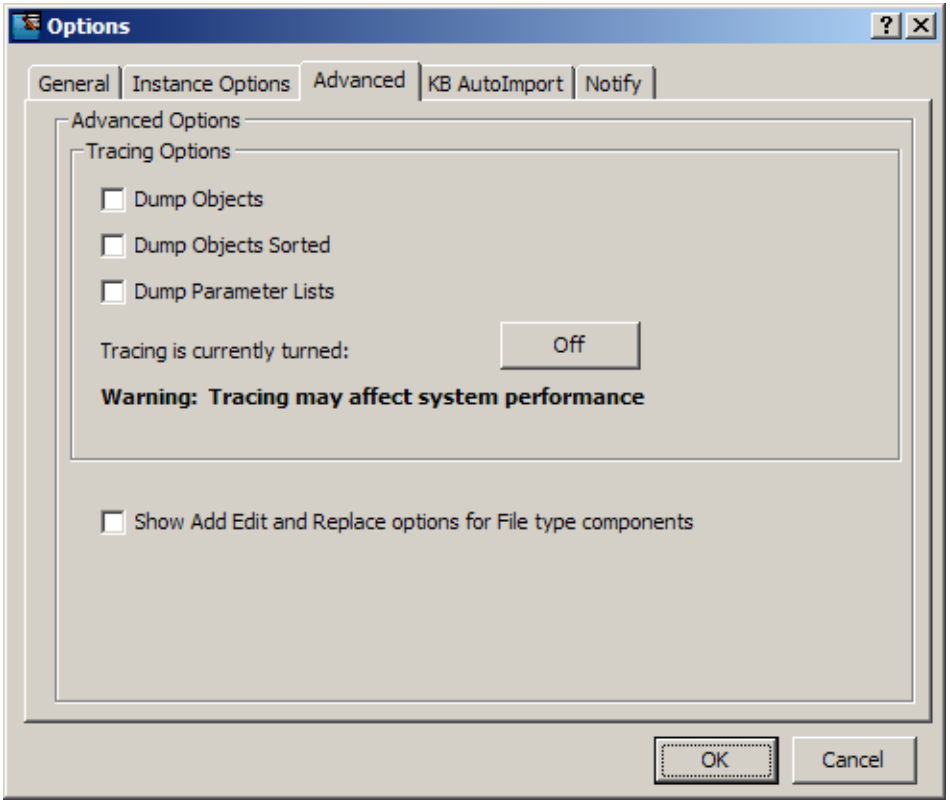


Table 13 describes the Advanced tab options.

Table 13 Functions of the Options Advanced tab

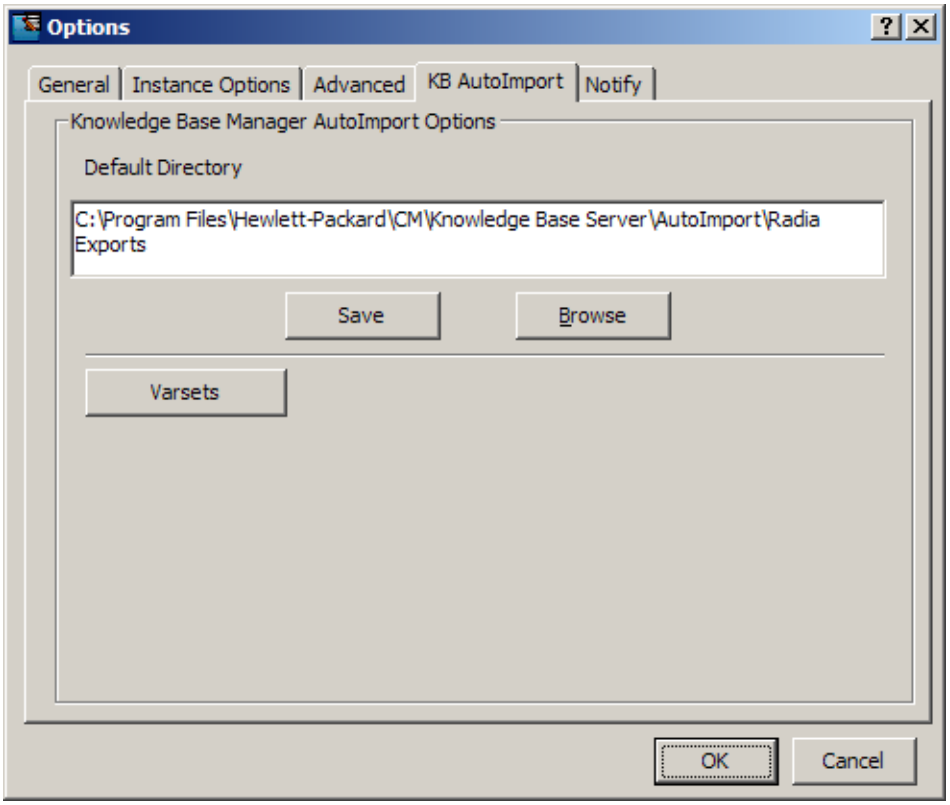
Option	Function and Use
Tracing Options	Use these settings only for troubleshooting the CM Admin CSDB Editor. Do not change these settings unless instructed to do so by HP Technical Support.

Option	Function and Use
Show Add, Edit and Replace options for File type components	WARNING: Use this setting only in non-production environments. This setting determines whether to display the following commands on the shortcut menus for file-type components: • Edit this Component • Replace Component Data • Add a Component For details, see Appendix A, Modifying Component Instances.

KB AutoImport Tab

Use the KB AutoImport tab to define the default auto-import directory for the CM Knowledge Base Server. For details, refer to the *CM KB Server Guide.*

Figure 10 CM Admin CSDB Editor Options KB AutoImport tab



To specify the KB AutoImport options

1 Specify or browse to the default directory to use to import **state files** into the CM Knowledge Base Server database. The default location is an export directory installed by the CM KB Server. This directory includes the required variable set subdirectory: `\VarSets`.

> Any directory that will be used as an export directory must include a subdirectory named `\VarSets`.

2 Click **Save** on the KB AutoImport tab. This ensures that the default directory that was specified for the CM KB Server default auto-import directory is set.

Notify Tab

Use the Notify tab to specify the `radskman.exe` command line values for UID and STARDIR that will be sent to agents when they are notified. (The UID is used to identify the current session, and STARTDIR specifies the IDMLIB starting directory.) For example, select the **Custom UID and or STARTDIR** option to accept or enter values to notify agents by machine name, as opposed to the name of the user who is currently logged on.

The Notify options set here are used by the Notify Manager and displayed on the Notifications Details dialog box as shown in Figure 11 below. For more information, see the topics CM Notify on page 145, Notify Dialog Boxes on page 147, and Set Notification Details on page 151.

Figure 11 CM Admin CSDB Editor Options Notify tab

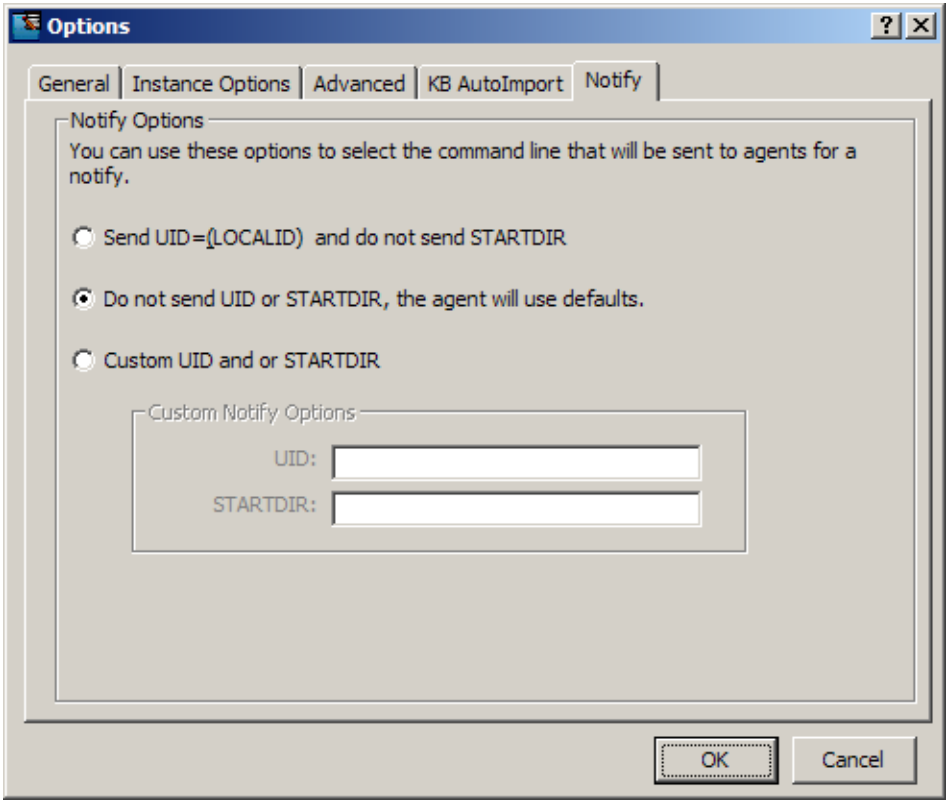


Table 14 on page 40 describes the Notify tab options.

Table 14 Functions of the Options Notify tab

Option	Function and Use
Send UID=[LOCALID] and do not send STARTDIR	Creates a Notify command with UID=[LOCALID]. This command notifies agents by the name of the user who is currently logged on. To notify agents by the name of the machine, use the Custom UID and or STARTDIR option.
Do not send UID or STARDIR, the agent will use defaults.	Creates a Notify command without any values for UID or STARTDIR; agents use the default values for UID and STARTDIR. To specify non-default values, use the Custom UID and or STARTDIR option.
Custom UID and or STARTDIR	Creates a Notify command with the UID and STARTDIR values entered in the Custom Notify Options text boxes. Use this option to notify agents by machine name, or to specify the appropriate starting directory. • Accept the Custom UID value of $MACHINE to notify the agents by machine name. You can also enter another UID value to identify the current session using a custom value. • Accept the Custom STARTDIR value of SYSTEM to notify and install an application in the machine context. You can also enter another STARTDIR value.

Summary

- Use standard Windows operations to change the look of the CM Admin CSDB Editor window and its contents.
- Frequently used functions of the CM Admin CSDB Editor window are available from its Toolbar.
- The CM Admin CSDB Editor provides the administrator with the ability to control some of its features.

3 CM-CSDB Overview

At the end of this chapter, you will:

- Understand the schema of the HP OpenView Configuration Management Configuration Server Database (CM CSDB) and how to navigate it.
- Understand default values for instances and how inheritance works.
- Be familiar with the various HP OpenView Configuration Management Configuration Server Database Editor (CM Admin CSDB Editor) options available through the shortcut menus.
- Know how to work with CM domains, classes, and instances.

The CM Configuration Server Database

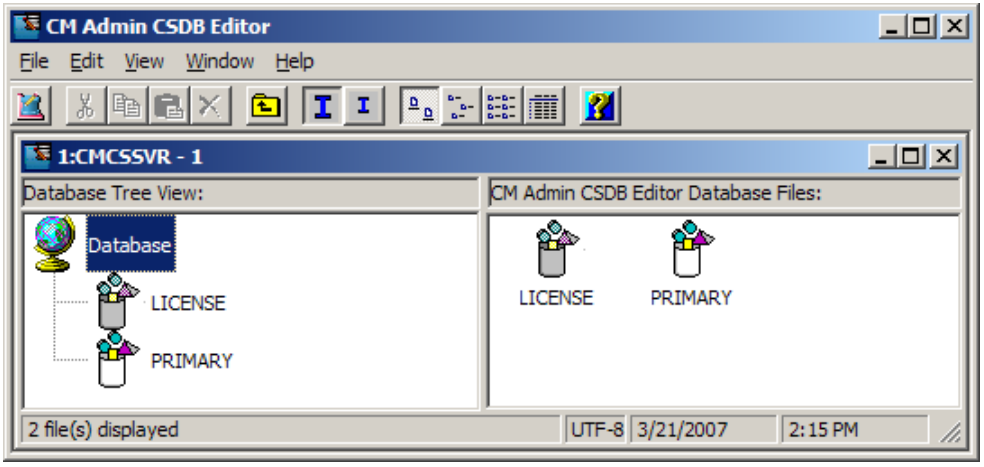
The CM CSDB, located on the HP OpenView Configuration Management Configuration Server (CM Configuration Server), stores your desired state. This includes all of the information that CM uses to manage applications on an agent computer, including:

- The software or data that CM distributes.
- The desired state for each agent computer.
- The policies determining which subscribers are assigned to which packages.
- Security and access rules for CM administrators.

Use the CM Admin CSDB Editor to view and manipulate the CM CSDB.

When you install the CM Configuration Server, LICENSE and PRIMARY are the only two files available. As you use CM, your CM CSDB will change.

Figure 12 CM-CSDB: LICENSE and PRIMARY Files



- The LICENSE File is read-only and used for CM Configuration Server processing. This file is for HP use and should not be modified.
- The PRIMARY File contains most of the information regarding software management. It has eight default domains, as discussed in Table 15 on page 45.

  For more information on the PRIMARY File domains and their classes and attributes, refer to the *HP OpenView Configuration Management*

Configuration Server Database Reference Guide (CM CSDB Reference Guide).

Table 15 PRIMARY File default domains

Domain	Description
ADMIN	Use the ADMIN Domain to limit administrative access to the CM-CSDB, and specify rules for connecting classes.
APPMGMT	Use the APPMGMT Domain to deploy server applications using the Application Management Profiles component of the CM Server Management Agent. Refer to the *HP OpenView Configuration Management Solutions for Servers, Application Management Profiles Guide* for more information.
AUDIT	Use the AUDIT Domain to configure tasks that will inventory agent computers' assets. Refer to the *HP OpenView Configuration Management Inventory Manager Installation and Configuration Guide (CM Inventory Manager Guide)* for additional information.
CLIENT	Use the CLIENT Domain to configure Client Operations Profiles, including setting access points for application data. Refer to the *HP OpenView Configuration Management Application Manager Installation and Configuration Guide (CM Application Manager Guide)* and the *HP OpenView Configuration Management Application Self-service Manager Installation and Configuration Guide (CM Application Self-service Manager Guide)* for additional information.
PATCH	Use the PATCH Domain to store the list of available patches within a PATCH Class. See Service Optimization on page 126 for additional information about this domain.
POLICY	Use the POLICY Domain to organize subscribers into logical groups. Each agent computer managed by CM is represented by an instance of the POLICY Domain USER Class. Refer to the "Implementing Entitlement Policy" chapter in the *CM Application Manager Guide* or the *CM Application Self-service Manager Guide* for more information
PRDMAINT	Use the PRDMAINT Domain to store packages for self-maintenance of CM agents, and any Radia Clients at versions 4.x. Refer to the *CM Application Self-service Manager Guide* for more information on self-maintenance.
SOFTWARE	Use the SOFTWARE Domain to store information about the software being managed and the methods used to deploy the software. Each ZSERVICE instance represents one software application or a single unit of other content to be managed on agent computers.

Domain	Description
SYSTEM	Use the SYSTEM Domain to store administrative and process control definitions.

Table 16 below describes the optional domains of the PRIMARY File. You can select these optional domains during the CM Configuration Server installation, or by installing optional Configuration Management components.

In addition to the optional domains, if you have installed HP OpenView Configuration Management for Servers (CM for Servers) you will encounter the following PRIMARY File domain:

— BASELINE

Refer to the *HP OpenView Configuration Management Solutions for Servers, Configuration Baseline Auditor Guide* for more information on the BASELINE domain, classes and attributes.

Table 16 PRIMARY File optional domains

Domain	Description
MACHINE	Use the MACHINE Domain to define templates for the machines being managed by the HP OpenView Configuration Management OS Manager (CM OS Manager). Refer to the *HP OpenView Configuration Management OS Manager User Guide (CM OS Manager Guide)* for more information.
PATCHMGR	Use the PATCHMGR Domain with the HP OpenView Configuration Management Patch Manager (CM Patch Manager). Refer to the *HP OpenView Configuration Management Patch Manager Installation and Configuration Guide (CM Patch Manager Guide)* for more information.
OS	Use the OS Domain to define the operating system services to be deployed to target machines with the CM OS Manager. Refer to the *CM OS Manager Guide* for more information.
USAGE	Use the USAGE Domain with the HP OpenView Configuration Management Application Usage Manager (CM Application Usage Manager). See the *HP OpenView Configuration Management Application Usage Manager Installation and Configuration Guide (CM Application Usage Manager Guide)* for more information.

- As you begin to use CM, the PROFILE and the NOTIFY Files appear.

— The PROFILE File appears after the first agent computer has registered with the CM Configuration Server. The PROFILE File contains information collected from agent computers. This information is used to connect to computers to deploy software managed by CM as well as to see the configuration of the agent computer.

— The NOTIFY File appears after a notify request has been initiated. The NOTIFY File is divided into domains, where each domain is one Notify operation. Each Notify file domain has one NOTIFY class. Each NOTIFY class contains an instance for each selected member of the Notify operation's audience list. For additional information, see CM Notify on page 145.

Navigating the CM-CSDB

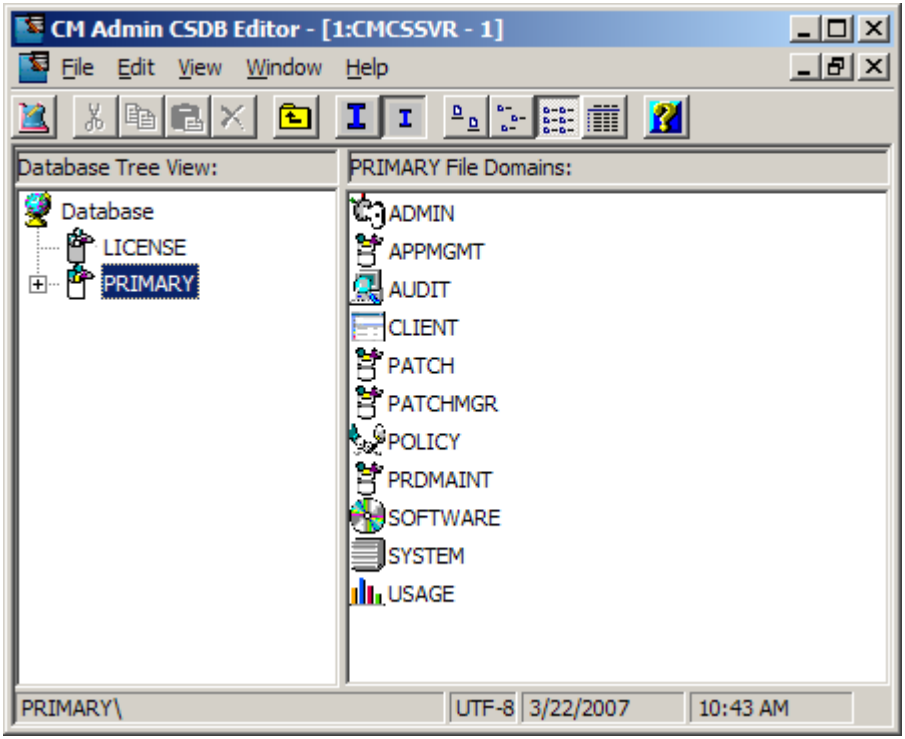
The CM-CSDB is hierarchically organized and subdivided into files, domains, classes, and instances. The tree view enables you to view successively lower levels in the hierarchy, and easily locate and work with specific CM-CSDB components.

To navigate the tree structure

1 Open the CM Admin CSDB Editor, and double-click **PRIMARY** in the tree view.

The PRIMARY file domains in your CM-CS Database will vary according to which CM products have been installed in your environment.

The CM Admin CSDB Editor window will be similar to the following:



When expanded the domains of the PRIMARY File appear as branches beneath its icon in the tree view, and simultaneously appear as individual icons in the list view of the CM Admin CSDB Editor window.

2 To open a CM-CSDB component, double-click its icon.

For example, to display a component from the next level down the hierarchy, double-click the SOFTWARE Domain icon either in the tree view or on the list view of the CM Admin CSDB Editor window. The classes of the SOFTWARE Domain will be displayed in the list view of the CM Admin CSDB Editor window.

If you double-click the SOFTWARE Domain icon in the tree view, the tree view opens to display the SOFTWARE Domain classes below its icon. If you double-click the SOFTWARE Domain icon on the list view of the CM Admin CSDB Editor window, a small box containing a plus sign appears next to the icon in the tree view, but the classes are not displayed beneath it.

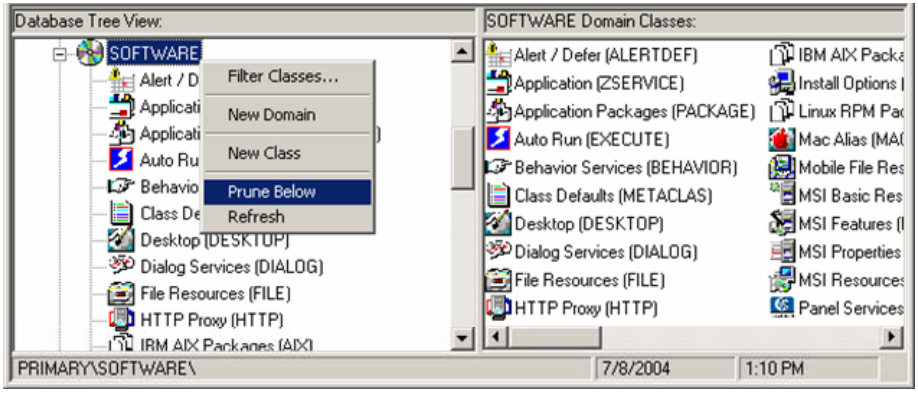
Navigation History

As you open and close components in the CM Admin CSDB Editor, the tree view keeps track of which branches you've opened and closed. It marks previously opened and closed branches at each non-elementary level with a plus or minus sign in a small box next to the component's icon. Clicking the plus sign expands the branch of the tree view to the highest level of detail previously opened in the current CM Admin CSDB Editor session. Clicking the minus sign collapses the tree view below the clicked node. To discard the history of which branches were previously opened beneath a particular component, use the Prune Below function.

To use the Prune Below function

1 Right-click the SOFTWARE Domain icon in the tree view.

A shortcut menu opens.



2 Click **Prune Below**.

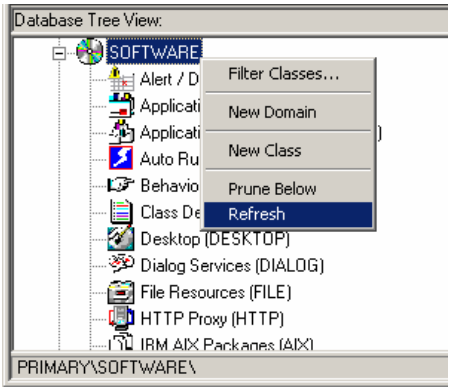
The tree view collapses all open nodes below the SOFTWARE Domain, and discards the history of their having been open.

To rebuild the display of a node and its branches in the tree view

From the current data in the CM-CSDB, use the Refresh function.

1 Right-click the SOFTWARE Domain icon in the tree view.

A shortcut menu opens.



2 Click **Refresh**.

3 The tree view collapses all open nodes below the SOFTWARE Domain, retrieves all data for the SOFTWARE Domain from the CM-CSDB, and re-expands the tree view of the SOFTWARE Domain to show the classes it contains. The display now includes all changes made to the CM-CSDB during the current session.

When you refresh a node in the tree view, the data for that node (and its sub-nodes) are retrieved from the CM-CSDB, and the refreshed node is expanded in the tree view.

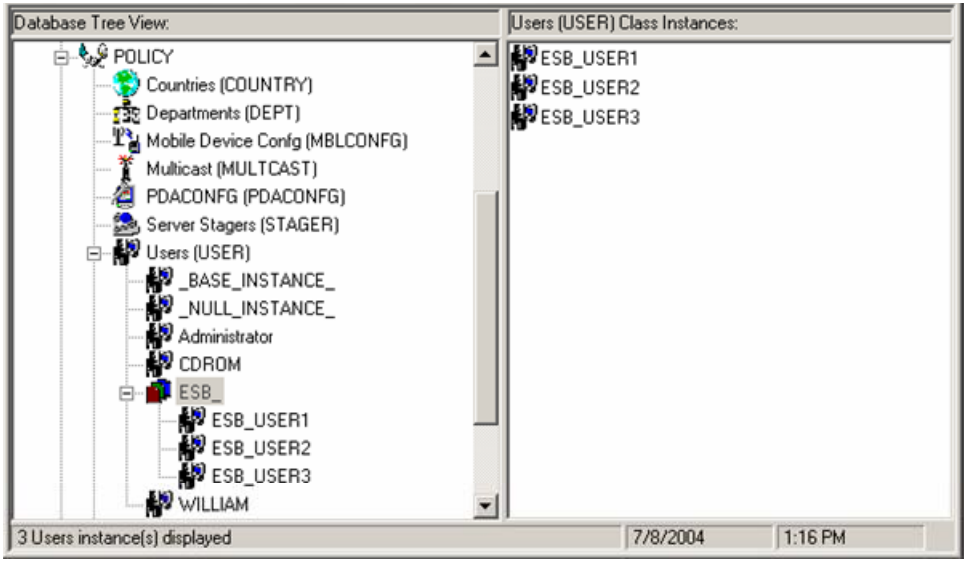
Contrast this with the Prune Below operation and expanding/collapsing sub-nodes by clicking the plus or minus sign next to a node. Neither of these retrieves data from the CM-CSDB; they simply change the view of data already retrieved.

Compound Names

Use compound names to make navigation in the tree view more manageable. CM provides one, two, or three collapsible levels, based upon one, two, or three high-level qualifiers of the instance name.

A compound name has one, two, or three prefixes, each separated from the rest of the name by an underscore character. For example, you might want to group your users according to the building where they are located. Instance names for these users will have a building identifier prefix. If you had users in the Empire State Building, you might assign ESB_ as the instance name prefix to identify them. Each user in the Empire State Building will be assigned an instance name beginning with ESB_. The tree view will automatically add a level.

Figure 13 Compound names



The number of prefix levels in an instance name (0, 1, 2, 3) that are displayed in the tree view is controlled by the Show n Instance Prefixes in the Tree option setting. This is found on the General tab of the Options dialog box and can be accessed from the View menu. The Options dialog box is described in Setting Preferences: The Options Dialog Box starting on page 30.

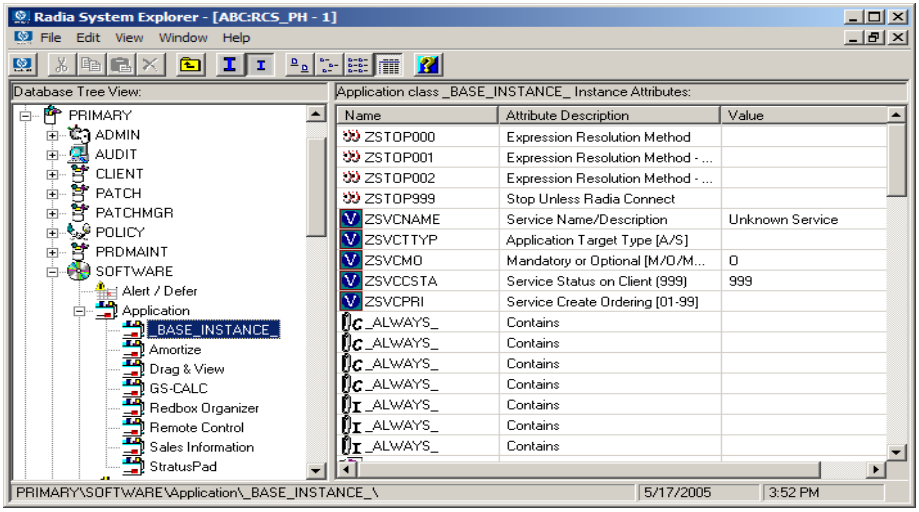
Each underscore in an instance name delineates a level for the tree view. Thus, the instance name ESB_FLOOR10_USER1 has two levels of prefix, which, if the setting Show n Instance Prefixes in the Tree were set to 2 or 3,

would be displayed in two collapsible non-elementary levels (FLOOR10, within ESB).

Inheritance in the CM-CSDB

Each class in the CM-CSDB has a *base instance*, named _BASE_INSTANCE_. It provides the default values for the other instances of the class. Any attribute in an instance will inherit its value from the same attribute in the _BASE_INSTANCE_ unless it has been assigned a specific value.

Figure 14 _BASE_INSTANCE_ of Application (ZSERVICE) Class



Inherited values appear in the instance no differently from values that you enter directly into the instance. You can tell if a variable's value is inherited from the _BASE_INSTANCE_ by inspecting the _BASE_INSTANCE_. Also, if you delete a value from a variable (see Working with Classes on page 53 to learn how), and the value re-appears in the instance automatically, the value is being inherited from the _BASE_INSTANCE_.

CM Admin CSDB Editor Shortcut Menus

In the CM Admin CSDB Editor you can add, copy, edit, rename, and delete CM-CSDB components (domains, classes, and instances). It also provides extensive support for connecting instances.

You can access these functions from shortcut menus that open when you right-click targets in the CM Admin CSDB Editor window. The menu that opens contains choices depending upon which target you right-click.

Working with Domains

Adding a Domain

To add a domain to a file

1. Right-click the PRIMARY File or any domain, and select **New Domain** from the shortcut menu that opens.

   The Create Domain dialog box opens.

2. Enter a name for the new domain and click **OK**.

   Domain names can be a maximum of eight characters including letters and numbers.

Working with Classes

Adding a Class

New classes can be added by

- copying an existing class,

  or

- creating a new class

Which to choose depends upon whether a class exists whose attributes are substantially similar to the class you want to add. If so, copy the existing class; otherwise, create a new class.

Copying a class creates a new class with the same attributes as the copied class, and a _BASE_INSTANCE_ for the new class with its attribute values set as in the _BASE_INSTANCE_ of the copied class.

Copying a Class

To copy an existing class

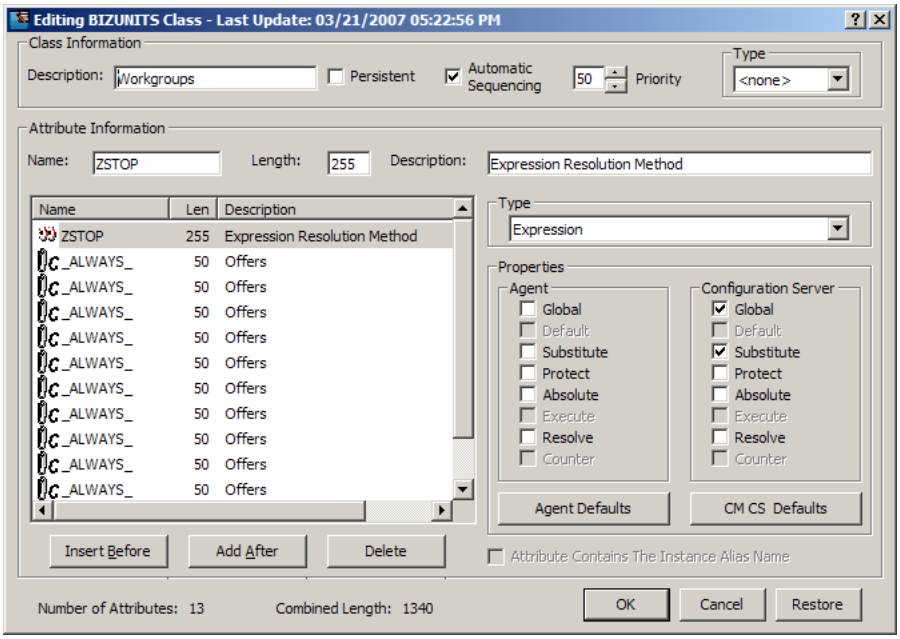
1 Highlight the class to be copied and right-click it.

2 Select **Copy Class** from the shortcut menu.

The Copy class dialog box opens.

3 Enter a name for the new class (BIZUNITS in this example), and click **OK**.

- If the option to open the Class Editor when a class is copied is not in effect (see Setting Preferences: The Options Dialog Box on page 30 for details), the original class is copied with the new name, the Class Editor is not opened, and no warning (see the Class Edit Confirmation warning message on page 55) is issued. You will have to manually open the Class Editor to edit the new class.
- If the option to open the Class Editor is in effect, the Editing Class dialog box opens. See Editing a Class on page 57.



4 To complete the addition of the copied class, change the Description field to identify the copied class, edit the class to reflect the new class' requirements, and click **OK**.

The Class Edit Confirmation warning message opens. It warns you to back up your database before proceeding.

5 Click **Yes**.

The new class addition is complete.

If you click **No** or **Cancel** in the Class Edit Confirmation warning message the Class Editing dialog box closes, but the copying of the class does not terminate.

If you have created a class and want to delete it, see Deleting a Class on page 69.

After you have created a class, check the MASTER_CONNECTION_RULES variable to determine if you need to establish a drag-and-drop rule for the new class. For additional information regarding connection rules and drag-and-drop connections, see Maintaining Connection Rules in the CM-CSDB on page 181.

Create a New Class

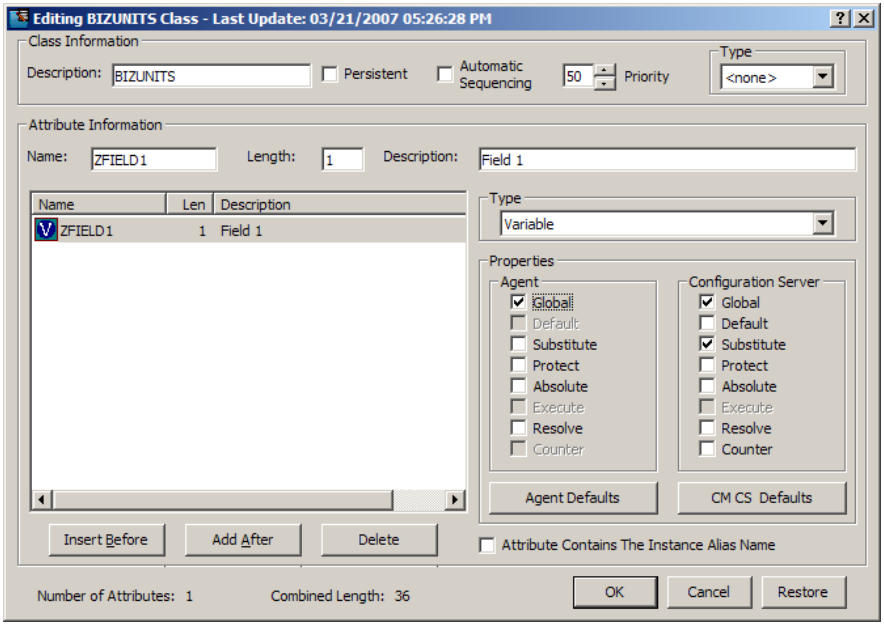
To create a new class

1. Highlight the domain where you wish to insert the new class.
2. Right-click the domain name.
3. Select **New Class** from the shortcut menu.

   The Create Class dialog box opens.
4. Enter a name for the new class (BIZUNITS in this example), and click **OK**.

   — If the option to open the Class Editor when a class is copied is not in effect (see Setting Preferences: The Options Dialog Box on page 30 for details), the original class is copied with the new name, the Class Editor is not opened, and no warning (see the Class Edit Confirmation warning message on page 55) is issued. You will have to manually open the Class Editor to edit the new class.

   — If the option to open the Class Editor is in effect, the Editing Class dialog box opens. See Editing a Class on page 57.



5. To complete the addition of the copied class, change the Description field to identify the copied class, edit the class to reflect the new class' requirements, and click **OK**.

The Class Edit Confirmation warning message opens. It warns you to back up your database before proceeding.

6 Click **Yes**.

The new class has been added.

If you click **No** or **Cancel** in the Class Edit Confirmation warning message the Class Editing dialog box closes, but the copying of the class does not terminate.

If you have created a class and want to delete it, see Deleting a Class on page 69.

After you have created a class, check the MASTER_CONNECTION_RULES variable to determine if you need to establish a drag-and-drop rule for the new class. For additional information regarding connection rules and drag-and-drop connections, see Maintaining Connection Rules in the CM-CSDB on page 181.

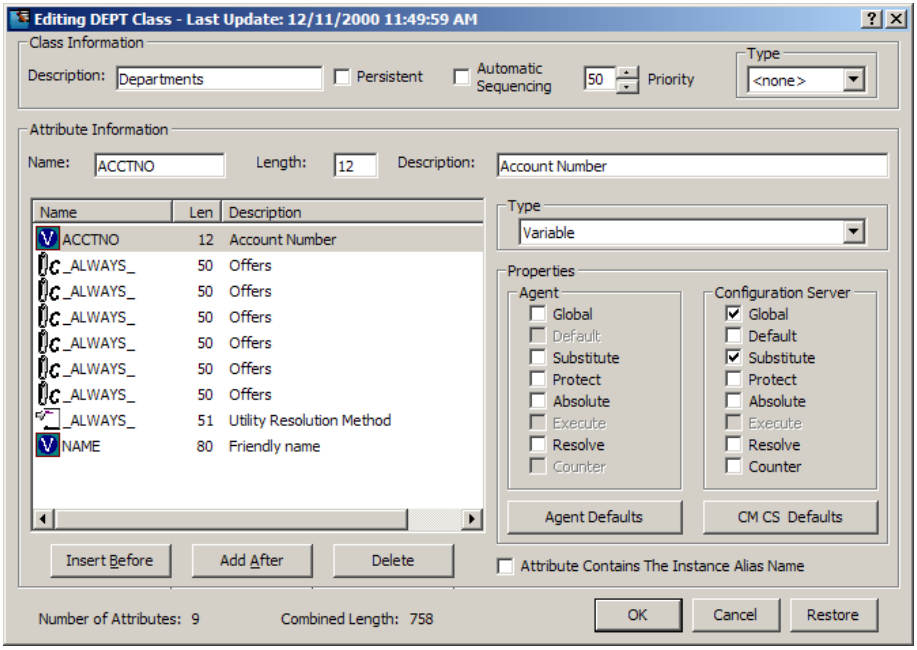
Editing a Class

When you choose **Edit Class** from a shortcut menu, or during the addition of a new class as described in Adding a Class on page 53, the Editing Class dialog box opens.

To edit a class

1 Right-click the **Departments** class in the tree view, and choose **Edit Class** from the shortcut menu.

The Editing Class dialog box opens.



2 Set the Class Information properties at the top of the dialog box as described in Table 17 below.

Table 17 Class Information properties

Property	Significance and Setting
Description	The friendly name that describes the class and its purpose. This name appears next to the class icon in the tree view.
Persistent	This property is available to CM administrators who have a thorough understanding of the CM resolution process. A persistent class is one whose attributes and objects remain in global memory for the duration of the resolution process. The instances created from persistent classes are available to the CM agent for storage on its desktop. If a class is not persistent, it is transient. A transient object exists only while it is being processed during resolution. Instances created from transient classes are not available to the CM agent for storage on the desktop. For additional information regarding persistent and transient objects and how they are incorporated into the resolution process, see Chapter 5, Understanding the Resolution Process.

Property	Significance and Setting
Automatic sequencing	Normally, resolution proceeds from attribute to attribute in the order the attributes appear in the Attribute List. See Chapter 5, Understanding the Resolution Process. Selecting this check box imposes a grouping on the order in which attributes of the class are processed during resolution: • Expressions • Variables • Classes (connections) • Methods Groups are processed in the order in which they appear above, and within these groupings, the attributes are processed in the order in which they appear in the Attribute List. Unless you need to alter the normal sequence of processing attributes during resolution, this check box should be cleared.
Priority	This field is two characters long and has a value between 00 and 99. During object resolution, the CM Configuration Server puts this value in the ZOPJPRI variable. If there is a blank or zero value in this field, the CM Configuration Server supplies the default value specified for that particular class. The default values for the classes are: PATH 5 REGISTRY 60 DESKTOP 70 FILE 50 METACLAS 10 ZSERVICE 50 MACALIAS 70 All others 50
Type	Classifies classes according to their purpose and use. CM uses this classification to affect the resolution process. For example, resolution to populate the CM Service List on an agent computer does not process classes classified as Component classes. The following values can be selected from the drop-down list: <none> – This class is not classified.

Property	Significance and Setting
	Configuration – This class identifies content to be managed by CM on agent computers on an aggregate level. Typically, instances of this class do not have data that can be distributed. They are connected to instances of one or more Component classes. Examples: ZSERVICE, PACKAGE, VGROUP, VERSION, and so forth. **Component** – This class identifies the items (files, registry entries, links, icons, etc.) that comprise the content identified by a Configuration class instance. Typically, instances of this class have data that can be distributed. Examples: FILE, REGISTRY, and DESKTOP. **Policy** – This class identifies one, or a group of agent computers. Typically, instances of Policy classes are connected either to an instance of another Policy class, or instances of Configuration classes.

3 To modify the definition of an attribute, click its name in the Attribute List to highlight it. The name, length, and description of the highlighted attribute appear in the Attribute Information text boxes. The Type drop-down list indicates the type of the highlighted attribute.

4 Type your desired changes into the Attribute Information data entry fields, and click on the desired Type option to set the attribute's type. Class attributes can be one of four types as described in Table 18.

Table 18 Class Attribute types and their functions

Attribute Type	Usage
Expression 	Contains a single line REXX command that is executed during resolution. In an attribute named ZSTOP, causes resolution of the current instance to terminate, if the expression evaluates to “true.”
Variable 	A piece of named storage containing a variable value. The variable’s value forms a part of the agent’s resolved desired state, and can influence the resolution process through messaging or symbolic substitution.

Attribute Type	Usage
Connection (available)  Connection (set)  Note: The letter in the icon denotes the connection type: • A – Always • C – Class • I – Includes • R – Requires	Class connections determine the path of resolution of an agent's desired state during the agent connection process. Available connections are attributes into which a connection to another class can be set. Set connections are attributes containing a specification for a connection to another class. A class connection is, in effect, a branch in the resolution process. See Chapter 5, Understanding the Resolution Process. • An Always connection is always resolved, no matter what the name of its attribute. • A Class connection is resolved if its attribute's name is _ALWAYS_, or if the name of its attribute matches the current value of the system message. • An Includes connection is always resolved, no matter what the name of its attribute. • A Requires connection blocks resolution of instances in the required instance and its children, whose instance names are identical to instance names in the requiring instance and its children. Requires and Includes connection types are useful, among other uses, for expressing inter-package relationships, connecting instances of the PACKAGE Class. When one package Requires another, components of the required package are superseded by the same component in the requiring package. When one package Includes another, both are deployed as a single entity.
Method 	Methods are programs executed as part of the resolution process. The method attribute identifies the program to be executed.

As you make changes, they are immediately reflected in the Attribute List.

5 The **Properties** check boxes are either enabled or disabled depending upon the attribute type.

Agent Properties

All CM agent properties, except for Protect and Execute, are reserved for future enhancements, and you should leave them set to their default values.

- The CM Agent Protect property, when selected, indicates that the value of an attribute stored in an object on the agent computer will be encrypted.
- The CM Agent Execute property, which applies only to method attributes, indicates, when selected, that the method to be executed will be executed on the agent computer.

Configuration Server Properties

The following table summarizes the significance of the Configuration Server properties.

Table 19 Configuration Server Properties and Their Functions

Property	Significance and Setting
Global	When selected for a variable in a transient class, indicates that during resolution the variable will be allowed to migrate (flow) from the child object to a parent object, prior to the transient object being dissolved.
Default	Used internally by CM and is not available to CM administrators. When selected, prevents default values in transient-class objects from being incorporated into parent, persistent-class objects.

Property	Significance and Setting
Substitute	When selected, indicates that symbolic substitution will occur for symbols (references to other variables identified by an initial ampersand) in the value of the variable. Clearing this check box treats the value of the variable as literal text, preventing symbolic substitution from occurring. The Substitute property determines whether symbolic substitution occurs when a variable is created as a result of a CM Configuration Server GET operation, such as occurs when a REXX program calls the EDMGET function. If a variable in the object retrieved by EDMGET contains a symbol, and the Substitute check box was selected for that variable when the object was created, substitution occurs as the variable is created in storage. For example, the expression &(ZMASTER.ZUSERID) will evaluate to WILLIAM after symbolic substitution, if the value of the ZUSERID attribute of the ZMASTER object is WILLIAM at the time symbolic substitution occurs.
Protect	When selected, indicates that the value of the attribute is to be encrypted during resolution. The attribute value exists in encrypted form in memory during resolution, but remains as plain text in the CM-CSDB.
Absolute	When selected for a variable in a persistent class, prevents identically named variables in child objects (instantiated from a transient class) from overlaying the values for those variables in the parent object.
Execute	When selected, indicates that the value of the attribute identifies a method to be executed by the CM Configuration Server.
Resolve	When selected, indicates that symbolic substitution will occur for symbols (references to other variables identified by an initial ampersand) in the value of the variable. Clearing this property's check box causes the value of the variable to be treated as literal text, preventing symbolic substitution from occurring. The Resolve property determines whether symbolic substitution occurs when a variable is PUT into an object, such as occurs when a class instance is retrieved from the CM-CSDB and instantiated as an object in storage.

Property	Significance and Setting
Counter	When selected, this check box indicates that the variable contains an integer that the CM Configuration Server will accumulate. The accumulated total will be stored in a variable with the same name in all parent persistent objects. For example, this property can be used for such tasks as calculating the total uncompressed size of all resources distributed for a service, and storing the result in the ZSERVICE object. Select this check box for the ZRSCSIZE variable in the FILE Class, and a ZRSCSIZE variable will be added to the ZSERVICE object, containing the sum of the ZRSCSIZE values of all FILE instances in the service. The total size of all services' resources for the user will automatically be stored in a ZRSCSIZE variable in the user's ZMASTER object. The ZRSCSIZE variables in the ZSERVICE and ZMASTER objects are created and maintained automatically by the CM Configuration Server when the Counter check box is selected for the ZRSCSIZE variable in the FILE Class template. Any variable containing an integer can be accumulated and stored in parent persistent objects by selecting the Counter check box in the variable's definition in the class template. For example, select this check box on the ZCMPSIZE variable of the FILE Class to calculate the total compressed size of resources distributed for a service.

You can reset the property settings for the selected (i.e., highlighted) attribute to their default values by clicking the default buttons at the bottom of the Properties group box.

Adding an Attribute to a Class

1. To add a new attribute, first determine where in the Attribute List the attribute should be inserted. Unless the Automatic Sequencing check box is selected, as described above, attributes are processed during resolution in the order in which they appear in the Attribute List.
2. Click an attribute next to the insertion location, to highlight it. Then click **Insert Before** to insert the new attribute above the highlighted attribute, or click **Add After** to insert the new attribute after the highlighted attribute. A blank attribute (a variable attribute by default) is inserted at the indicated location.
3. Enter a name, length, and description in the Attribute Information data entry fields, and then select the Type option from the drop-down list to indicate the desired attribute type. Attribute names can be up to eight characters long.

Attribute names can have significance in the resolution process. Values of variables in child objects instantiated from within transient classes can replace the values of identically named variables in parent objects represented within persistent classes. Methods can be conditionally executed or skipped, and class connections can be conditionally followed or ignored based upon the name of the method or class connection attribute. If the attribute's name is _ALWAYS_, the method will be executed or the class connection will be followed unconditionally. Otherwise, the method will be executed, or a class connection will be followed only if the name of the attribute is identical to the current value of the system message. See Chapter 5, Understanding the Resolution Process.

Specify the length of the attribute, in characters. Enter a number greater than or equal to the maximum number of characters for a value of the attribute. For connection and method attributes, a length of 50 is normally adequate. The length needed for a variable attribute depends on the data it is intended to hold.

The description you enter will appear in the Attribute Description column of the Editing Instance dialog box when editing instances of this class. The clearer the description you enter now, the easier it will be for those who edit instances of this class later. For variable attributes, enter a description that clearly identifies the data to be held in the variable attribute. For class connections, you can enter Connect to, or a description that is more meaningful for your purposes. For methods and expressions, try to indicate the purpose of the method or expression in the description.

4 To set the value of an attribute as the display name in the CM Admin CSDB Editor, select the attribute, and then select the **Attribute Contains the Instance Alias Name** check box. This is usually set on the Name (friendly name) attribute. Only one attribute should have this turned on per instance.

Optional Attribute Editing Controls

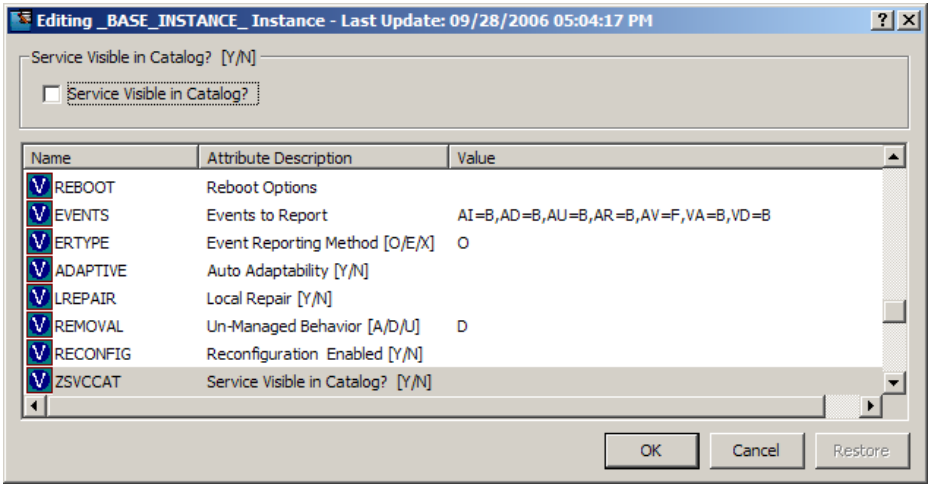
The Attribute Description for variable attributes can be used to determine the control that the CM Admin CSDB Editor displays for entering or editing that variable attribute in the Editing Instance dialog box. Normally, variable attributes are entered or edited as free text in a text box. You can change this to a check box for a flag (a yes/no field), a set of check boxes for a flag set (a field consisting of multiple single-character codes), or a drop-down list for variables from which the user must choose a value from a small list of predetermined values.

To enable this feature, the Use Edit Masks When Editing Instances check box must be selected on the Instance Options tab of the Options dialog box. See Table 12 on page 34 for details.

Flag Attribute

A flag attribute is a one-character field whose value can be Y or N. To direct the CM Admin CSDB Editor to offer a check box for entering or editing flag attributes in instances, the Attribute Description for the attribute in the class template must terminate with [Y/N], as in this example:

Figure 15 ZSVCCAT variable contains [Y/N] for flag attribute



The ZSVCCAT attribute is one character in length, and its Attribute Description, Service Visible in Catalog? [Y/N], ends with [Y/N]. Thus, to enter or edit this variable, the CM Admin CSDB Editor offers a check box.

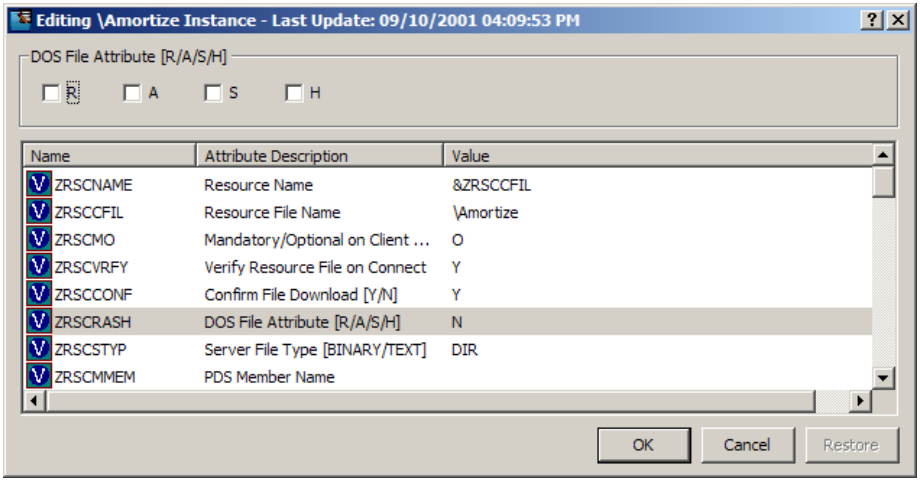
Flag Set Attribute

A flag set attribute is a two- or more character field that holds a value composed of zero or more single-character flags. If a flag is on, the character value that is associated with the flag appears in the flag set string. If a flag is off, the character value that is associated with the flag does not appear in the flag set string. An empty flag set string indicates that no flags are on.

For example, the DOS File Attribute flag set for a particular file is contained in the ZRSCRASH attribute of the FILE Class. This four-character attribute holds a string indicating which of the DOS File Attributes (R=Read-only, A=Archive, S=System, H=Hidden) should be turned on for the file associated

with the FILE instance when that file is deployed to a user desktop. When editing this attribute in an instance, it appears as follows:

Figure 16 An attribute ending with a set of one-character choices

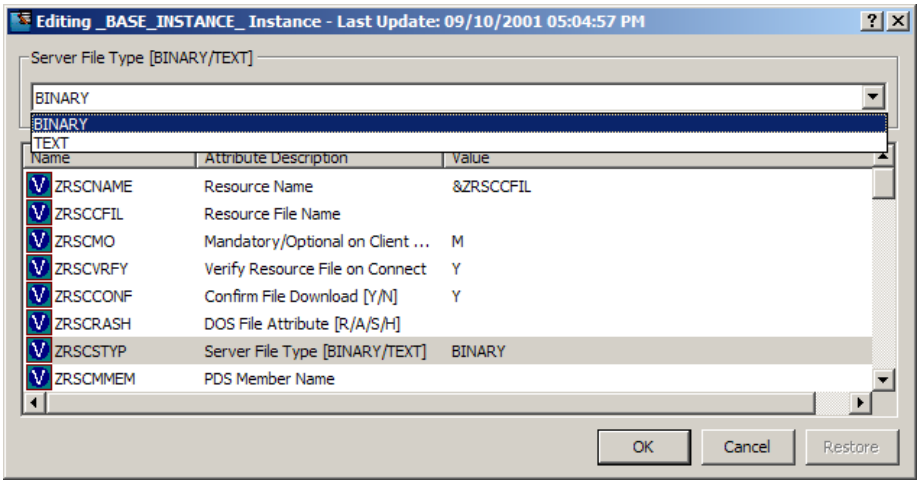


If the Attribute Description for a multi-character attribute terminates with one-character choices separated by slashes and enclosed within square brackets, and the number of choices equals the length of the attribute (as in the ZRSCRASH attribute, above), the CM Admin CSDB Editor treats the attribute as a flag set. It presents a check box for each potential value in the flag set in the Editing Instance dialog box. If a check box is selected, its associated value is inserted in the attribute.

Drop-Down List

The CM Admin CSDB Editor offers a drop-down list for entering or editing attributes whose Attribute Description in the class template terminates with a series of choices separated by slashes and enclosed in square brackets.

For example, the ZRSCSTYP attribute of the FILE Class offers two choices in the Attribute Description (as shown in the following figure) after clicking on the drop-down list control. An attribute description that terminates with a series of choices separated by slashes and enclosed in brackets is offered a drop-down list for editing attributes.



1 As you make changes to a class, you might change your mind and wish to start over. After a change has been made during class editing, the Restore button is enabled. Click this button to discard changes you have made to the class since the Editing Class dialog box opened.

   The Confirm Restore message appears.

   — Click **Yes** to discard changes and restart editing the class.

   — Click **No** to dismiss the confirmation without discarding your changes.

2 When you've completed editing the class, click **OK** to save your changes and exit the Editing Class dialog box.

3 If there are blank attribute names or descriptions in the class, or if an attribute's length is zero, an alert message opens.

4 If there are no blank attribute names or descriptions in the class, and no attribute's length is zero, the Class Edit Confirmation message opens. It warns you to back up your database before proceeding.

5 Click **Yes** to update all existing instances of the class.

   Click **No** to discard your changes to the class, and avoid updating the instances.

   Click **Cancel** to return to the Editing Class dialog box.

Deleting a Class

To delete a class and all of its instances

1. Right-click on the class in the CM Admin CSDB Editor tree view.
2. Select **Delete Class** from the shortcut menu that opens.

   A Confirm Delete dialog box opens.
3. Click **OK** to delete the class and all of its instances.

   Click **Cancel** to close the message without deleting anything.

Working with Instances

Much of the work of CM administrators consists of adding, editing, and deleting instances of various classes, and setting or removing connections between them.

Naming Instances

Use these guidelines when naming instances in the CM-CS Database:

- Use characters that can be displayed and printed.
- Do not include a period. Periods are reserved characters used to delimit the File, Domain, Class and Instance names (F.D.C.I) of a fully-qualified instance in the CM-CS Database.
- Do not include special characters that have a specific meaning to the file system or command shell of any OS platform, such as these:
  \ | / : ; < > * ? ` " ' , ( ) &
- Do not include sequential underscores. This reserved sequence indicates a Group Resource in the CM-CS Database.
- Do not include embedded spaces or control characters, such as blanks, CR (Carriage Return), LR (Line Return), or tabs.

Friendly Instance Names

Friendly instance names do not have the same restrictions as instance names. Use characters that can be displayed and printed. Spaces and other

special characters are allowed. To avoid confusion, HP recommends against using periods in friendly names.

Adding an Instance

New instances can be added by copying an existing instance, or adding a new instance from scratch. Which to choose depends upon whether an instance exists whose attribute values are substantially similar to the instance you want to add. If so, copy the exiting instance; otherwise, add a new instance from scratch.

Copying an Instance

Copying an instance creates a new instance with the same attribute values as the copied instance.

To copy an existing instance

1 Choose **Copy Instance** from a shortcut menu.

A Copy Instance dialog box opens.

2 Enter a display name (i.e., a friendly name) and an instance name for the new instance, and click **OK**.

The instance will be created and will appear in both the tree view and in the list view of the CM Admin CSDB Editor window.

The friendly name appears in the tree view, and the instance (i.e., internal) name appears in the title bar of the list view of the CM Admin CSDB Editor window, when the instance is selected and opened in the tree view.

Adding an Instance

Creating an instance results in a new instance with attribute values set as in the _BASE_INSTANCE_ of the class.

To add a new instance

1 Choose **New Instance** from a shortcut menu.

A Create Instance dialog box opens.

2 Type a display name (i.e., a friendly name) and an instance name, and click **OK**.

The instance will be created and will appear in both the tree view and in the list view of the CM Admin CSDB Editor window.

The friendly name appears in the tree view, and the instance (i.e., internal) name appears in the title bar of the list view of the CM Admin CSDB Editor window.

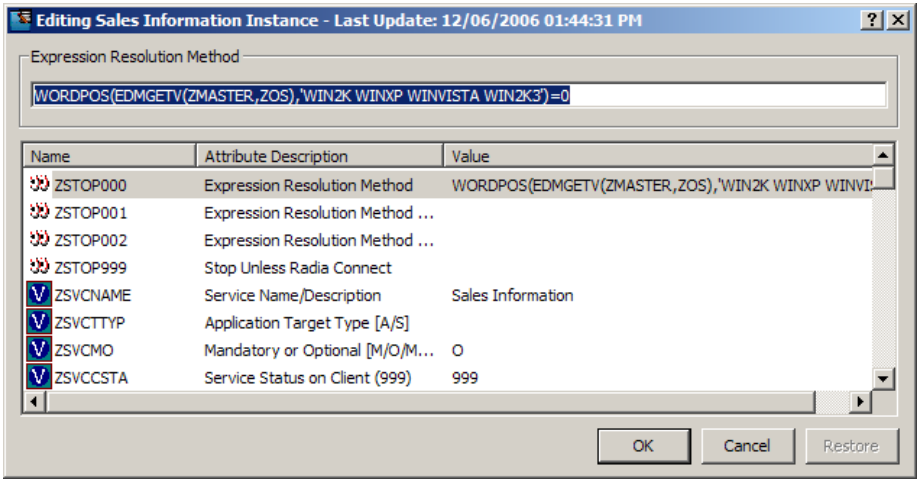
You can now edit the new instance to contain attribute values intended for the new instance.

Editing an Instance

To edit an instance

1 Choose **Edit Instance** from a shortcut menu, or double-click an attribute's name in the list view of the CM Admin CSDB Editor window.

The Editing Instance dialog box opens.



If you chose Edit Instance from a shortcut menu, the dialog box opens with the first attribute in the instance opened for editing. If you double-click an attribute name to open the dialog box, the attribute whose name you double-clicked is open for editing.

The dialog box lists the attributes comprising the instance, and their current values. The highlighted attribute is open for editing in the data entry field at the top of the dialog box.

2 Type the value into the text box at the top of the dialog box.

See Optional Attribute Editing Controls on page 65 for information about how the CM Admin CSDB Editor determines which edit control to use for

a particular attribute, based upon a specification in the attribute's description in the class template.

3 To highlight and select a different attribute, click its name.

4 Continue to select each attribute you want to edit, and to type in the desired value for each, until you have completely edited the instance.

5 Click **OK** to close the dialog box and save your changes.

An Instance Edit Confirmation message box opens if you selected the Prompt: Before Instances Updated check box on the Instance Options tab of the Options dialog box (see Setting Preferences: The Options Dialog Box starting on page 30).

6 If the Instance Edit Confirmation message box opens:

— Click **Yes** to complete the update of the instance.

— Click **No** to discard your changes to the instance, and return to the CM Admin CSDB Editor main window.

— Click **Cancel** to return to the Editing Instance dialog box.

When editing connection attributes, notice the warning about editing class connections manually, at the bottom of the dialog box. Connection values can be complicated to type, and a single typographical error can produce unexpected and undesirable results during resolution. The recommended way to make class connections between attributes is to use the drag-and-drop operation, which is described in Drag-and-Drop Connections on page 91.

Editing Multiple Instances

The CM Admin CSDB Editor enables you to edit multiple instances in a single operation.

A multiple instance edit operates upon a set of selected instances. One of the selected instances serves as a template. The template instance's values appear in the Editing Instance dialog box. When you change a value in the template instance, the change is applied to all of the instances in the selected set. No changes are stored in the CM-CSDB until you exit the dialog box and confirm the change operation.

To initiate and execute a multiple instance edit

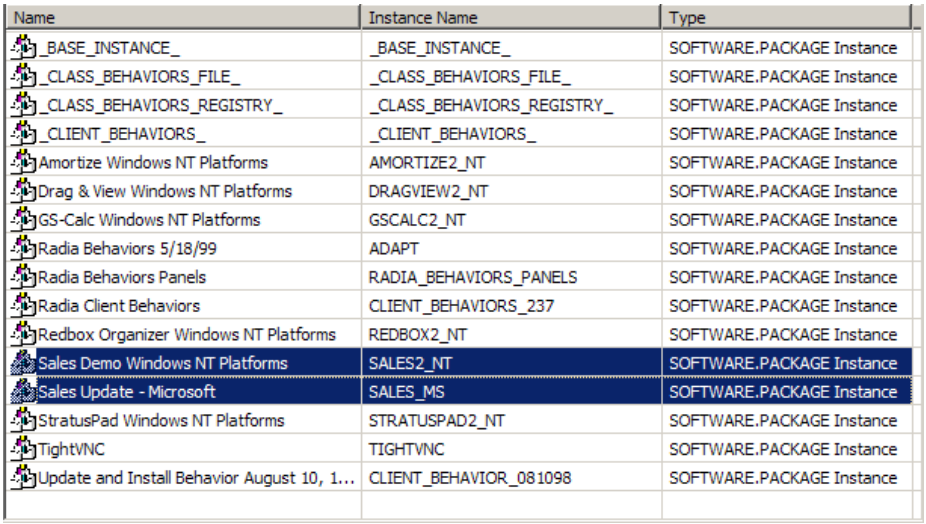
1 Select the instances in the list view of the CM Admin CSDB Editor window.

Click on an instance to select it.

— **Ctrl**-click to add an instance to the selected set; or

— **Shift**-click and/or **Ctrl**-**Shift**-click to add a range of instances to the selected set.

Selected instances are highlighted:



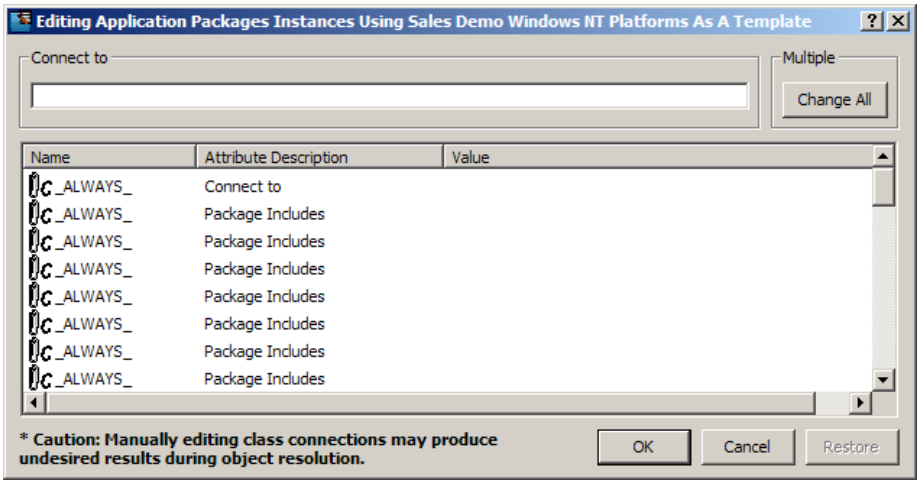
Name	Instance Name	Type
_BASE_INSTANCE_	_BASE_INSTANCE_	SOFTWARE.PACKAGE Instance
_CLASS_BEHAVIORS_FILE_	_CLASS_BEHAVIORS_FILE_	SOFTWARE.PACKAGE Instance
_CLASS_BEHAVIORS_REGISTRY_	_CLASS_BEHAVIORS_REGISTRY_	SOFTWARE.PACKAGE Instance
_CLIENT_BEHAVIORS_	_CLIENT_BEHAVIORS_	SOFTWARE.PACKAGE Instance
Amortize Windows NT Platforms	AMORTIZE2_NT	SOFTWARE.PACKAGE Instance
Drag & View Windows NT Platforms	DRAGVIEW2_NT	SOFTWARE.PACKAGE Instance
GS-Calc Windows NT Platforms	GSCALC2_NT	SOFTWARE.PACKAGE Instance
Radia Behaviors 5/18/99	ADAPT	SOFTWARE.PACKAGE Instance
Radia Behaviors Panels	RADIA_BEHAVIORS_PANELS	SOFTWARE.PACKAGE Instance
Radia Client Behaviors	CLIENT_BEHAVIORS_237	SOFTWARE.PACKAGE Instance
Redbox Organizer Windows NT Platforms	REDBOX2_NT	SOFTWARE.PACKAGE Instance
Sales Demo Windows NT Platforms	SALES2_NT	SOFTWARE.PACKAGE Instance
Sales Update - Microsoft	SALES_MS	SOFTWARE.PACKAGE Instance
StratusPad Windows NT Platforms	STRATUSPAD2_NT	SOFTWARE.PACKAGE Instance
TightVNC	TIGHTVNC	SOFTWARE.PACKAGE Instance
Update and Install Behavior August 10, 1...	CLIENT_BEHAVIOR_081098	SOFTWARE.PACKAGE Instance

2 Right-click on the instance that you want to use as the template for the multiple instance-edit operation.

A shortcut menu appears:

3 Select **Edit Multiple Instances** from the shortcut menu.

The Editing Instance dialog box appears:



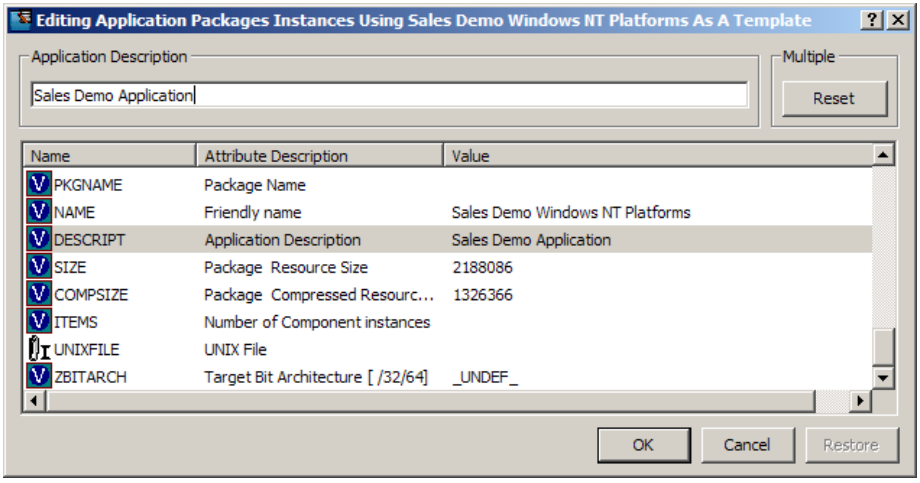
The title bar displays how many instances are in the set of selected instances, the selected instances' class friendly name, and the friendly name of the instance serving as the template.

4 Scroll to and select a variable whose value you want to copy to all instances in the selected set.

— If the selected variable in the template instance already has the desired value, click **Change All** to copy that value to the other instances in the selected set.

— If the selected variable does not have the desired value, edit the value.

In either case, the caption of the Change All button will change to Reset.



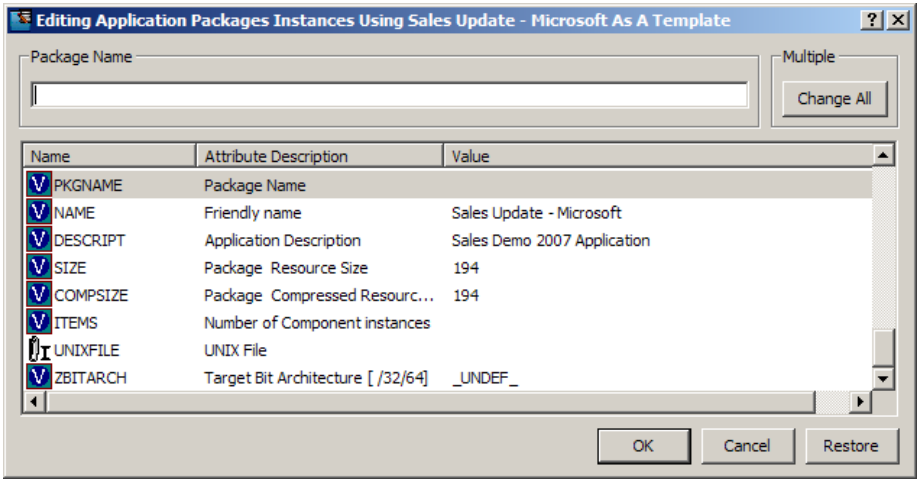
To restore the value of a currently selected variable to its original contents:

— Click **Reset**.
As you scroll through the instance's variables, you can determine if you've made a change to the selected variable by referring to the caption on this button.

  – If the caption is Change All, you have made no change to the variable during the current Editing Instance dialog box session.

  – If the caption is Reset, you have made a change to the selected variable.

After making a change to at least one variable, and selecting a different variable, the Restore button becomes enabled:



To undo all changes made in the current Editing Instance dialog-box session

1 Click **Restore**.

The Confirm Restore message opens.

2 Click **Yes** to discard all changes made in the current Editing Instance dialog box session.

Click **No** to resume the Editing Instance dialog box session without restoring original values.

Changes made during the Editing Instance dialog box session are not saved in the CM-CSDB until you click **OK**. You are prompted to confirm your changes.

3 Click **OK** to save the changes made in the template instance, to all of the selected instances, into the CM-CSDB.

Click **Cancel** to resume the Editing Instance dialog box session.

You can click **Cancel** in the Editing Instance dialog box to exit the dialog box without saving any changes to the CM-CSDB.

Renaming an Instance

Be careful when renaming instances. Instances are connected together via the connection attributes they contain. When you rename an instance, any other instance connected to the renamed instance will have a broken connection. This can produce unexpected and undesirable effects during resolution. Before renaming an instance, make sure you know which other instances are connected to the renamed instance, and edit them appropriately.

To rename an instance

1 In the tree view right-click the instance to be renamed.

A shortcut menu opens.

2 Select **Rename Instance** from the shortcut menu.

A Rename Instance dialog box opens.

3 Enter a new display name (that is, a friendly name) and/or a new instance name for the instance, and click **OK**.

Deleting an Instance

Be careful when deleting instances. Instances are connected via the connection attributes they contain. When you delete an instance, any other instance connected to the deleted instance will have a broken connection. This can produce unexpected and undesirable affects during resolution. Before deleting an instance, make sure you know which other instances are connected to the instance to be deleted, and edit them appropriately.

To delete an instance

1 Right-click on the instance to be deleted.

A shortcut menu opens.

2 Select **Delete Instance**.

The Confirm Delete message opens.

3 Click **Yes** to delete the instance.

Click **No** to close the dialog box without deleting the instance.

Summary

- The CM-CSDB stores your desired state.
- The CM-CSDB is hierarchically organized and subdivided into files, domains, classes, and instances.
- The tree view of the CM Admin CSDB Editor tracks the navigation history during an open session.
- Use compound names to help you manage large numbers of instances within the CM-CSDB.
- The base instance of each class provides default values for instances in that class. Any attribute in an instance will assume the value for that attribute found in the base instance of the class unless a specific value has been entered for that attribute in the instance.
- Use the CM Admin CSDB Editor shortcut menus to help you add, edit, copy, rename, and delete CM-CSDB components.

4 Building CM-CSDB Desired-State Policy

At the end of this chapter, you will:

- Understand the POLICY and SOFTWARE Domains.
- Be able to create instances of the POLICY Domain.
- Understand how to develop policies that meet your organizational needs.
- Understand how to connect policy instances to services.
- Be familiar with the HP OpenView Configuration Management Policy Server (CM Policy Server) and its uses.

Building Desired-State Policy for Software Distribution

Instances in the HP OpenView Configuration Management Configuration Server Database (CM-CSDB), the values they contain, and the connections among them define your enterprise's desired state. Most of CM administration begins in two domains:

- POLICY Domain
- SOFTWARE Domain

The Policy Domain

Each agent computer that CM manages is represented by an instance of the POLICY Domain USER Class. Use it to organize your subscribers into logical groups in preparation for distributing software.

Figure 17 PRIMARY.POLICY.Users Class

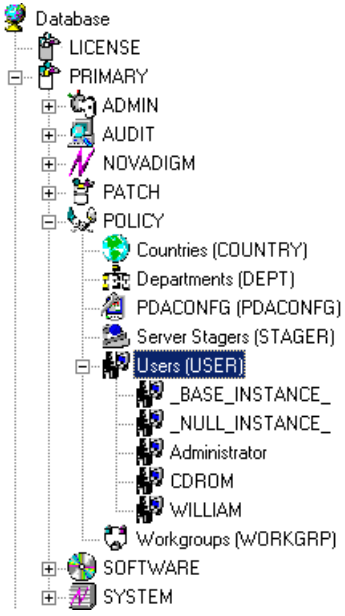


Table 20 on page 81 lists the POLICY Domain default classes that are delivered with the CM-CSDB version 4.1 or above. Refer to the *HP OpenView*

Configuration Management Configuration Server Database Reference Guide (CM CDSB Reference Guide) for more information on the POLICY Domain classes.

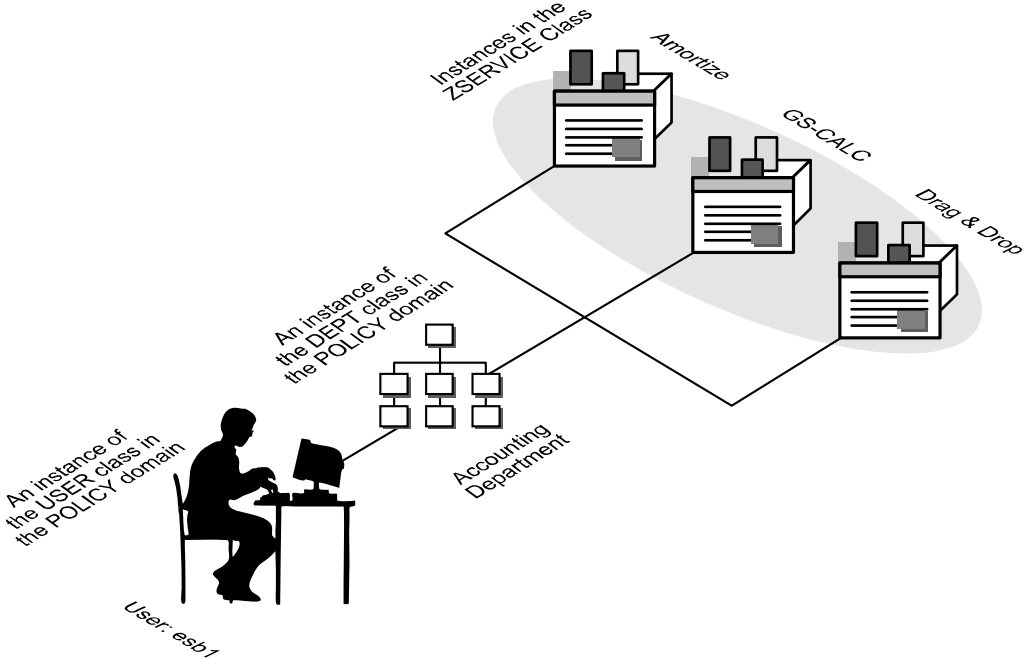
Table 20 POLICY Domain classes

Class	Description
Countries (COUNTRY)	Use for clock synchronizations with the CM Configuration Server. Do not assign services to this class.
Departments (DEPT)	Use to group subscribers into departments.
Machine Manufacturer (MANUFACT)	Use to assign policy based on the machine manufacturer. Refer to the *HP OpenView Configuration Management OS Manager User Guide (CM OS Manager Guide)* for more information.
Machine Models (MODEL)	Use to assign policy based on the machine model. Refer to the *CM OS Manager Guide* for more information.
Machine Roles (ROLE)	Use to assign policy based on the machine role. Refer to the *CM OS Manager Guide* for more information.
Machine Subnets (SUBNET)	Use to assign policy based on the machine subnet. Refer to the *CM OS Manager Guide* for more information.
Multicast (MULTCAST)	Use to configure agent computers to use multicasting. Refer to the *HP OpenView Configuration Management Multicast Server Installation and Configuration Guide (CM Multicast Server Guide).*
Mobile Device Confg (MBLCONFG)	Use to define the parameters for mobile device configuration when using the Radia Mobility Server (Mobility Server). Refer to the *Radia Mobile Management Guide* for more information.
PDACONFG (PDACONFG)	Use to define configuration parameters of, and to obtain inventory reports of, cradled-PDAs. This class allows the administrator to target and store PDA-specific information during resolution.
Server Stagers (STAGER)	Use to define HP OpenView Configuration Management Proxy Servers (CM Proxy Servers) or HP OpenView Configuration Management Staging Servers (CM Staging Servers) in your distribution network. Also, use to define the storage location on a CM Staging Server computer.
Users (USER)	Use to define individual subscribers. If you want to create policies at the user level, then you must have an instance in the USER Class for each subscriber.

Class	Description
Workgroups (WORKGRP)	Use to group subscribers into functional groups. For example, a project team could be made up of subscribers from several different departments.

Users can be grouped by connecting them to instances of other POLICY Domain classes such as Departments or Workgroups classes. This allows CM to manage content on agent computers based on the user belonging to a workgroup, or being a member of a department.

Figure 18 Create a policy for your organization.



You will probably begin developing your desired state by creating new users.

To create a new user

1 From the Windows taskbar, click **Start →Programs → HP OVCM Administrator → CM Admin CSDB Editor**. The CM Admin CSDB Editor Security Information dialog box opens.

The factory set user ID is RAD_MAST. No password is necessary. Your CM security administrator might have changed this during installation. Check with him or her to obtain your own user ID and password, if necessary.

2 If necessary, type a user ID and password, then click **OK**.

The CM Admin CSDB Editor splash screen briefly appears.

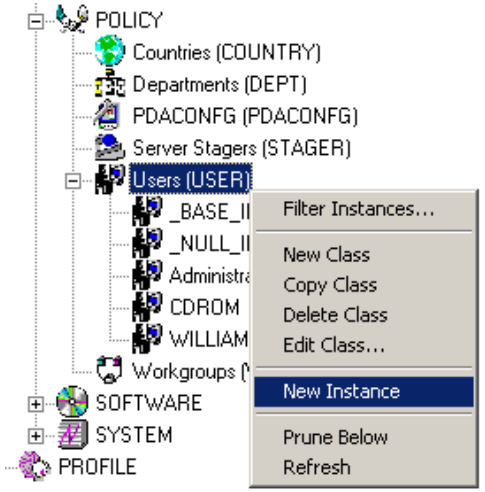
3 Double-click **PRIMARY**.

4 Double-click **POLICY**.

5 Double-click **Users (USER)** to expand the class. A list of all of the USER instances opens.

6 Right-click **Users**.

A shortcut menu opens.



7 Select **New Instance** from the shortcut menu.

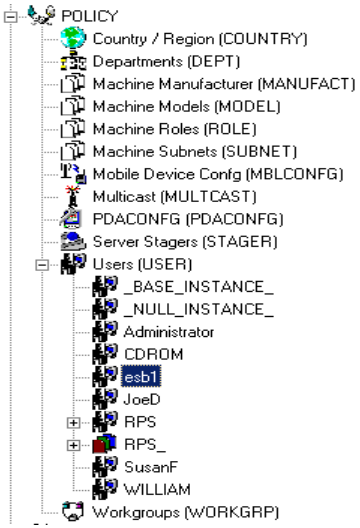
The Create Instance dialog box opens.

8 Enter the display name of the new user. This is the friendly name as it will appear in the tree view. As an example, create a new user instance, ESB1.

9 Enter a user's instance name.

10 Click **OK.**

A new USER instance, esb1, is created.

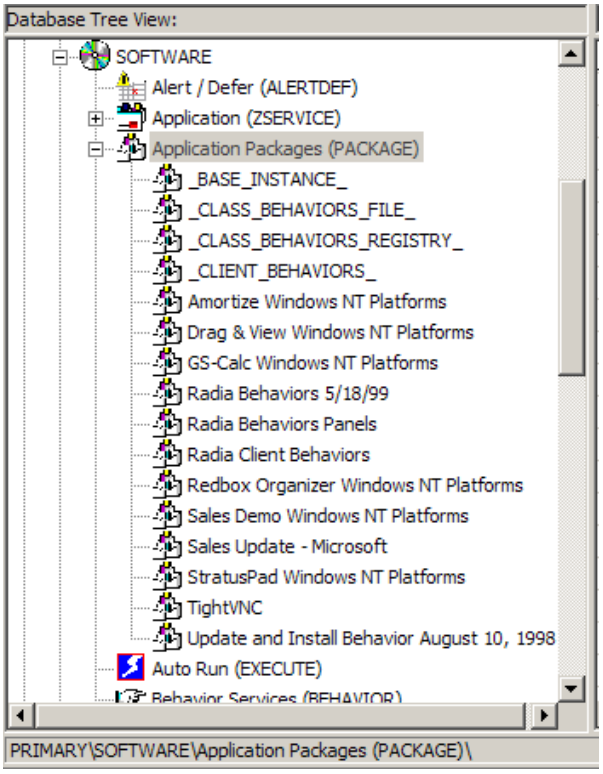


Use the CM Admin CSDB Editor to create, copy, delete, rename, and edit user instances that suit your enterprise needs.

Software Domain

An instance of the SOFTWARE Domain PACKAGE Class represents one software application or unit of other content to be managed on agent computers. The CM-CSDB is shipped with sample packages shown in the following example.

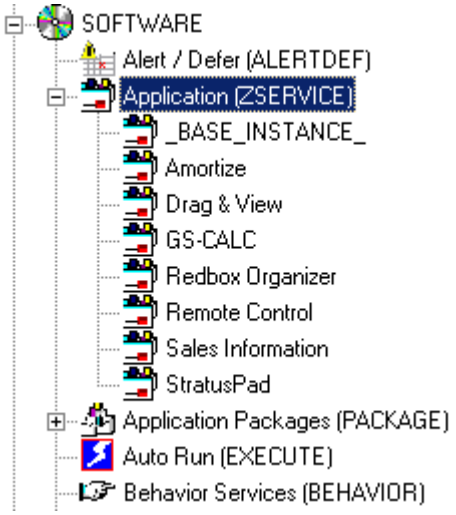
Figure 19 Application Packages (PACKAGE) Class



The administrator uses the CM Administrator Packager (CM Admin Packager) to package and publish content (software, text files, and so forth) to the CM-CSDB. These packages are stored in the Application Packages (PACKAGE) Class.

After a package has been created, the administrator needs to create an Application (ZSERVICE) for the package to be deployed to end users. This is referred to as creating a Service. The CM-CSDB ships with sample Services for the packages within the PACKAGE Class.

Figure 20 Sample Application (ZSERVICE) instances

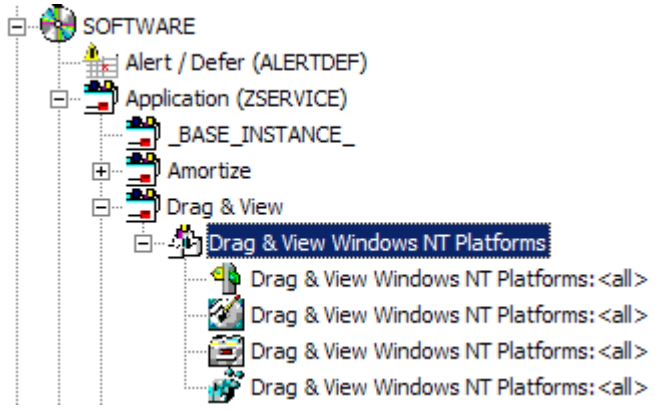


For information on packaging and publishing application packages, see the *CM Application Manager Guide* or the *CM Application Self-service Manager Guide*. For information on creating a ZSERVICE, see Running the New Application Wizard on page 108.

The ZSERVICE instance that represents a unit of managed content can be connected to many other class instances. Those class instances can then be connected further to other class instances. If, for example, the unit of managed content is a software application, each file in the application will be represented by an instance of the FILE Class. In addition, there can be instances of the REGISTRY Class that represent an application's registry keys, instances of the DESKTOP Class representing the links and shortcuts that invoke the application after it is installed, and so forth. The desired state for a complex application such as MS Word can consist of several thousand connected instances within the CM-CSDB.

The CM Admin CSDB Editor graphically shows the connections among instances in the tree view. This enables the CM administrator to work with instances that are connected to one another in a localized portion of the tree view.

Figure 21 Example of Drag & View service and its connections



Inter-Package Relationships

Relationships between packages are established by the kind of connection that exists between the PACKAGE Class instances in the CM-CSDB. Each variable can hold a connection to another package.

Figure 22 Variable connections to packages

C _ALWAYS_	Package Includes
C _ALWAYS_	Package Includes
C _ALWAYS_	Package Includes
C _ALWAYS_	Package Includes
C _ALWAYS_	Package Includes
C _ALWAYS_	Package Includes
I INCLUDES	Package Includes
I INCLUDES	Package Includes
I INCLUDES	Package Includes
I INCLUDES	Package Includes
I INCLUDES	Package Includes
I INCLUDES	Package Includes
R REQUIRES	Required Package Level
R REQUIRES	Required Package Level
R REQUIRES	Required Package Level
R REQUIRES	Required Package Level
R REQUIRES	Required Package Level
R REQUIRES	Required Package Level

Each variable in the above example can hold a connection to another package.

Two packages can be related such that either one package requires the other, or one package includes the other.

Requires Connection

When one package requires another, the components of the requiring package supersede the same components in the required package. For example, if package A requires package B, the components in A supersede any of the same components in B. This relationship is useful for deploying maintenance to installed application software, where Maintenance Package A requires the Base Package B.

Consider the following example of applying the Service Release 1 update to the Microsoft Office application suite.

When MS Office was introduced, there was no Service Release 1 update. The CM administrator packaged the components of the Office suite for Windows. Later, when Microsoft issued Service Release 1 update for Office, the CM administrator was able to create new packages for each component and for each operating system which representing the difference between the base install of the component and the Service Release 1 update. This was done by having the Service Release 1 Update require the packaged base install of the Office suite for Windows.

Includes Connection

The ability to create a relationship between CM packages such that one package includes another package enables the CM administrator to deploy application software constructed from multiple CM packages.

When one package includes another, both are deployed.

A typical use for this capability arises in installations where the application software to be deployed must be accompanied by customized extensions.

For example, your organization might have a series of customized Microsoft Word templates that need to be deployed along with MS Word. The CM administrator can package MS Word in a separate package for each target platform, and package the customized templates in a separate package of their own. Using the CM Admin CSDB Editor, the administrator would create an Includes connection in each of the MS Word package instances, connecting to the package instance representing the customized templates. MS Word would then be deployed with the customized templates whenever a subscriber installs MS Word.

Determining Policies

After using the CM Admin Packager and CM Admin CSDB Editor to create services, you must determine who gets what. In other words, you must set up policies that identify which user receives what software.

Before you set up policies in CM, you will want to have a carefully constructed plan in place. Consider the following:

- What group policies do you need to create?
- What groups should your users be assigned to?
- What software will CM manage for the users?

Later, you will assign each USER instance to one or more groups, as necessary. Policies are easier to administer when subscribers are grouped according to the content that CM will manage for them.

For example, your sales department might have 50 employees. Rather than authorizing each user for a spreadsheet application, you might want to create an instance in the Departments class called Sales. Then, you can group the 50 employees into the Sales department instance, allowing you to authorize applications, or even take away authorization to applications, for all 50 employees simultaneously.

In the next example, you will create a Departments (DEPT) instance called Sales.

To create a Departments (DEPT) instance

1 From the Windows taskbar, click **Start** → **Programs** → **HP OVCM Administrator** → **CM Admin CSDB Editor**. The CM Admin CSDB Editor Security Information dialog box opens.

The factory set user ID is RAD_MAST. No password is necessary. Your CM security administrator might have changed this during installation. Check with him or her to obtain your own user ID and password, if necessary.

2 If necessary, type a user ID and password, then click **OK**.

The CM Admin CSDB Editor splash screen momentarily appears.

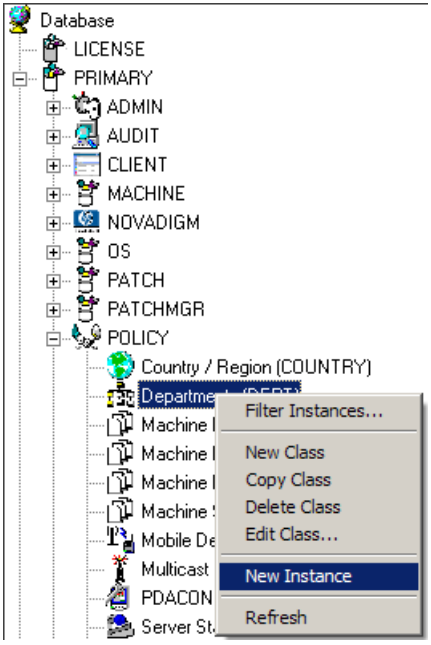
3 Double-click **PRIMARY**.

4 Double-click **POLICY**.

5 Double-click **Departments (DEPT)** to expand the class.

6 Right click on the **Departments (DEPT)** class.

A shortcut menu opens.



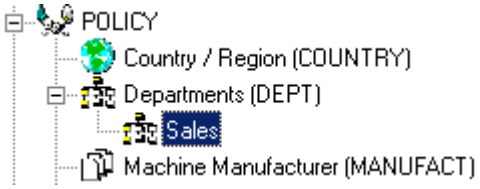
7 Select **New Instance**.

The Create Instance dialog box opens.

8 Enter the display and instance names in the appropriate fields. For this example, we use Sales.

9 Click **OK**.

The Sales department is added.



The next example shows how to connect the user ESB1 to the Sales instance.

Connection Types

You can connect CM-CSDB instances to one another in several ways.

Manual Connections

You can set the connections between instances manually by typing the path of the connection into a Connection attribute. See Table 18 on page 60 for additional information about connection types.

While we support this method, we recommend that you use drag-and-drop or the Show Connections dialog box, as shown below, to create connections.

Drag-and-Drop Connections

The CM Admin CSDB Editor features the ability to make drag-and-drop connections between instances of connectable classes. Use this feature to avoid two common errors that occur when you type in connection specifications between instances:

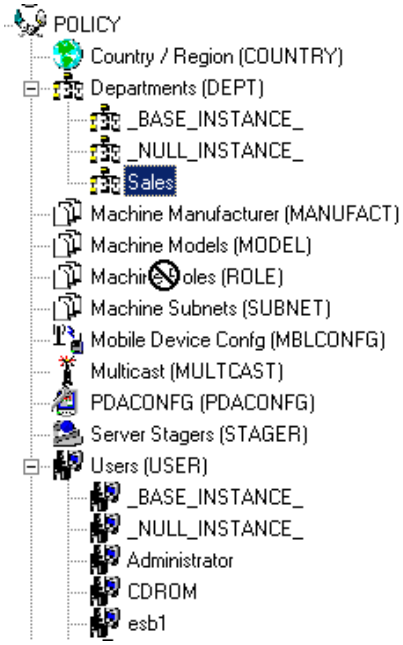
- Typographical error
- Typing in a connection to an instance of a class with which a connection is prohibited

You can also use drag-and-drop connections among multiple CM Admin CSDB Editor windows.

The drag-and-drop method is useful when the instances to be connected are close enough to display in the tree view at the same time. After you begin the dragging process, you will note that the cursor changes as you move up and down the tree view.

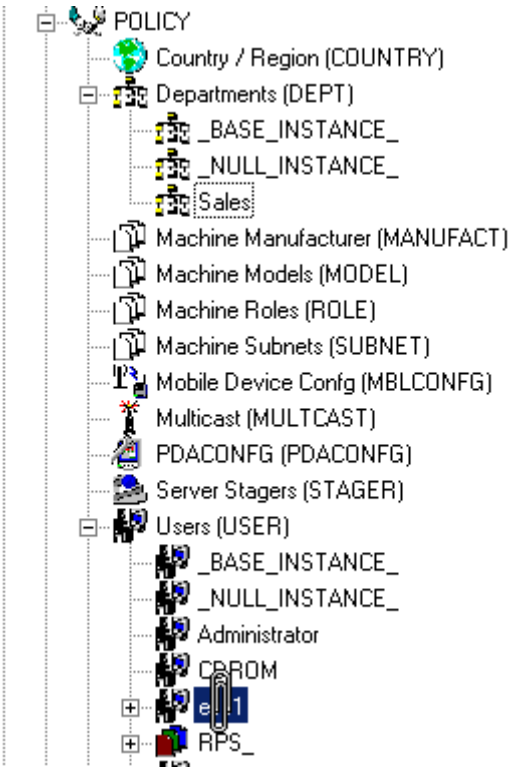
The circle with a slash indicates that a connection to that particular instance is not possible.

Figure 23 Circle with a slash



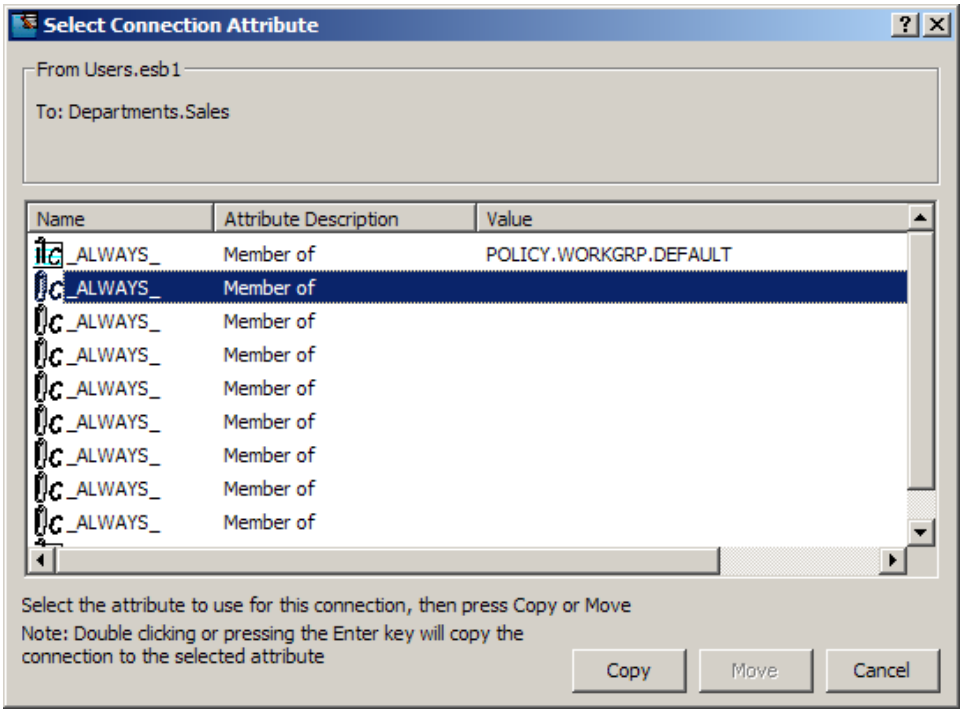
A paperclip indicates that a connection is allowed to that instance.

Figure 24 Paperclip



After you release the paperclip, the Select Connection Attribute dialog box opens.

Figure 25 Select Connection Attribute dialog box



The CM Admin CSDB Editor highlights the first eligible connection attribute in the selected instance, as shown in Figure 25 above.

Press **Enter**, or click **Copy** to accept this connection attribute. To use a different available connection attribute, scroll to it and double-click it.

In the CM Admin CSDB Editor tree view, an instance containing connections lists those connected instances in the tree immediately beneath itself. When you make a drag-and-drop connection by dragging such a connected instance, you can move the connection from the instance from which the connection was dragged.

Click **Move** to move the connection. The connection specification will be removed from the instance from which it was dragged, and will be stored in the instance on which it was dropped.

If, instead, you click **Copy**, the connection specification will be left in the instance from which it was dragged, and a copy will be stored in the instance on which it was dropped.

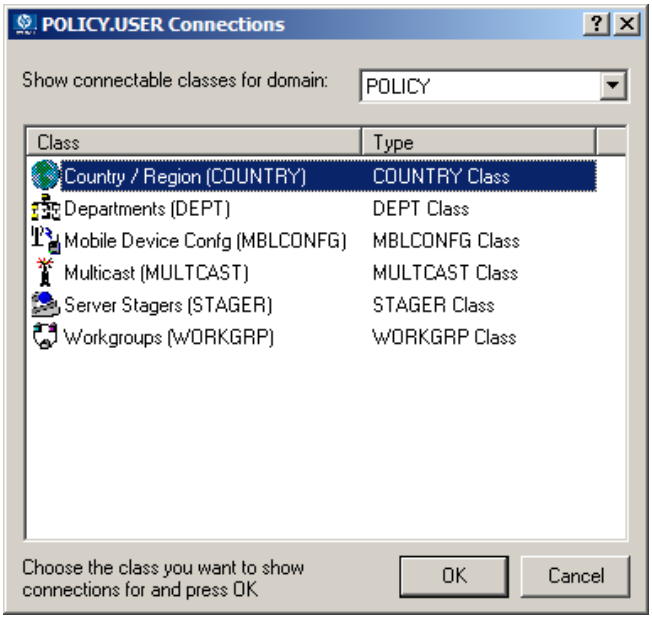
After confirming the connection, the connection is complete.

Show Connections

Use the Show Connections dialog box when the instances you want to connect are not close enough to be displayed in the tree view at the same time. Right-click an instance and choose Show Connections from the shortcut menu.

The Show connectable classes dialog box opens.

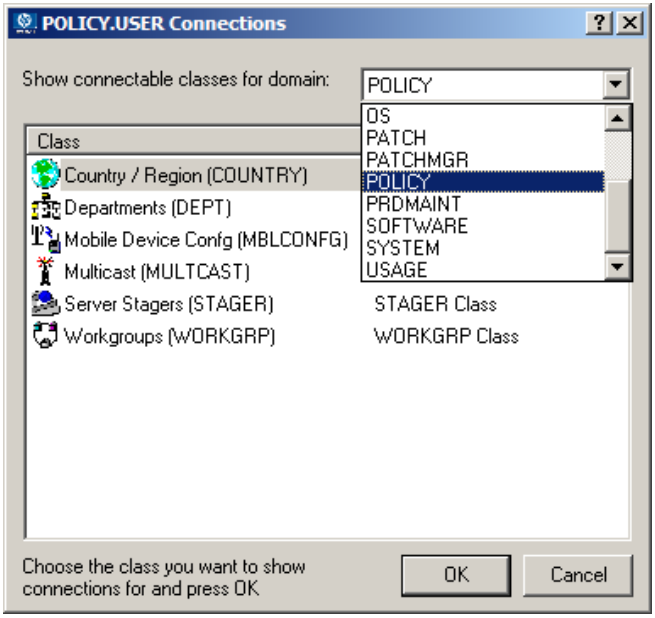
Figure 26 Show connectable classes dialog box



The dialog box lists the classes in the same domain to which connection of the selected instance is permitted.

To connect to a class in a different domain, click the **Show connectable classes for domain** drop-down list to see a list of domains.

Figure 27 Domain list for connectable classes



If there are no permitted connectable classes in a selected domain, a warning message tells you.

Rules for which classes can be connected are in the CM-CSDB. See Chapter 7, Maintaining Connection Rules in the CM-CSDB for details on how to maintain these rules.

After an acceptable class has been selected, the list view shows all instances for that class.

Now you can select the class instance from the list view (on the right), and drag-and-drop it onto the target connection instance (on the left).

To connect a user to a Departments (DEPT) instance

1 From the Windows taskbar, click **Start** → **Programs** → **HP OVCM Administrator** → **CM Admin CSDB Editor**. The CM Admin CSDB Editor Security Information dialog box appears.

The factory set user ID is RAD_MAST. No password is necessary. Your CM security administrator might have changed this during installation. Check with him or her to obtain your own user ID and password, if necessary.

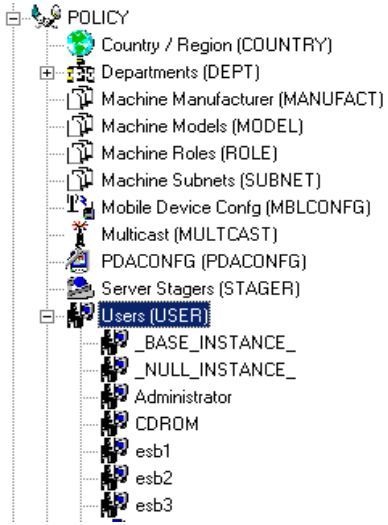
2 If necessary, type a user ID and password, then click **OK**.

The CM Admin CSDB Editor splash screen appears momentarily.

3 Double-click **PRIMARY**.

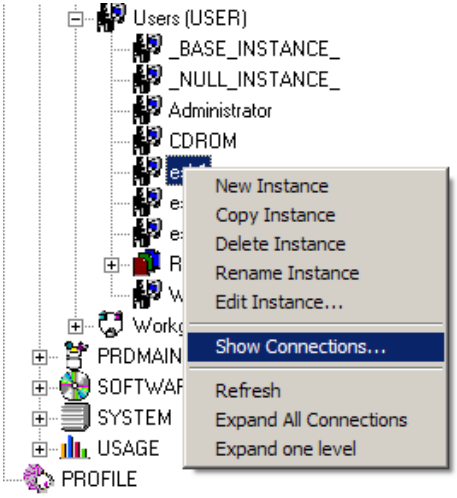
4 Double-click **POLICY**.

5 Double-click **Users (USER)** to expand the class.



6 Right-click the **esb1** user instance.

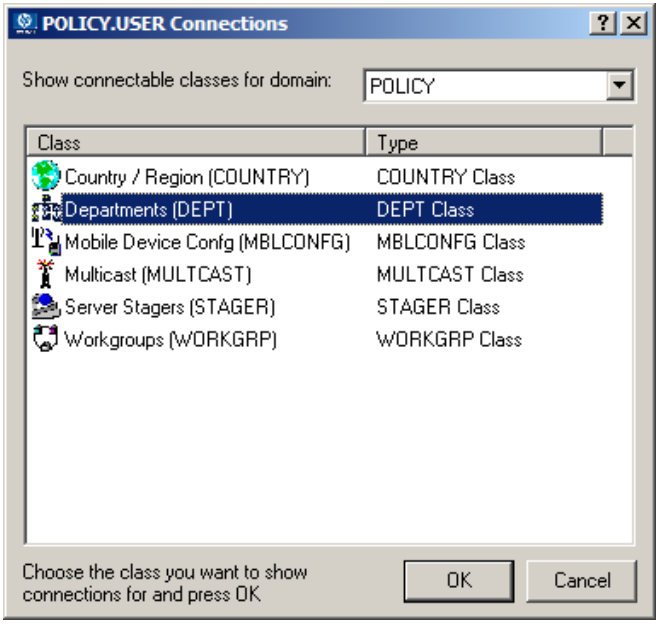
A shortcut menu opens.



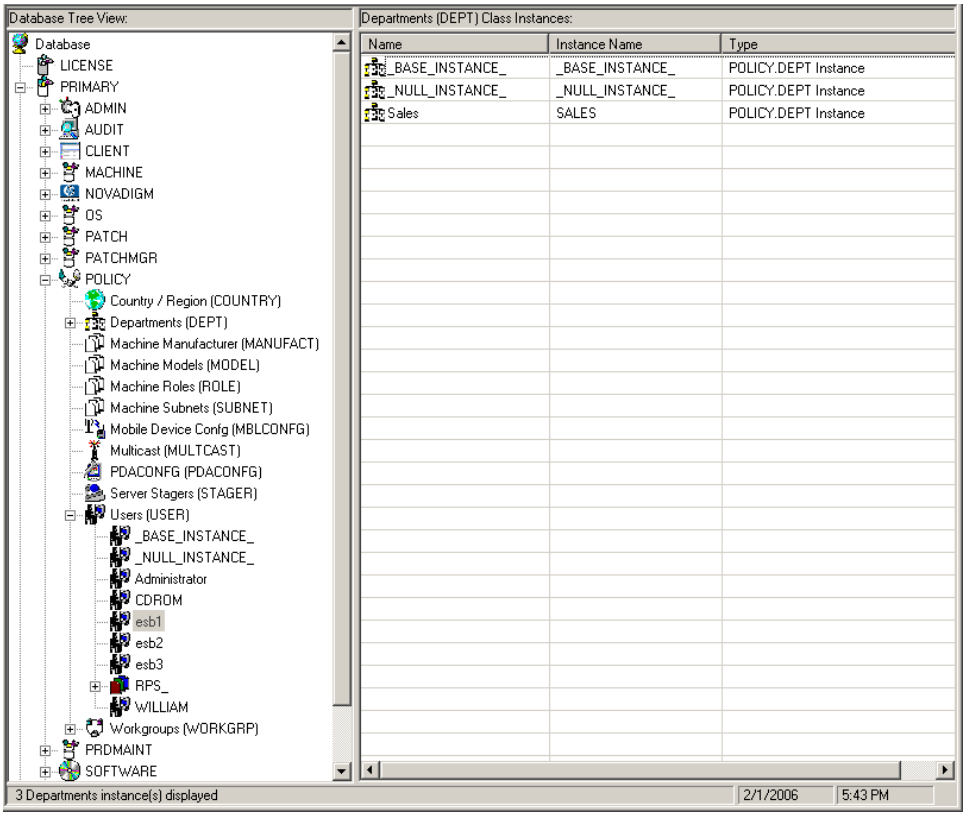
7 Select **Show Connections**.

The Show connectable classes dialog box opens.

8 Select the **Departments (DEPT)** Class and click **OK**.



This list view displays all DEPT Class instances.



9 Select the Sales instance from the list view, and drag-and-drop it onto the USER esb1 in the tree view. The Select Connection Attribute dialog box opens.

10 Click **Copy** to complete the connection.

The Confirm Connection dialog box opens.

11 Click **No** to cancel the connection process; click **Yes** to confirm.

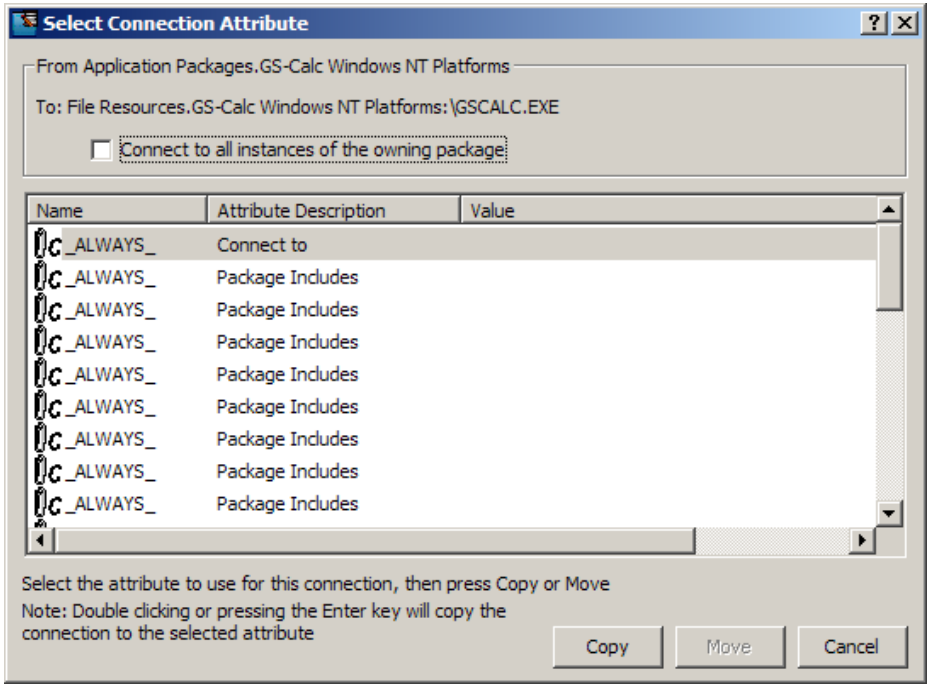
An Instance Connection dialog box opens.

Drag-and-Drop Component Class Instances

No matter what method you use, an additional option is available in the Select Connection Attribute window when you drag an instance of a component class onto an instance of the PACKAGE Class. Examples of component classes are FILE, DESKTOP, and REGISTRY.

Figure 28 below shows the **Select Connection Attribute** window that results from dragging a FILE instance onto a PACKAGE instance.

Figure 28 Select Connection Attribute dialog box



An additional option is the **Connect to all instances of the owning package** check box. When you select this check box, the connection specification stored in the PACKAGE instance will be wild-carded to include all of the dragged component class instances (in this example, FILE instances) that were published together (by CM Admin Packager) in the same application as the dragged instance. In this example, if the check box is selected, all of the GS-CALC FILE instances will be connected in a single connection specification,.

Leaving this check box cleared results in a connection specification that refers only to the instance that was dragged. In the example, this isthe instance GS-Calc Spreadsheet: `\GSCALC.EXE`.

Connecting Services to Policies

After you have determined your policies and set up your users and groups, you can begin to entitle each user or group to applications. To do so, you will connect POLICY Class instances to instances in the Application (ZSERVICE) Class.

In the following example, you will connect the Sales Department Policy instance to the Amortize Application (ZSERVICE) instance.

To connect a service to a policy

1. From the Windows taskbar, click **Start** → **Programs** → **HP OVCM Administrator** → **CM Admin CSDB Editor**. The CM Admin CSDB Editor Security Information dialog box appears.

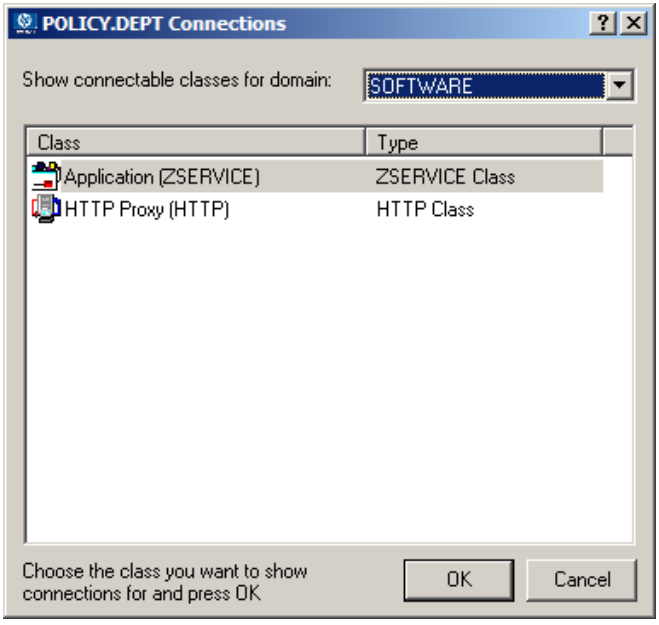
   

   The factory set user ID is RAD_MAST. No password is necessary. Your CM security administrator might have changed this during installation. Check with him or her to obtain your own user ID and password, if necessary.

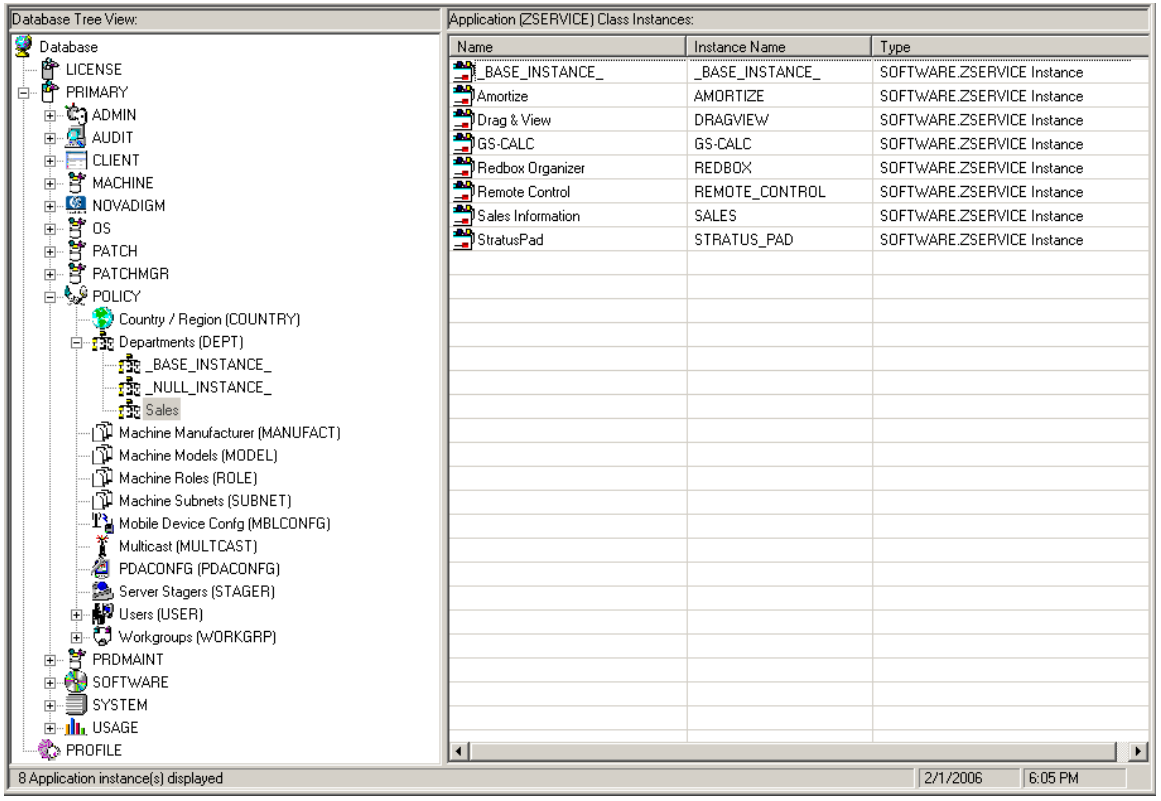
2. If necessary, type a user ID and password, then click **OK**.

   The CM Admin CSDB Editor splash screen momentarily appears.

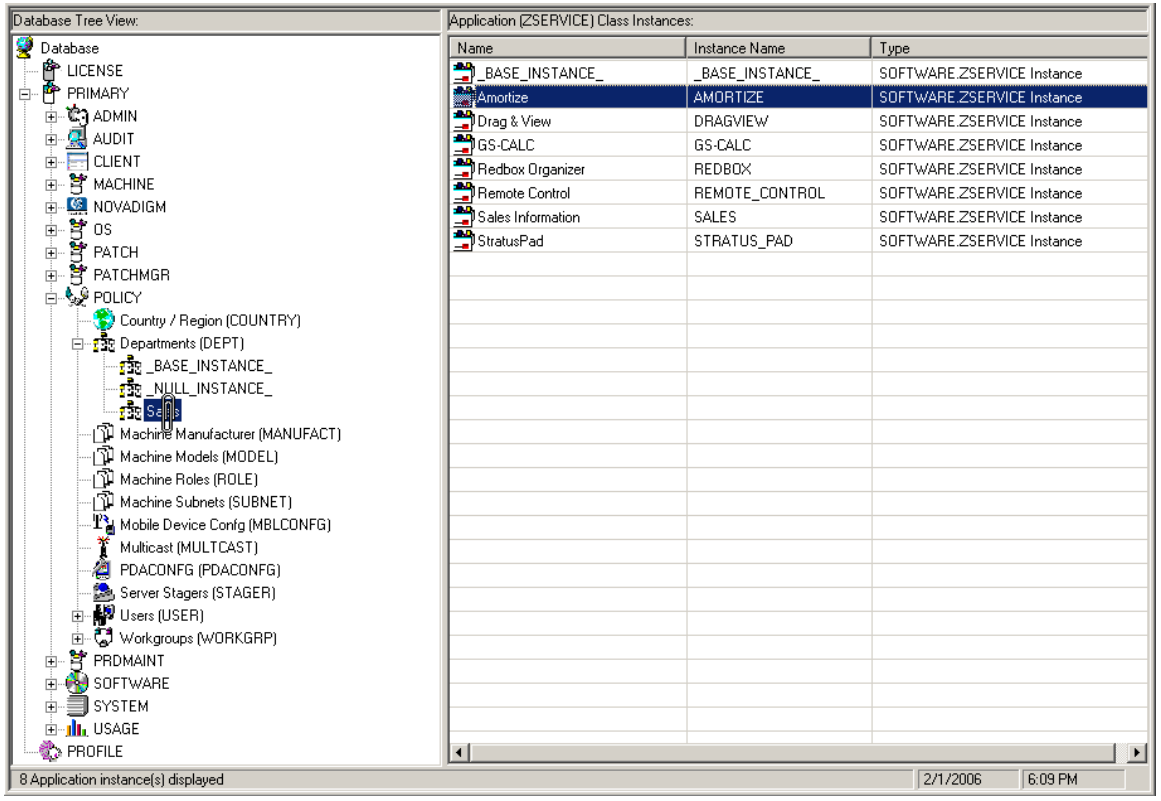
3. Double-click **PRIMARY**.
4. Double-click **POLICY**.
5. Double-click **Departments (DEPT)** to expand the class.
6. Right-click the **Sales** instance (in the tree view) and select **Show Connections**. The POLICY.DEPT Connections dialog box opens. This dialog box displays a list of classes that you can connect the selected instance to.
7. From the Show connectable classes for domain drop-down list, select **SOFTWARE**.



8 Click **Application (ZSERVICE)**, and then click **OK**. The instances in the ZSERVICE Class appear in the list view.



9 Select the **Amortize** instance from the list view and drag it to the appropriate Departments instance (in this example, Sales). When your cursor turns into a paper clip, release the mouse button.



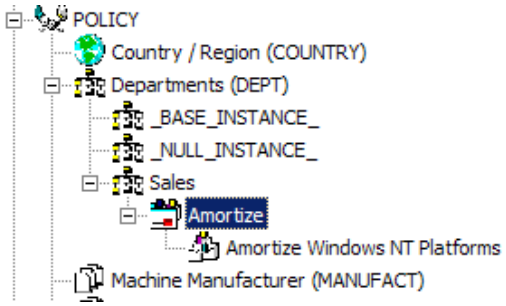
10 The Select Connection Attribute dialog box opens.

11 Click **Copy** to create the connection from the Sales department to Amortize.

12 Click **Yes** to confirm the connection.

13 Click **OK** when you receive the confirmation that Sales has been connected to the Amortize.

Notice that Amortize is listed under the Sales department instance, which indicates that the entire Sales department is now authorized to receive the Amortize application.



A Word about the CM Policy Server

The CM Policy Server is a web server used for administrative purposes such as mapping services to users in the directory tree. The CM Policy Server, an add-on to the CM Configuration Server, determines what services should be distributed and managed for the agent that is currently logged on by querying the CM Policy Server.

The CM Policy Server plug-in for the CM Integration Server leverages your investment in directory services while using CM for software management. This greatly reduces the total cost of ownership of your environment. In other words, directory services handle policy management and CM manages services. This saves you time because you do not have to define or maintain lists of users in the CM Configuration Server.

The CM Policy Server is an optional feature available with the CM Infrastructure product set. Contact your HP sales representative for details.

Summary

- Instances in the CM-CSDB, and the connections among them, define the desired state for an enterprise.
- Each agent computer managed by CM is represented by an instance of the POLICY Domain, USER Class.
- Instances of the SOFTWARE Domain, Application Packages (PACKAGE), represent one software application or unit of content to be managed on an agent computer.
- Instances of the SOFTWARE Domain, Application (ZSERVICE) Class, represent the PACKAGE resources that will be deployed. A ZSERVICE instance can contain many Application Packages.
- Define the policies that identify which users receive what software.
- You can use manual connections, drag-and-drop connections, and using the Show Connections dialog box to connect CM-CSDB instances to each other.
- The CM Policy Server is a web server used for administrative purposes such as mapping services to users in the directory tree.

5 Additional Features of the CM Admin CSDB Editor

At the end of this chapter, you will:

- Be familiar with the features of the HP OpenView Configuration Management Configuration Server Database Editor (CM Admin CSDB Editor).
- Know how to create services for packages using the **New Application Wizard**.
- Know how to configure multiple versions of a service using the **Version Group Editor**.
- Know how to optimize services by creating patches.
- Know how to create a **Service Group** to manage products that require more than one service package to establish full operation.
- Know how to use the **Notify** capabilities of Configuration Management (CM).
- Know how to filter a large database.

Features of the CM Admin CSDB Editor

This chapter describes these additional features of the CM Admin CSDB Editor:

- The New Application Wizard
- The Version Group Editor
- Service Optimization
- CM Service Groups
- CM Notify
- Filtering the CM-CSDB

For information on publishing content to the HP OpenView Configuration Management Configuration Server Database (CM CSDB), refer to the *HP OpenView Configuration Management Application Manager Guide (CM Application Manager Guide)*.

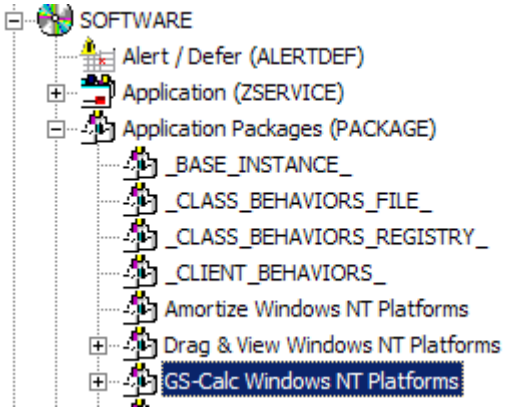
Running the New Application Wizard

The New Application Wizard automates creation of an Application (ZSERVICE Class) instance from a PACKAGE Class instance, normally created by the CM Admin Packager or the CM Packager for WI.

The GS-CALC package will be used as an example.

To create a GS-CALC service using the New Application Wizard

1. Start the CM Admin CSDB Editor and navigate to the PACKAGE Class of the SOFTWARE Domain.
2. Locate the GS-Calc package in the tree view.

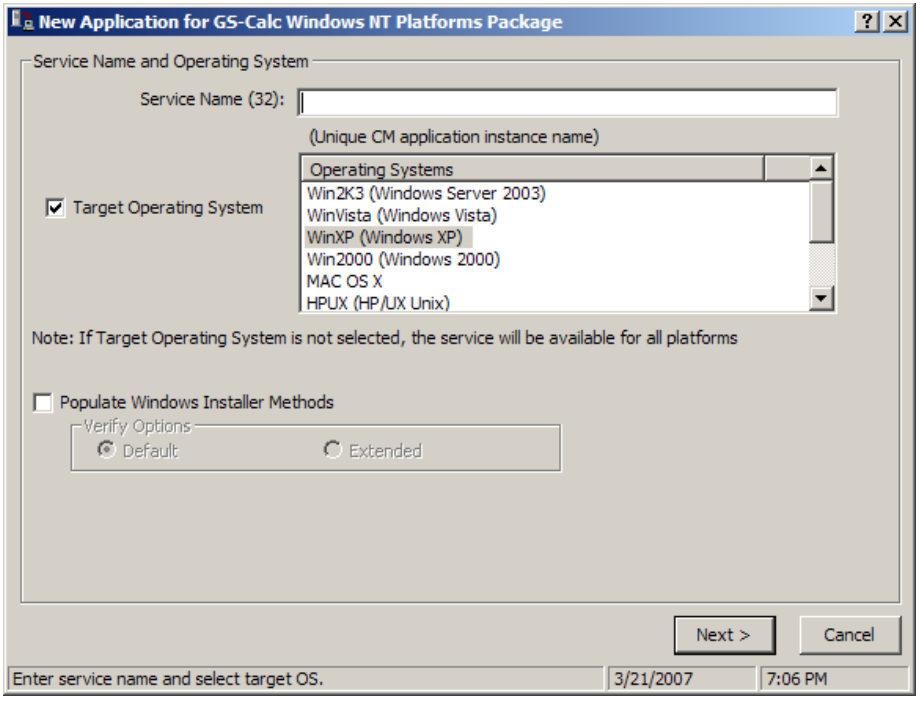


3 Right-click the **GS-Calc Windows NT Platforms** package class instance (highlighted in the above example).

A shortcut menu opens.

4 Click **New Application Wizard**.

The New Application Wizard dialog box opens.



- The initially-selected entry in the Operating Systems list reflects the OS on which the CM Admin CSDB Editor is running.
- The New Application Wizard dialog box varies for applications created with the CM Packager for WI. See Defining MSI Basic Behaviors on the New Application Wizard on page 117.

5 Type a name (up to 32 characters) in the Service Name text box for the service instance (in the SOFTWARE DOMAIN ZSERVICE Class) that represents the application software.

6 Indicate on which operating systems the application package can run:

If the Target Operating System check box is cleared, you will not be able to select a specific operating system. The application will be available to all end users regardless of their operating system.

If the Target Operating System check box is selected, you need to select which operating systems this particular package will run on. Click, **Shift**-click, and **Alt**-click can be used to select more than one operation system.

If the package was created with the CM Packager for WI and contains a connection to the MSIBASIC Class in the CM-CSDB, you have the opportunity to define the MSI Basic Behaviors. See the topic Defining MSI Basic Behaviors on the New Application Wizard on page 117.

7 If the package was created with the CM Admin Packager:

Select the **Populate Windows Installer Methods** check box for Windows Installer-enabled applications. This inserts the proper calls to HP deployment and management methods for Windows Installer-enabled applications into the method variables of the service instance created by the New Application Wizard.

If the Populate Windows Installer Methods check box is selected, select whether to use the Default or Extended Verify Options. CM provides the ability to perform default or extended verification on Windows Installer packages.

Select **Default Verify Options** to verify only the files marked as **key files** in the Windows Installer package; basically this is what MSI installer would verify. This means that if an INI file or some other type of file is not marked as a key file, and it is the only file missing from the package, it will not be repaired. Default verification sets the ZSERVICE.ZVERIFY to RadiaMSI /vd.

Select **Extended Verify Options** to verify all files in the Windows Installer package. Therefore, if a file not marked as a key file cannot be found, it will initiate a repair. Selecting extended verification sets the ZSERVICE.ZVERIFY to RadiaMSI /ve.

This check box should be selected for Windows Installer-enabled services only.

The New Application Wizard will populate the following four variables in the Application class instance that it creates, with the appropriate call to the MSI method, as follows:

Table 21 Variables populated by the New Application Wizard

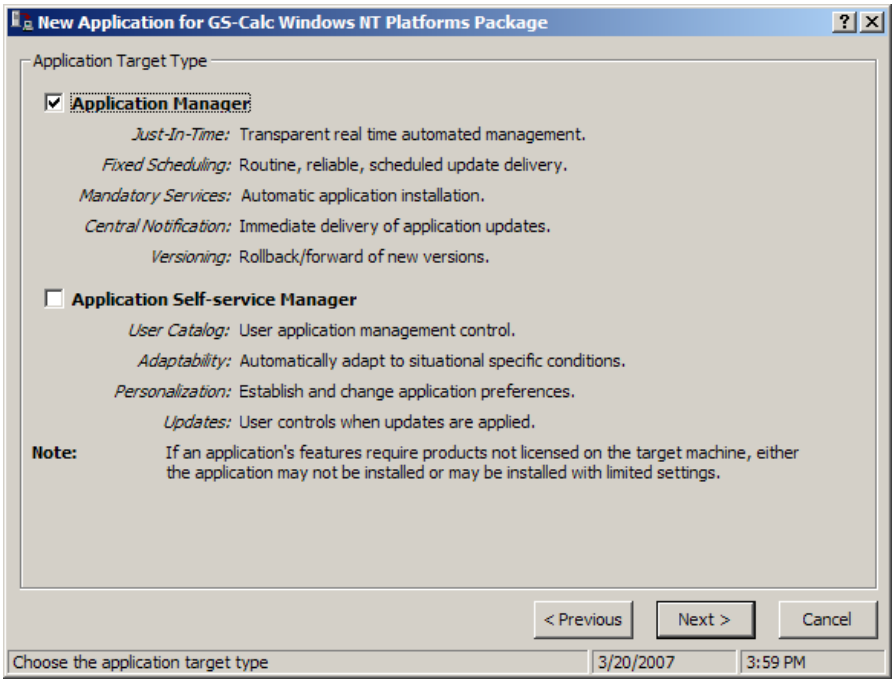
Variable (Option)	Inserted Method Call
ZCREATE	RadiaMSI /I
ZDELETE	RadiaMSI /u
ZUPDATE	RadiaMSI /I
ZVERIFY (Default)	RadiaMSI /vd
ZVERIFY (Extended)	RadiaMSI /ve

If the New Application Wizard displays a warning message indicating that one or more method variables are missing, update the Application Class template to include the missing variables. For information on editing class templates, see Editing a Class beginning on page 57.

8 Click **Next**.

The Application Target Type dialog box opens.

Select one of the two check boxes to indicate which type of support CM must provide for this application. The differences between the CM Application Manager and the CM Application Self-service Manager are listed beneath each choice. You must choose one or the other. If you click **Next** without making a choice, or select both check boxes, you will receive an error message, and you will be asked to try again.

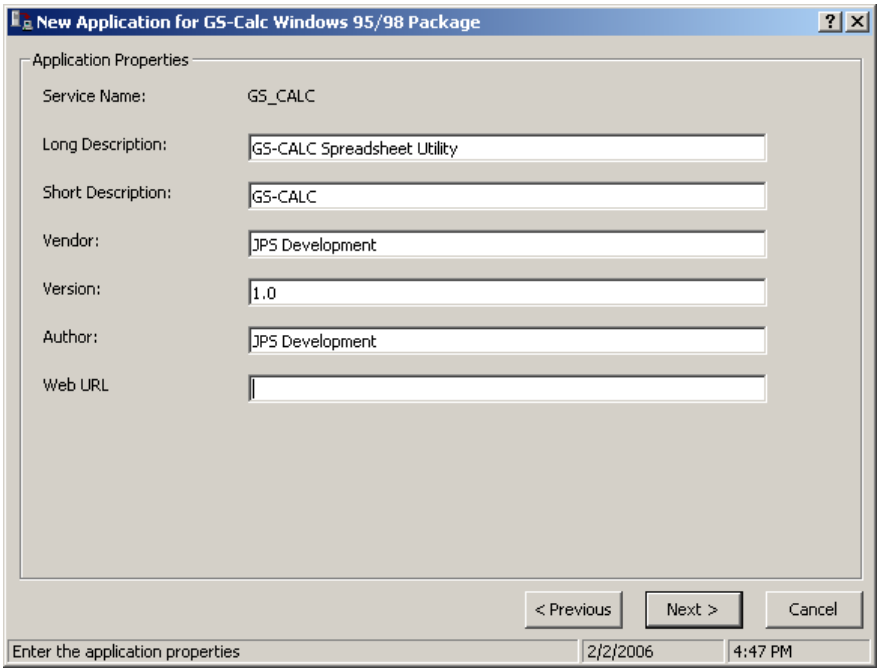


Selecting **Application Manager** makes the application mandatory for end users, and stores M in the ZSVCMO variable of the Application (ZSERVICE) instance for this application in the CM CSDB.

Selecting **Application Self-service Manager** makes the application optional for end users, and stores O in the ZSVCMO variable of the Application (ZSERVICE) instance for this application in the CM-CSDB.

9 Click **Next**.

The Application Properties dialog box opens.



Fill in the data entry fields as described in Table 22 below.

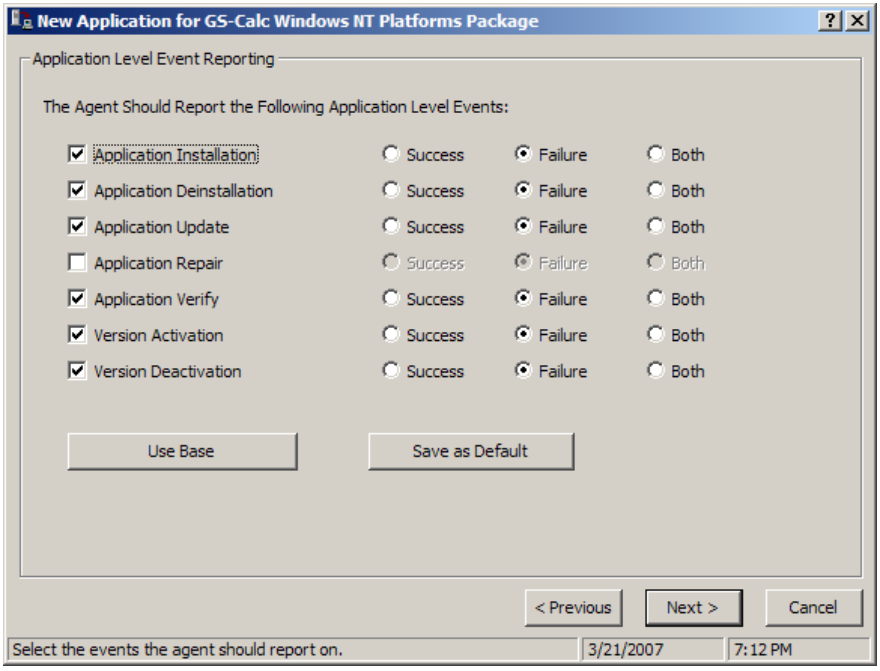
Table 22 Application Properties data entry fields

Property	Stored in ZSERVICE Attributes	Usage
Long Description	DESCRIPT	This name will appear to subscribers in the CM Subscriber Service List.
Short Description	ZSVCNAME NAME	This will appear in the CM Admin CSDB Editor's tree view as the friendly name of the Application (ZSERVICE) instance created by the New Application Wizard.
Vendor	VENDOR	The subscriber can configure his user interface to view the vendor name in his CM Subscriber Service List. This will help him identify applications.
Version	VERSION	The subscriber can configure his user interface to view the version number of the application in his CM Subscriber Service List.
Author	AUTHOR	The subscriber can configure his user interface to view the author of the application in his CM Subscriber Service List.

Property	Stored in ZSERVICE Attributes	Usage
Web URL	URL	The subscriber can configure his user interface to view the Web URL of the application in his CM Subscriber Service List. This is the Web address where he can obtain additional information about the application.

10 Click **Next**.

The Application Level Event Reporting dialog box opens.



Use this dialog box to indicate which events the CM agent must report to the CM Configuration Server. To enable reporting for an event, select its check box. After reporting is enabled for an event, indicate whether to report success, failure, or both by selecting the event's option buttons. The example above requires CM agents to report failure of application installation, de-installation, update, repair, or verify to the CM Configuration Server.

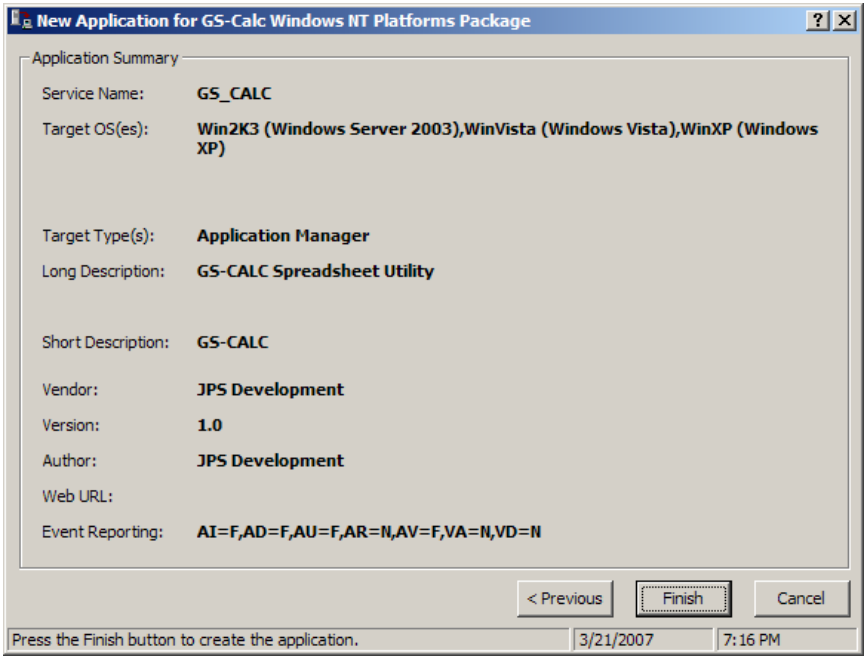
There are also two buttons: Use Base and Save as Default.

Select **Use Base** if you would like the package to inherit the settings from the base instance of the Application (ZSERVICE) instance.

Select **Save as Default** to use the current settings as the default settings for the Application Level Event Reporting dialog box.

11 Click **Next**.

The Application Summary dialog box opens, confirming your selections and data entry in the New Application Wizard.



The Event Reporting summary at the bottom of this dialog box is an abbreviated specification that summarizes the options that you selected for event reporting in the Application Level Event Reporting dialog box. It consists of a series of **keyword=value** pairs, separated by commas. Each keyword identifies a reporting event as described in the following table.

Table 23 Event reporting keywords and their denotations

Keyword	Reporting Event
AI	Application Installation
AD	Application Deinstallation
AU	Application Update
AR	Application Repair
AV	Application Verification

Keyword	Reporting Event
VA	Version Activation
VD	Version Deactivation

The value associated with each keyword indicates which option button, if any, was selected for the event, as follows:

Table 24 Values associated with each event reporting keyword

Value	Selected Option Button
S	Success
F	Failure
B	Both (report success and failure)
N	None (no reporting for the event)

12 If you need to make changes, click **Previous** to return to an earlier dialog box and make your changes.

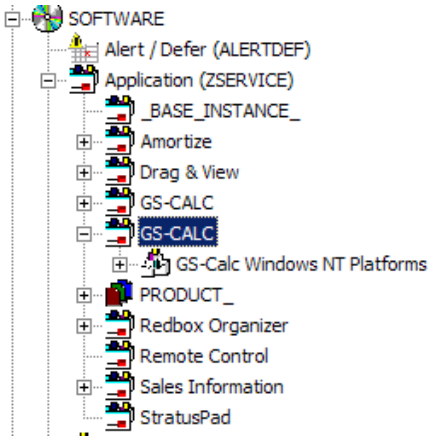
Otherwise, click **Finish** to create the service instance for the application and link it to the promoted package, then click **OK** in response to the message "Application successfully added."

The **GS-Calc Windows NT** package is now a ZSERVICE.

13 To refresh the Application (ZSERVICE) list in Tree View, right-click on the Application (ZSERVICE) entry of the SOFTWARE Domain. A shortcut menu opens.

14 Click **Refresh**.

The Application list refreshes, listing the new ZSERVICE. The ZSERVICE is ready to be connected to users for deployment.



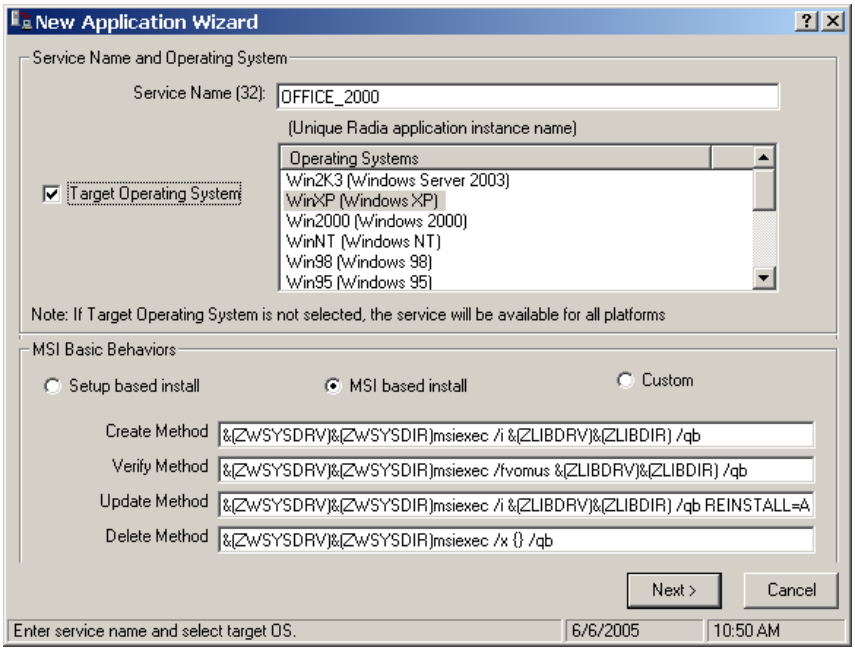
Defining MSI Basic Behaviors on the New Application Wizard

In contrast to Advanced Package Management, where CM provides on-going management of the components of an MSI package, Basic Package Management is used for packages that:

- Do not support administrative installation.
- Are simply to be installed and uninstalled by CM, but whose components are not to be managed by CM. That is, the CM agent is not to be instrumental in any component verification or repair capability.

The New Application Wizard permits you to define MSI Basic Behavior methods for packages created with the CM Packager for WI. These packages have a connection to the MSIBASIC Class in the CM CSDB.

Figure 29 MSI Basic Behaviors in New Application Wizard



Packages that cannot be administratively installed will default to Basic Package Management with the CM manageability features disabled. Use the MSI Basic Behaviors section of the New Installation Wizard dialog box to specify the install type and MSI native mode methods to be used for package management.

To define the MSI Basic Behaviors section of the New Application Wizard

1 Select the install type that reflects how the package is installed:

Setup based install
The package installs with a `SETUP.EXE` file that calls `MSIEXEC` to install the package. This will run the native `setup.exe` for the MSI application. If `setup.exe` supports any command line switches, such as **`/s`** for silent or ADDLOCAL=ALL for complete installation, you can add them to the Create Method. Not all MSI applications support the same command-line switches. For `setup.exe`, refer to the application's documentation for details.

MSI-based install
The package installs with MSIEXEC.

Custom
The package uses a custom install method.

2 Review and modify, if necessary, the native MSI commands and switches that are to be used for the Create, Verify, Update, and Delete Methods for basic package management.

The Version Group Editor

The **Version Group Editor** is a feature of CM Admin CSDB Editor when the CM Application Manager is installed. Use it to create and edit instances of the **Version Group** Class. Features of the Version Group Editor enable you to create, edit, and delete Version instances of the Version Group and control the Version Group's deployment.

Creating a Version Group Instance

The example in this section describes how to create an instance of the Version Group class that represents a set of versions of an application.

To create a Version Group instance for a package

1 From the Windows taskbar, click **Start** → **Programs** → **HP OVCM Administrator** → **CM Admin CSDB Editor**. The CM Admin CSDB Editor Security Information dialog box appears.

The factory set user ID is RAD_MAST. No password is necessary. Your CM security administrator might have changed this during installation. Check with him or her to obtain your own user ID and password, if necessary.

2 If necessary, type a user ID and password, then click **OK**.

The CM Admin CSDB Editor splash screen briefly appears.

3 Double-click **PRIMARY**.

4 Double-click **SOFTWARE**.

5 Right-click **Version Groups (VGROUP)** and select **New Instance** from the shortcut menu.

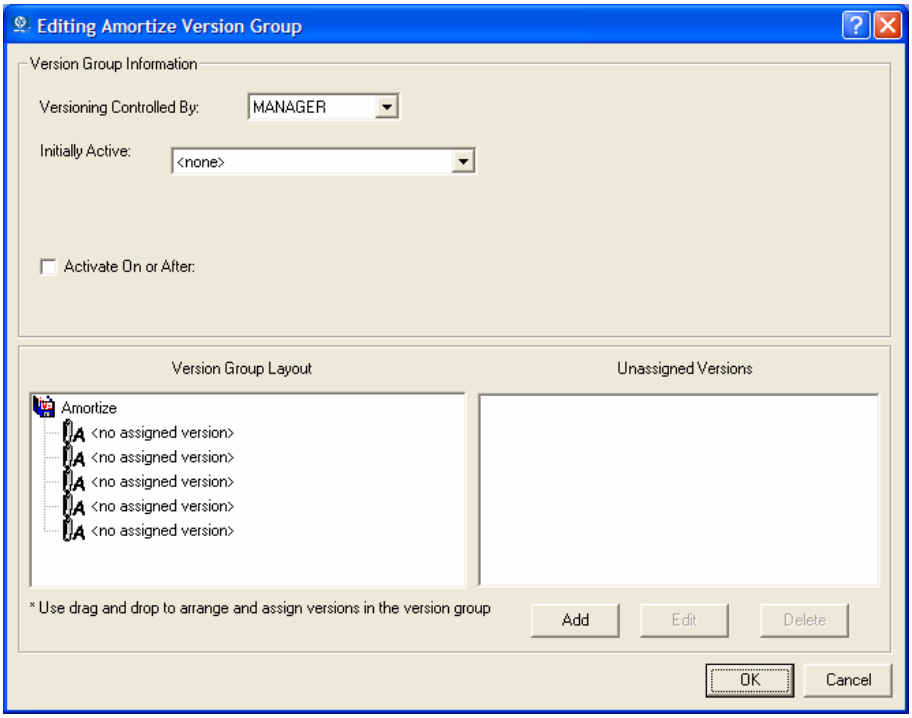
A Create Instance dialog box opens.

Type a friendly name for the new Version Group in the Create Instance dialog box, such as **`Amortize`**.

The friendly name will appear in the tree view of CM Admin CSDB Editor.

6 Click **OK**.

The Editing Version Group dialog box opens.



The Version Group Editor contains all the controls you need to manage the content and deployment of a Version Group.

7 Make selections on this dialog box based on the following explanations.

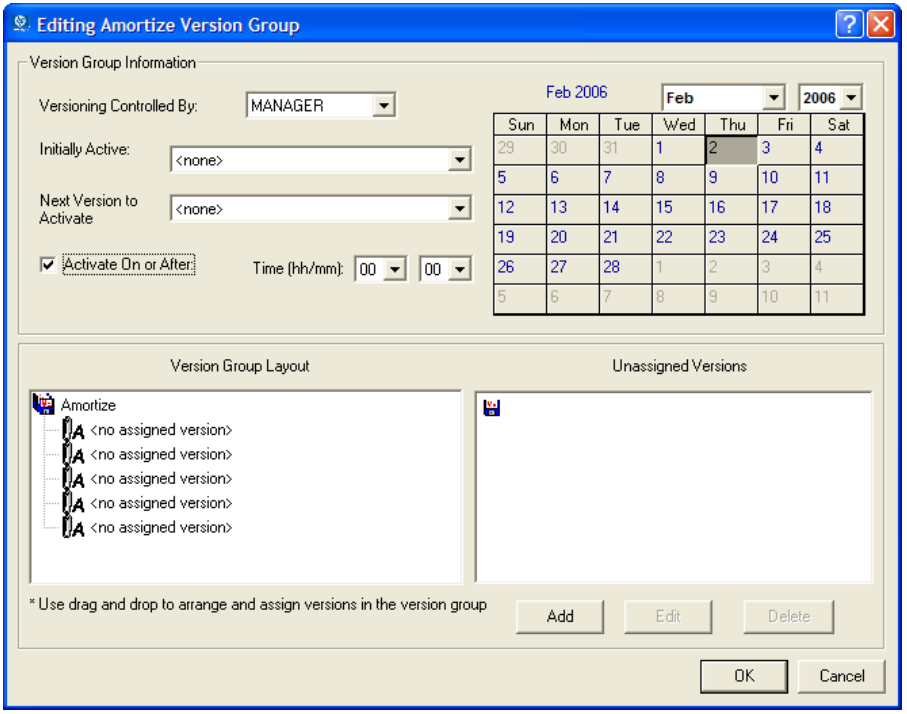
Versioning Controlled By drop-down list
determines whether the CM administrator (Manager) or end user (**Client**) chooses the version to deploy.

— Choose **MANAGER** to select the CM administrator. The CM Application Manager supports version control only by the CM administrator; it does not support version control by the end user.

— Choose **CLIENT** to select the end user. This will disable version deployment.

Initially Active drop-down list
selects which version to activate on the agent computer the next time the subscriber's CM agent connects to the CM Configuration Server. The available versions are those that appear in the Version Group Layout section of the Editing Version Group dialog box.

Activate On or After check box
provides access to additional controls that the CM administrator can use to prevent version activation prior to a specific date and time. When selected, the controls appear as follows:



Additional features are displayed when **Activate On or After** has been selected.

Next Version to Activate
indicates which version of the application to activate if the newly deployed version fails to install and activate successfully.

Calendar and Time
sets the date and time used for deployment. Use these functions to set the Timer.

If using the Version Group Editor to maintain a TIMER instance, consider the following:

- Timers are only created and connected to a VGROUP if versioning is controlled by the CM Configuration Server. This is done by selecting **MANAGER** on the Versioning Controlled By drop-down list in the Version Group Editor. In addition, with MANAGER selected, the **Activate On or After** check box must be selected. By selecting this check box, the Calendar and Time controls will become available.
- The calendar will not allow the administrator to select a date earlier than the current day.

For more information about Timers, refer to the *CM Application Manager Guide*.

The Version Group Layout and associated controls at the bottom of the window enable you to create and edit Version instances and assign them to the Version Group as needed.

8 Click **Add** at the bottom of this dialog box.

A Create Version dialog box opens.

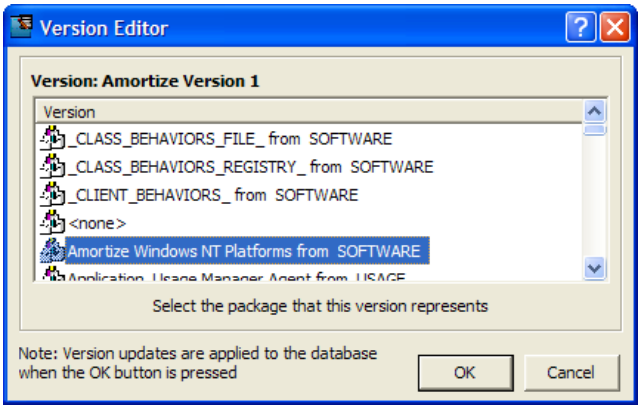
9 Enter a suffix for the friendly name for the new Version instance in the text box.

The friendly name for the Version instance will be a concatenation of the Version Group's friendly name and the suffix you enter here. In the example presented here, the friendly name will be Amortize Version 1.

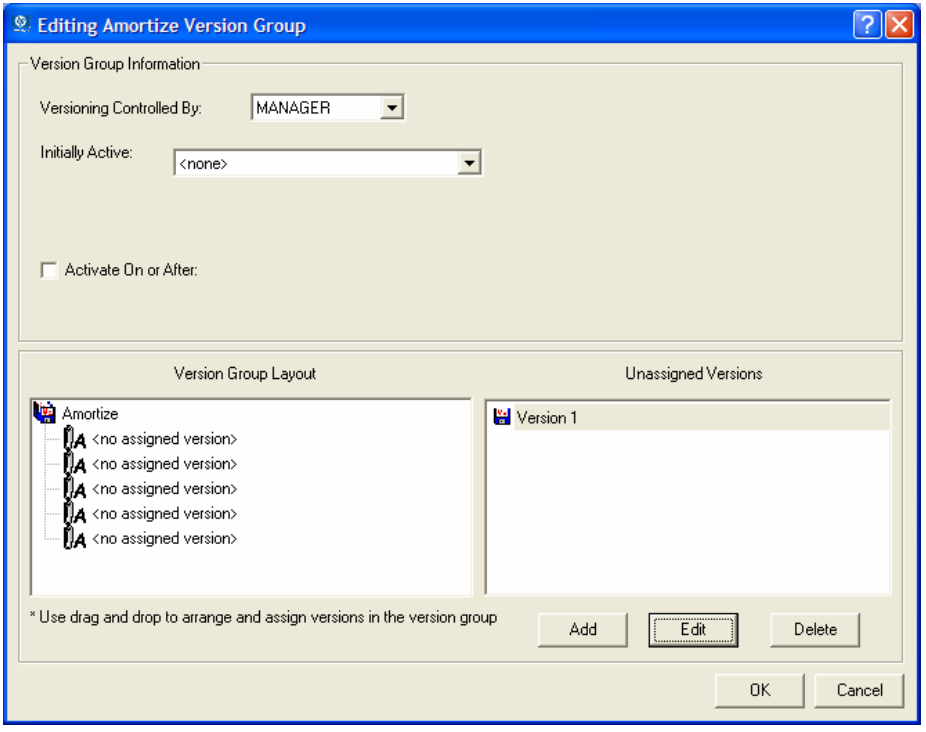
10 Click **OK**.

The Version Editor dialog box opens. It contains a list of Application Package instances stored in the CM-CSDB. Use this dialog box to associate the newly created Version instance with an Application Package instance.

11 Scroll down the list until the desired Application Package instance appears in the window. In this example, we will connect it to the Amortize 1 package instance. Select the package by double-clicking it, or single-clicking it and clicking **OK**.



The Version instance will be created, you will be returned to the Editing Version Group dialog box, and the version instance will appear in the Unassigned Versions list.



12 Repeat the process of adding version instances for each version of the application software for this Version Group.

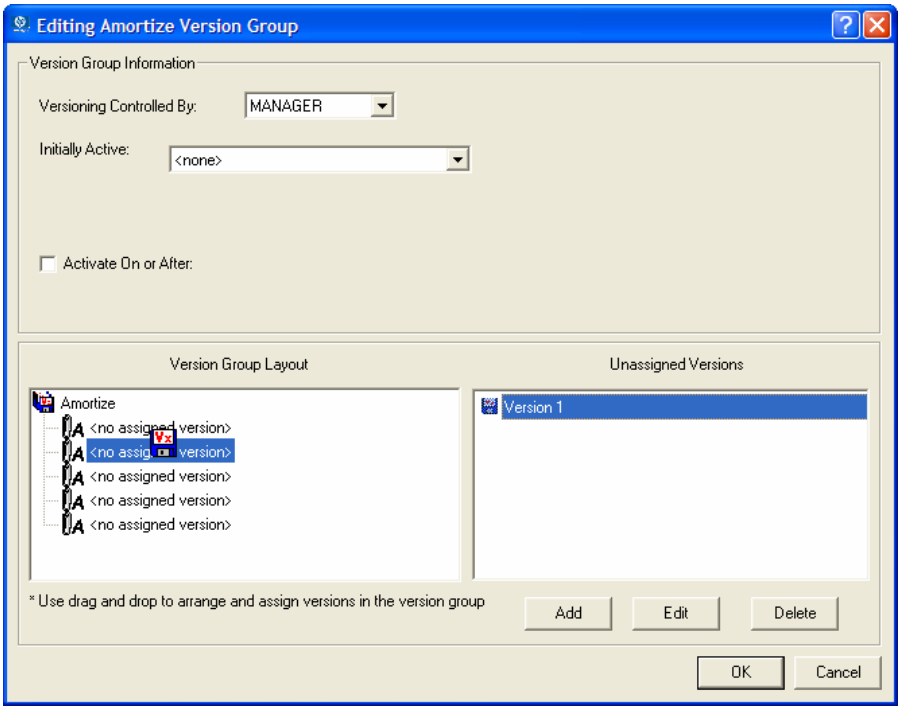
Version Group Assignments

The previous exercise showed how to create a version instance for a package. After the version is created, you will need to connect it to a Version Group.

To assign a Version Instance to a Version Group

1. Drag a Version instance icon from the Unassigned Versions list to one of the connections labeled <no assigned version> in the Version Group Layout.
2. Repeat this until all needed versions have been assigned.

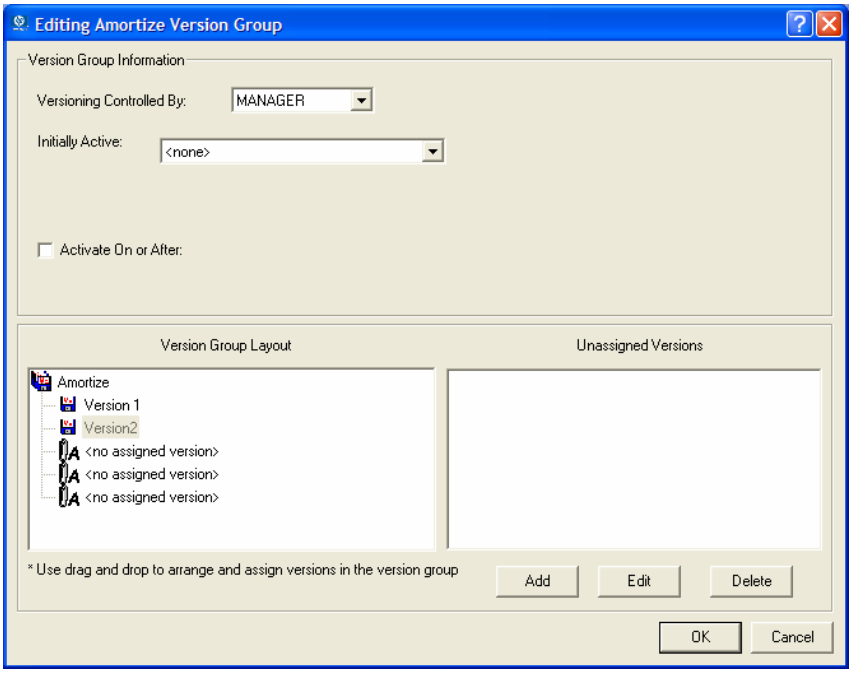
   The Editing Version Group window will look similar to:



3. Click the Initially Active drop-down arrow to display assigned versions. You can choose to deploy any of the assigned versions, or <none>.

   When a CM agent connects to the CM Configuration Server, the packages (including files, links, shortcuts, and registry entries) associated with all assigned versions are transferred (if they are not already present on the agent computer), in compressed form, to holding locations on the agent computer. The version you select in the Initially Active drop-down list is then installed and activated on the agent computer.

Select the version to deploy from the Initially Active drop-down list. It will be highlighted in the Version Group Layout list.



If the Initially Active control is set to <none>, the CM agent connection to the CM Configuration Server will transfer all assigned packages that have not yet been transferred, but none will actually be installed on the agent computer. This could be quite confusing for the end user, given the following scenario: Your end user launches the CM Application Self-service Manager. An agent connection takes place and the end user chooses to install a versioned application from the Service List. On the CM Application Self-service Manager user interface, the user will see the file download activity as the version packages are transferred in compressed form and stored in holding locations on his computer. The Service List will indicate that the application has been installed, but since the Initially Active control indicates <none>, no version will be activated. Again, this situation can be disconcerting to end users and result in unnecessary calls to your organization's support group.

You can avoid this with proper planning. If you need to deploy a versioned application without activating any version immediately, consider making one of the versions a stub application that simply displays a suitable message to the end user. For example, create a stub application whose job is to indicate that the application is not yet ready for installation. Instead of selecting <none> for the Initially Active

version, select the stub application as the initially deployed version until you are ready to deploy the real thing.

To remove a Version Assignment in a Version Group

- Drag-and-drop the version icon from the Version Group Layout to the Unassigned Versions list. The assignment will be removed.

To delete a Version Instance from a Version Group

1. Highlight the desired Version instance, and click **Delete**. The Version instance will move to the Unassigned Versions list (if it is not already there), and its icon will appear dimmed to indicate that it has been deleted.

   Deleted Version instances listed in the Unassigned Versions list will not actually be deleted from the Version Group until you close the Version Group Editor. While the Version Group Editor is still open, you can restore a deleted Version instance by clicking it, and then clicking **Un-delete.**

2. After the Version Group has been completely configured, click **OK** to close the window.

To edit a Version Group

1. Right-click the **Version Group** instance in the CM Admin CSDB Editor.

   A shortcut menu opens.

2. Select **Version Group Editor** from the menu, and continue as described above.

Service Optimization

The intent of Service Optimization is to maximize efficiency of distributing periodic updates to CM packages comprising a CM service. Later packages contain revisions and additions to files published in earlier packages.

Service Optimization uses **byte-level differencing** to create patches.

- **Byte-Level Differencing**
  Byte Level Differencing is the process of comparing two sets of data, at the byte level, and then identifying their differences. The differences

between the two packages are the bytes that are required to upgrade or fix an application.

- **Patch**
  A patch is a published package of data (bytes) that is required to upgrade or fix an application that is already published and installed on an agent computer. Since a patch contains only the differences between the packages, deployment of a patch takes less time and requires less network bandwidth. Patches should be built only for file changes that can be distributed more efficiently as patches than as replacement files.

Byte level differencing is supported in the CM Admin Packager, the CM CSDB, and the CM Admin CSDB Editor.

- **CM Admin Packager**
  The CM Admin Packager calculates the message-digest algorithm (**MD5**) signature of the file as it is being promoted. The MD5 algorithm can take input and create a 128-bit fingerprint (or message digest) of the input. This information is then stored, in compressed form, in the CM Configuration Server. For more information about MD5 signature, visit:

  www.faqs.org/rfcs/rfc1321.html

- **CM-CSDB**
  The PATCH Domain contains the list of available patches in a new PATCH Class. The file signatures of the patch are used to identify the associated instance that contains the data for the patch. For more information about the PATCH Domain, refer to the *CM Configuration Server Guide*.

  Prior to deploying patches to users, change the ZOBJPTCH variable from N to Y in the USER Class of the POLICY Domain.

Figure 30 PRIMARY File, PATCH Domain, PATCH Class

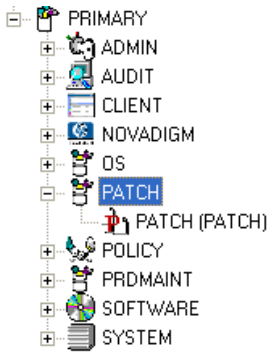


Table 25 below describes the PATCH Class attributes.

Table 25 PATCH Class attributes

Attribute	Description
NAME	The Friendly Name – the value of ZRSCCFIL in the To instance.
STATUS	The status of the patch. Valid values are Queued, Rejected, and Built.
MESSAGE	The explanation of the status. Reference this attribute to determine if a patch was built successfully.
PVERSION	The version of the patch builder.
POS	The operating system on which the patch was built.
ZRSCDATE	The date on which the patch was built.
ZRSCTIME	The time the patch was built.
ZRSCSIZE	The size of the patch.
ZCMPSIZE	The compressed size of the patch.
ZRSCSIG	The content-based signature of the patch type.
ZRSCTYPE	The type of resource. The only valid value is PATCH.
ZRSCCSTA	The status of patch resource. The only valid value is 999.
The controls used in patch generation:	
ZCONTRL1	ZSTOPPCT is the percentage of target data when the patch will stop building the path.
ZCONTRL2	ZSTOPMAX is the maximum size of the patch.

Attribute	Description
ZCONTRL3	ZTIMEOUT is the time allotted to patch creation before aborting the Create. See Table 27 on page 136.
SIGTYPE	Currently, MD5 is the only signature type supported.
FROMSIG	The content-based signature of the From file.
FROMSIZE	The size of the From file. This value is taken from the ZRSCSIZE of the From instance.
FROMFDCI	Original location of the From instance
FROMOID	Object ID of the original From instance.
TOSIG	The content-based signature of the To file.
TOSIZE	The size of the From file. This value is taken from the ZRSCSIZE of the To instance.
TOFDCI	The original location of the To instance.
TOOID	The object ID of the original To instance.

- **CM Admin CSDB Editor**
  Optimize Service is an option on the Application (ZSERVICE) instance shortcut menu. It allows the administrator to create, display, and delete patches for selected application components.

Using the Optimize Service Option

In order to create a patch using the Optimize Service option, you first need to promote your packages using the CM Admin Packager. For instructions on how to do this, refer to the *CM Application Manager Guide*.

Patches can be created for components that contain a signature only. Initially, only MD5 is supported.

Components being used for patching must be published from the same location, or computer, to qualify for byte-level differencing patching. This will populate the eight-byte CRC found in the suffix of the instance names.

Because the CM Admin Packager now calculates and uses the MD5 signature, and this signature is vital to the patching functionality, there is no backward compatibility to previously promoted components.

This section will provide you with detailed instructions for:

- Creating an Application (ZSERVICE) in the SOFTWARE Domain for packages for which you will create a patch.
- Creating the patches using the Optimize Service wizard.

We have provided you with detailed information about each of the screens that you will encounter.

In this example, you will optimize a series of text files called BLD. These files were packaged and promoted to the CM Configuration Server using the CM Admin Packager. Each text file is its own PACKAGE in the SOFTWARE Domain.

After you have promoted your packages, you will need to create a ZSERVICE instance.

In some cases, the difference in size between two packages is so minimal that creating a patch will actually increase the size of the file. In this case, the creation of the patch will not be allowed.

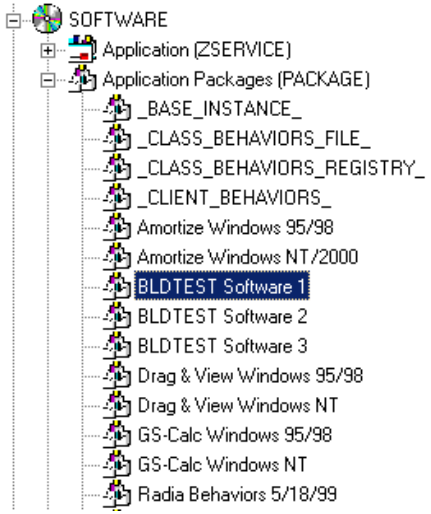
When publishing the files needed for this example, the following naming conventions were used:

- The file `bld1.txt` was originally published as `bld.txt`.
- When the second file was about to be published, it was renamed to `bld.txt`; the original `bld.txt` file was renamed `bld1.txt`.
- When the third file was about to be published, it was renamed `bld.txt`; the second file was renamed `bld2.txt`.

By using the same file name and the same machine during publishing, you are assuring the same CRC needed to correctly create patches.

To configure the Optimize Service feature

1 Expand the tree view to the PRIMARY.SOFTWARE.PACKAGE Class and locate the BLDTEST packages.



2 Right-click on the **Application (ZSERVICE)** Class.

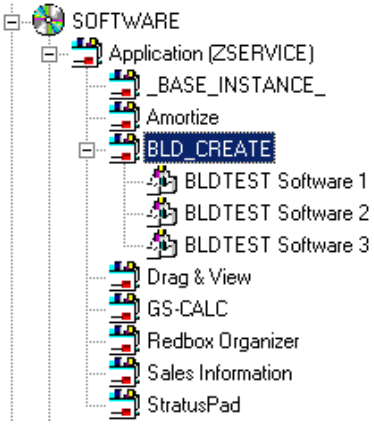
A shortcut menu opens.

3 From the shortcut menu, select **New Instance**.

4 Create a new instance named BLD_CREATE, and click **OK**.

The BLD_CREATE service is added to the CM CSDB.

5 Drag-and-drop the BLDTEST1, BLDTEST2, and BLDTEST3 packages to the BLD_CREATE instance in the Application (ZSERVICE) Class.



After you have created the Application (ZSERVICE) containing the Application Packages (PACKAGE) you can begin to create your patches.

Just as you create a plan or policy for users, you might want to develop a plan for creating, editing, and deleting patches. Some important things to remember when creating patches:

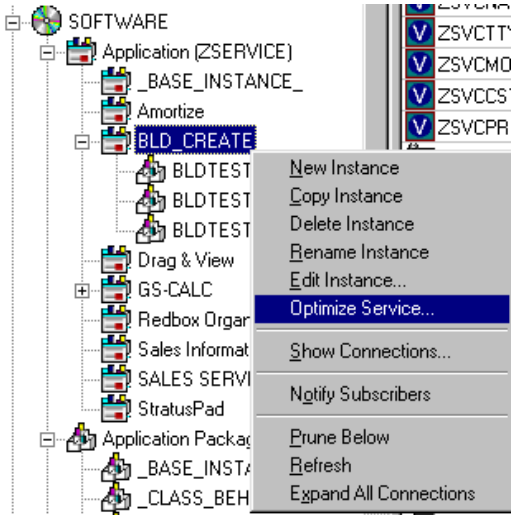
— It is important to devise a means by which to document the packages that will create your patches. When building different patches within a small database, it is relatively easy to follow the schema of connected paths that create a patch. As your database grows, it is possible for the patch class, and the packages that they represent, to grow unwieldy.

— Only those packages that contain an MD5 signature are eligible for patching.

— Patching is different from *versioning*. If a patch fails to deploy properly, there is no previous version option.

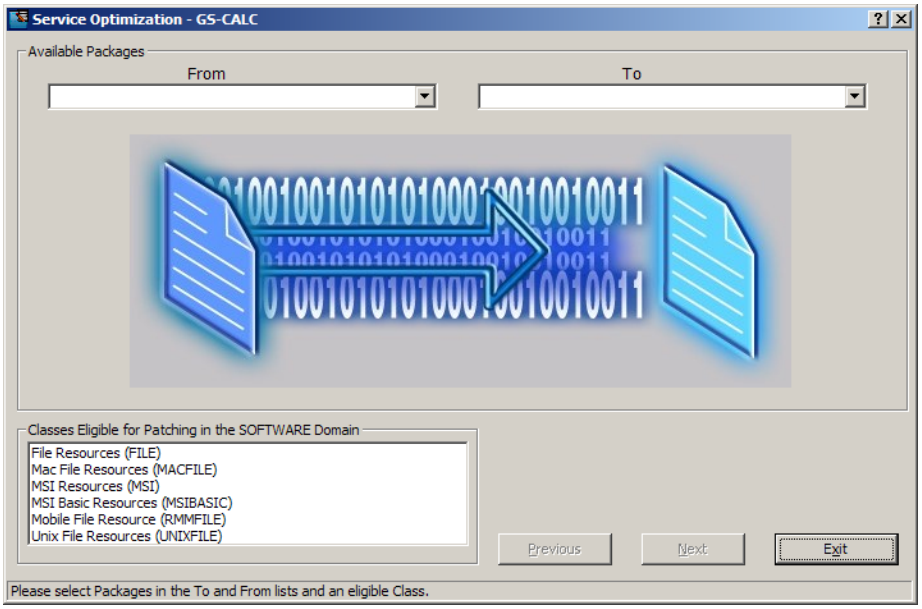
6 Right-click the Application (ZSERVICE) Class **BLD_CREATE**.

A shortcut menu opens.



7 Select **Optimize Service**.

The Service Optimization wizard opens.

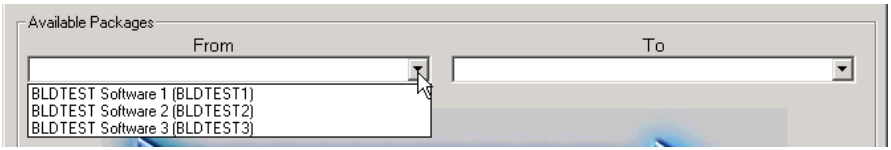


There are three areas in the first screen of the Service Optimization wizard:

- **Available Packages From**
  This lists the available packages for the service you are trying to optimize. When you click the **From** drop-down arrow, all of the packages for the Application (ZSERVICE) are displayed. In this example, click the **From** list to show all of the packages in the BLD_CREATE Application (ZSERVICE).
- **Available Packages To**
  After you have selected the From package, the To list becomes populated with those packages available for patching in the service you are optimizing.
- **Classes Eligible for Patching in the SOFTWARE Domain**
  Select the appropriate class eligible for patching within the SOFTWARE Domain. Available classes are File Resources (FILE), MAC File Resources (MACFILE), MSI Resources (MSI), MSI Basic Resources (MSIBASIC), Mobile File Resources (RMMFILE), and/or Unix File Resources.

8 Click the **Available Packages From** drop-down arrow.

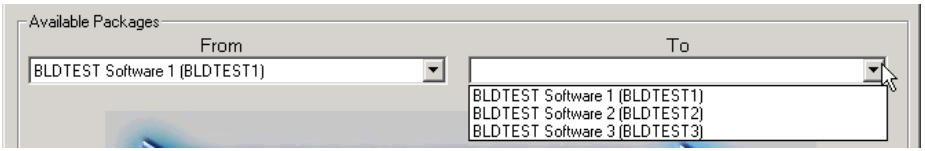
The packages that are available for the service you selected in step 6 (on page 132), BLD_CREATE, are listed.



9 From the drop-down list select **BLDTEST Software 1 (BLDTEST1)**.

The Available Packages To list is populated with the packages that are available for patching in the BLD_CREATE service.

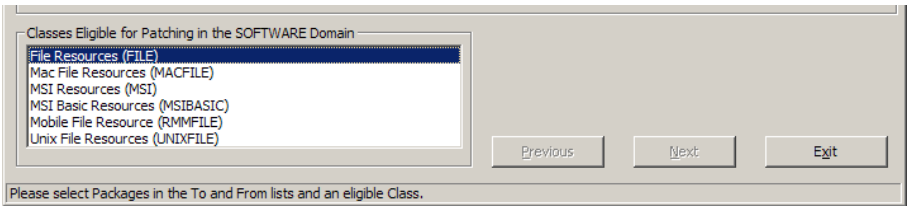
10 From the Available Packages To drop-down list select **BLDTEST Software 2 (BLDTEST2)**.



11 From Classes Eligible for Patching in the SOFTWARE Domain select the appropriate class for patching in the SOFTWARE Domain. Choose the class in which the file resources for the package are located.

Currently, File Resources (FILE), MAC File Resources (MACFILE), MSI Resources (MSI), MSI Basic Resources (MSIBASIC), Mobile File Resources (RMMFILE), and Unix File Resources (UNIXFILE) qualify for patch building.

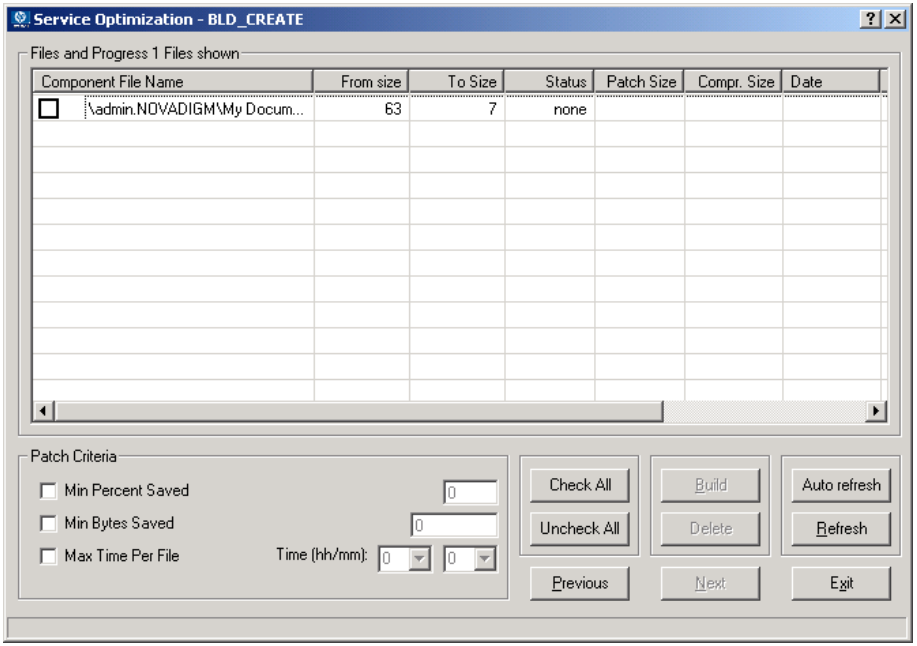
In this example, the only class that is available to be patched is File Resource (FILE).



To determine which classes are available for patching within your package, expand your Application (ZSERVICE) instance to the attribute level. Examine your instances to determine which resources are available for patching.

12 Select **File Resource (FILE)** and click **Next** to continue.

The Files and Progress dialog box opens.



This dialog box displays files and progress information. The Service Optimization wizard gathers the instance information based on the selections made in the Available Packages dialog box.

Table 26 below shows the options in the Files and Progress section of this dialog box. And Table 27 on page 136 shows the options in the Patch Criteria area.

Table 26 Files and Progress information

Column	Description
Component File Name	The name of the patch that is being created.
From Size	The original resource-package size.
To Size	The original target-resource size.
Status	The status of the package. The valid values are none, queued, rejected, active, and built.
Patch Size	The size difference between the original package data (BLD1) and the changed package data (BLD2).
Compressed Size	The compressed size of the package listed in the From Size.
Date	The date on which the package was published and promoted.

Column	Description
Size Limit	This information is pulled from the Patch Criteria, Min Percent Saved field. This field will become populated after the Patch has been built or upon termination of the patch building process.
% Limit	This information is pulled from the Patch Criteria, Min Bytes Saved field. This field will become populated after the Patch has been built or upon termination of the patch building process.
Time Limit	This information is pulled from the Patch Criteria, Max Time Per File field. This field will become populated after the Patch has been built or upon termination of the patch building process.

It is possible to place patch-building restrictions, such as percent size, bytes, and time elements by configuring the Patch Criteria options. You can control the creation of patches by setting criteria that will apply to all files whose check boxes are currently marked in the file list.

Suppose that the number of files in the list is large and you do not want to evaluate each file manually to determine if you want to build a patch for it. Click **Check All** to mark the check boxes of all of the files in the list. Then, use the Patch Criteria options to limit the building of patches to those files likely to produce a benefit.

Table 27 below details the options of the Patch Criteria area. The default for these three options is an empty (unselected) check box.

Table 27 Patch Criteria information

File Option	Description
Min Percent Saved	Specify the minimum percent of the To component to save. Otherwise, the patch will not be built.
Min Bytes Saved	Specify the minimum number of bytes that the patch must save. Otherwise, the patch will not be built.
Max Time per File	Specify the maximum processing time for the patch. Use the drop-down list to indicate the maximum time allowable for patch building. The time can be set in minutes, or hours and minutes.

After a component file has been selected, the buttons used for patch building are enabled.

To apply a criterion, select its check box. The adjacent data entry fields become enabled. Enter the criteria into the associated data fields. The

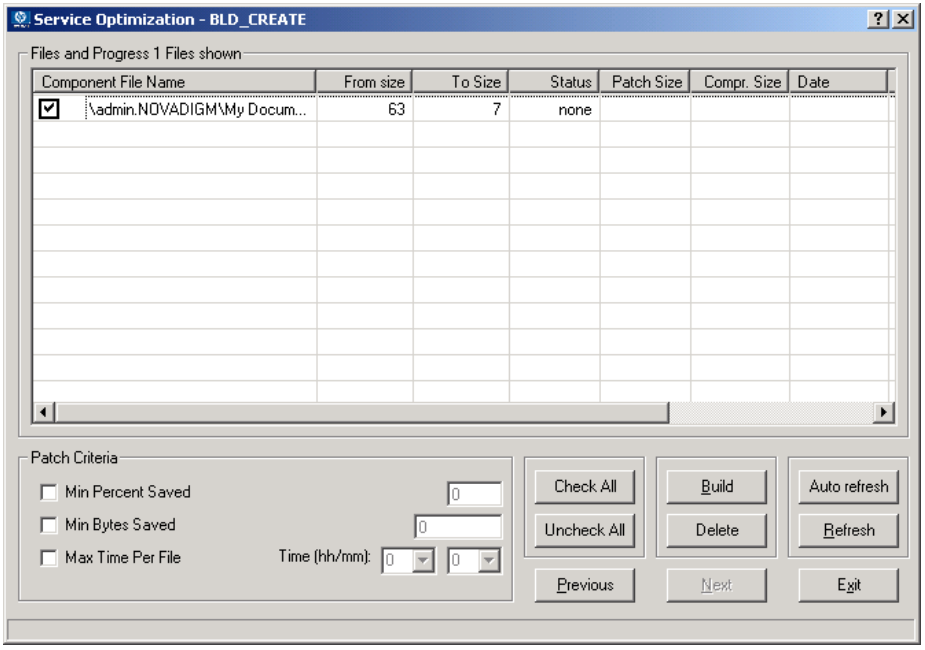
criteria for the patch are applied in combination. In order for a patch to be built for a selected file, it must meet all the criteria you apply.

Table 28 below explains the functions of the Patch Criteria buttons.

Table 28 Patch Criteria button functions

Button	Action
Check All	Selects all of the files that are listed in the Component File Name column.
Build	Click here to begin building patches.
Auto Refresh	Select **Auto refresh** to open the Auto Refresh Timer dialog box. When you use Auto Refresh, type a refresh interval between 1 and 99 minutes. You can cancel the auto refresh by typing 0.
Uncheck All	De-selects all files that are selected in the Component File Name column.
Delete	Deletes a selected component file.
Refresh	Refreshes build status.
Previous	Return to the previous screen.
Next	When enabled, click **Next** to proceed to the next screen.
Exit	Click **Exit** to exit the Optimization Wizard.

13 Select the check box next to the files to be patched.



14 Click **Build**.

The CM Admin CSDB Editor will prompt you to confirm the new patch build.

15 Click **Yes** to continue.

or

Click **No** to exit.

When you click **Yes,** any current patches of the pair will be deleted. A patch-creation request for each selected file pair will then be sent to the CM Configuration Server.

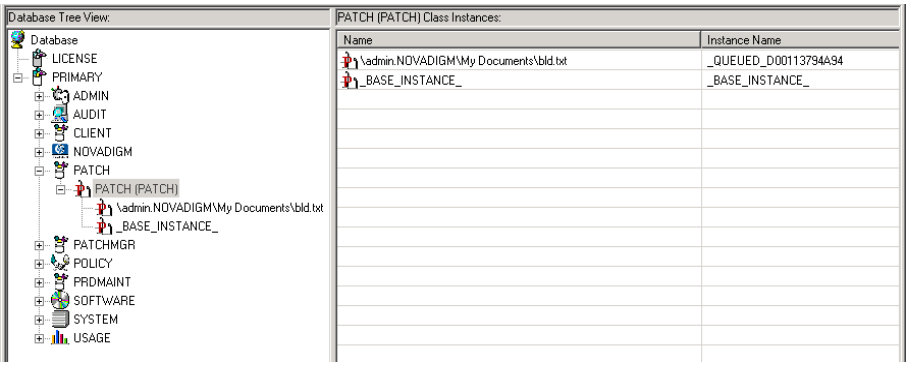
16 To update the Status, click **Refresh**.

17 Click **Exit** to exit the Service Optimization wizard.

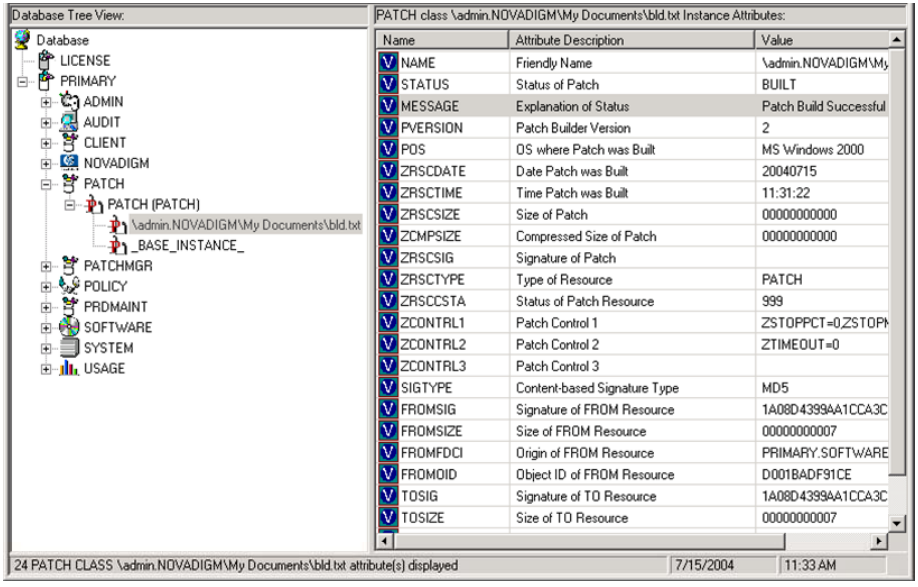
or

Click **Previous** to return to the Available Packages dialog box to build more patches.

The patch is located in the PATCH Domain, under the PATCH Class.



Examine PATCH.PATCH and verify that the patch instances are built. Examine rejected patches to determine why they were rejected. See the MESSAGE variable to determine the current status of requested patches.



CM Service Groups

CM manages products that require more than one service package to establish full product installation or operation. You can use **Service Groups** when a product requires other service packages or has dependencies on other services. This includes MSI-packaged products where:

- A product can use more than one MSI service package, for example where MS Office requires other language pack services.
- A large product, MS Office for example, might need to be split into smaller sub-services to install only specific parts of the product suite.

Service grouping will allow multiple services to be grouped by a higher, or **master**, service. This action allows administrators to logically group together similar or related components into one easy to understand collection.

Use the CM Admin CSDB Editor to define and then group services and packages to form the master service.

Remember to target actions, such as notifies or timer updates, to the master service. Any action applied to the master service is applied to all sub-services.

Component Enhancements

- **The Application Self-service Manager**
  Each service within the CM Service List will display only the master service.
- **The Application Manager**
  The RADSKMAN command contains the logic to process multiple service requests. It will perform the request on each of the individual sub-services and report the final results back to the master service.

Prerequisites

Prior to configuring a master service for a service group, change or confirm the following settings:

- CM Configuration Server (*versions 4.5 and later*), in the `edmprof` file, verify that ALLOW_DUPLICATE_INSTANCES in the MGR_OBJECT_RESOLUTION section is set to No.
- CM Configuration Server (*versions prior to 4.5*), in the `edmprof` file, verify that in the MGR_OBJECT_RESOLUTION section:
  - ALLOW_CIRCULAR_REFERENCE is set to Yes and,
  - ALLOW_DUPLICATE_INSTANCES is set to No.
- CM Configuration Server DB (CM-CSDB)

To configure a set of services (a grouping in which services depend on one another for installation), add a variable to the _BASE_INSTANCE_ of the Application (ZSERVICE) Class. This is a variable called ZSVCGRP. If you would like to add this variable to the ZSERVICE Class, use the following table to help you set the variable.

For information on editing a class, see Editing a Class on page 57.

Table 29 ZSVCGRP variable values

Value	Behavior
Y	Specify Y (Yes) for the CM Configuration Server to process the service groups as if they are unrelated applications or applications that are not dependent on one another for installation. If this variable is not present, this is the default.
D	Specify D (Dependent) for the CM Configuration Server to process the services under the service group as though they are dependent on one another for installation. If one sub-service fails, the remaining services will not be installed.
N	Specify N (No) for no service groups.

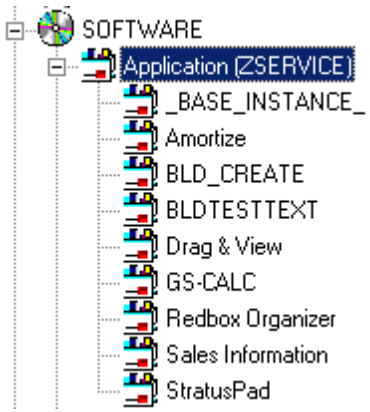
Creating a Master Service for a Service Group

After the changes have been made to the `edmprof` file, you can begin creating a master service instance for a service group.

Use the CM Admin CSDB Editor to create a new Application (ZSERVICE) instance for the master service. Next, for each service that is to be a member of the group, make a connection between the member service and the master service. You can add service members to the master service at any time.

For our example, we will create a master service instance called Sales Services. This master service will contain the Amortize and GS-Calc services that have been provided in the sample CM-CSDB.

Figure 31 Sample services in the ZSERVICE Class



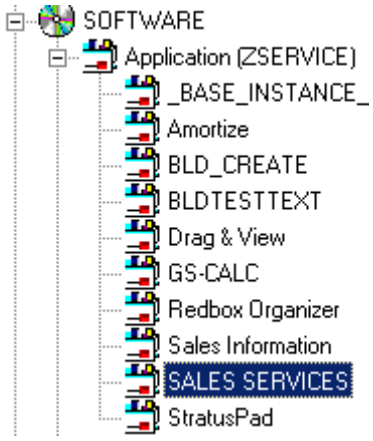
To create a Master Service for a Service Group

1. Open the CM Admin CSDB Editor and navigate to the **PRIMARY.SOFTWARE.ZSERVICE** Class.
2. Expand the class to display the Application (ZSERVICE) instances.
3. Right-click **Application (ZSERVICE)**.

   A shortcut menu opens.
4. Select **New Instance** from the shortcut menu.

   The Create Instance dialog box opens.
5. Enter the name of the new ZSERVICE instance (SALES SERVICES).
6. Click **OK** to create the instance.

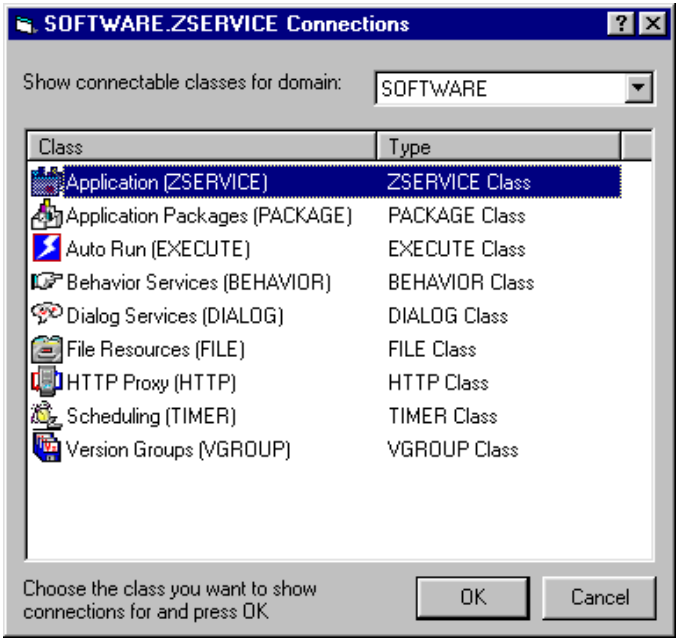
   The new instance appears in the Application (ZSERVICE) Class.



Now you can add services to the master service.

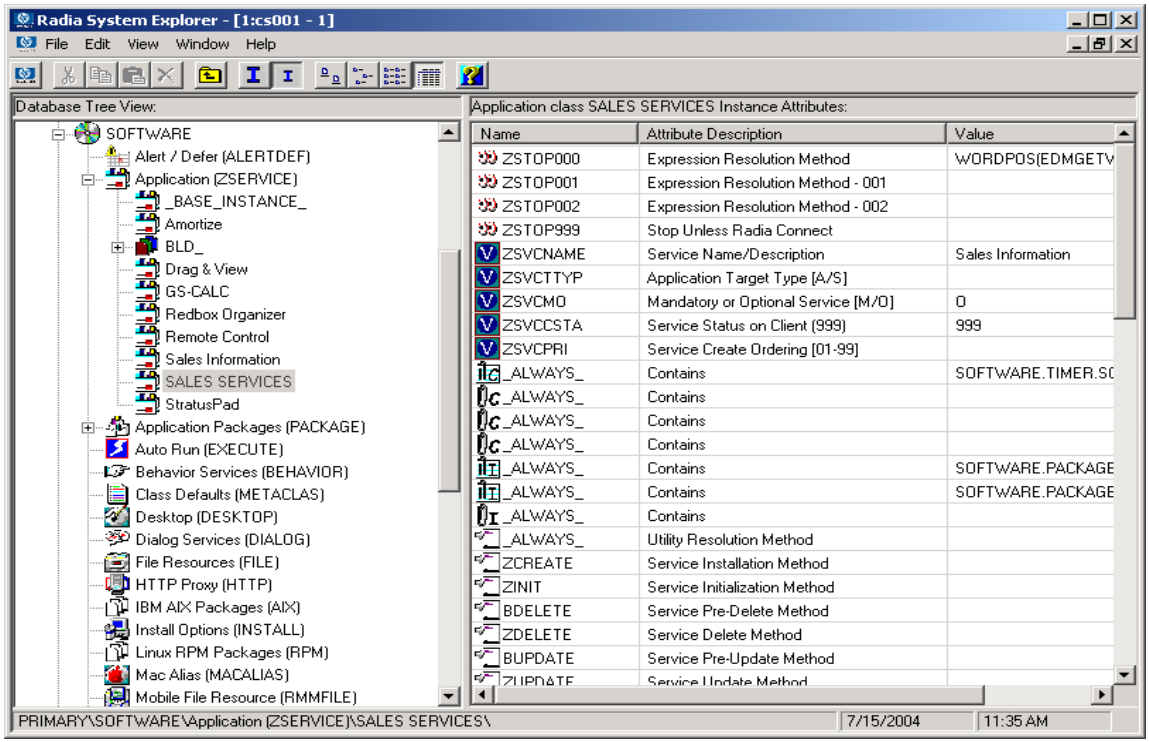
7 Right-click **SALES SERVICES**, and select **Show Connections** from the shortcut menu.

The SOFTWARE.ZSERVICE Connections dialog box opens.



8 Select the **Application (ZSERVICE)** Class from the SOFTWARE.ZSERVICE connections dialog box, and click **OK**.

The ZSERVICE Class instances are displayed on the list view of the screen.



9 Select the services from the right, and drag them to the ZSERVICE.SALES SERVICES instance.

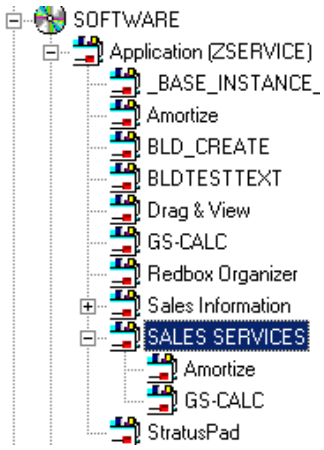
10 Drop them onto the service when the cursor changes into a paper clip.

11 Confirm the connection.

12 Continue adding services, following the steps that are outlined above.

The master service instance for this service group is now complete.

The services grouped into a master service will install from top-down according to the order in which they are connected to the master service instance in the database. Member services in a master service group do not honor the settings in ZSVCPRI.



This master service is now ready to be deployed to your users.

CM Notify

CM Notify enables the CM administrator to cause one or more agent computers to connect to the CM Configuration Server and to update or remove a specified application. It can also send e-mail notification to agent computers, based on the applications that CM manages for them.

For detailed information about CM Notify requirements, refer to the "Deploying Applications" chapter of the *CM Application Manager Guide*.

Notify communicates with agent computers that are members of an audience list. An agent computer is added to the audience list when CM installs an application to that computer.

There are a number of prerequisites for using Notify:

- The subscriber must be connected to the CM Configuration Server prior to the notification. Notify is designed to notify only agents whose information is in the PROFILE File in the CM CSDB.
- The subscriber must have already installed the application using CM in order to appear in the audience list for the specified application.
- If choosing to use e-mail to notify subscribers, be sure that the e-mail address for the subscriber is in the EMAIL variable of the USER Instance in the POLICY Domain, USER Class.

- The CM Configuration Server must be properly configured for Notify. For information on setting the appropriate values for the Configuration Server Settings file, `edmprof`, refer to the Engineering Note, *OV-ENKB01129: Configuring the Radia Configuration Server for E-mail Notification* on the HP OpenView support web site.
- Select how the CM Admin CSDB Editor should generate the Notify command for agents from the Notify tab of the Options dialog box. You can customize the UID and STARTDIR values when notifying subscribers to install an application. For details, see the Notify Tab on page 39.

The CM Admin CSDB Editor offers two ways to initiate a Notify action:

- Drag a POLICY Class instance (USER, DEPT, and WORKGRP) to an Application (ZSERVICE) instance. The audience list will consist of all agent computers that are members of the POLICY Class instance and for which CM has installed and manages the application represented by the Application Instance.
- Select **Notify Subscribers** from the shortcut menu associated with an Application instance. The audience list will consist of agent computers for which CM has installed and manages the application represented by the Application instance.

In either case, the application must already be installed on the agent computer before that computer is eligible to be notified.

After the Notify is initiated, CM Admin CSDB Editor presents the same dialog boxes to complete the notification process, regardless of how the Notify was initiated.

To create a notification event

1. Open the CM Admin CSDB Editor and navigate to the PRIMARY.SOFTWARE.ZSERVICE Class.
2. Expand the Application (ZSERVICE) Class.
3. Navigate to and expand the PRIMARY.POLICY Class.
4. Initiate a notify action using a drag-and-drop operation or a menu-based selection.

   Drag-and-drop Notify
   To initiate a drag-and-drop Notify, select a POLICY instance (workgroup, department, or individual user), drag-and-drop it onto an Application instance.

Menu-based Notify
To initiate a menu-based Notify, right-click on an Application instance. A shortcut menu opens.

5 Select **Notify Subscribers**.

Notify Dialog Boxes

Regardless of how the Notify was initiated (drag-and-drop or menu-based) the following process takes place.

The Notify action takes the set of all subscribers to an application from the PROFILE File as the starting point for creating a notify audience. The Notify Start message opens.

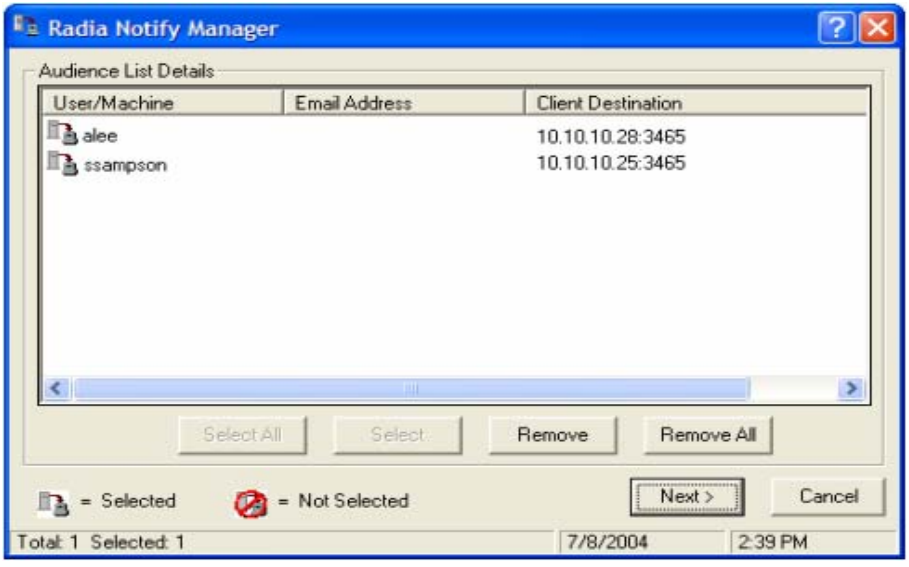
1 Select **Yes** to continue.

or

Select **No** to cancel the Notify.

The Notify action retrieves the set of the subscribers from the POLICY Domain. If the selected application does not have any subscribers, an Audience List Error message opens telling you that there are no users/machines in the audience.

The CM Notify Manager dialog box opens.



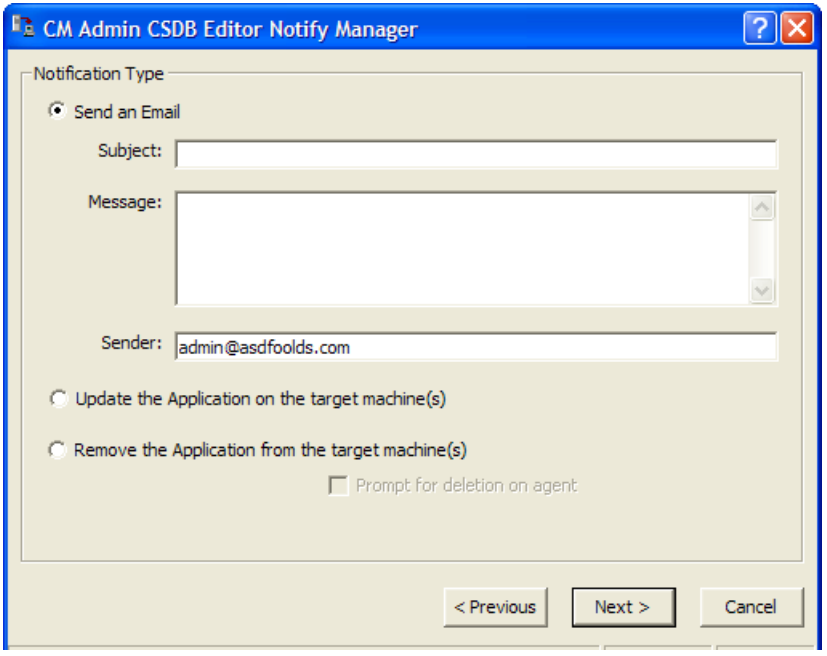
2 To notify the entire audience set, click **Select All**. This is the default when the audience list is first displayed.

To generate a tailored Notify audience, highlight the row associated with a subscriber, and choose either **Select** or **Remove**. Before using Select, you can click **Remove All** to clear the *selected* status of all agent computers in the audience list.

The icons to the left of the subscriber name and in the lower left corner of the dialog box denote which subscribers have been selected. You can also see the total number of agent computers in the audience list, and the number currently selected, displayed at the very bottom left of the dialog box.

3 When you are finished selecting agent computers in the Notify audience list, click **Next** to continue.

You can now select the type of Notify action. There are three options as shown in the following dialog box.



- **Send an Email**
  Use this option to inform subscribers of an application's status or non-mandatory availability. The subscriber's e-mail attribute in the user instance must contain a valid entry.

The @ symbol is required as part of the sender's address. An error will result if this symbol is not entered.

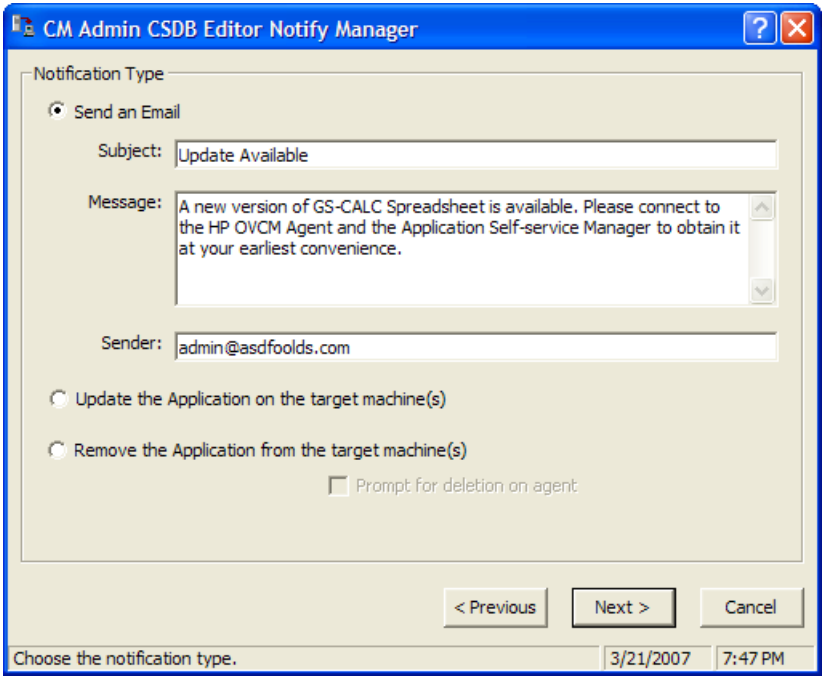
- **Update the Application on the target machine(s)**
  Use this option to install updates or new versions of an application. The updates or new versions must reside in the CM-CSDB.
- **Remove the Application from the target machine(s)**
  Use this option to remove an application from the specified desktops.

  The action you choose will be taken for all members of the audience list that you selected.

Send an E-mail Using Notify

If you choose **Send an Email**, and fill in the text boxes, the dialog box will resemble the following:xxxx

Figure 32 Send an Email option



Update an Application Using Notify

To update the application on selected agent computers in the audience list, choose **Update the Application on the target machine(s)**.

Remove an Application Using Notify

To remove the application from the selected agent computers in the audience list, choose **Remove the Application from the target machine(s)**.

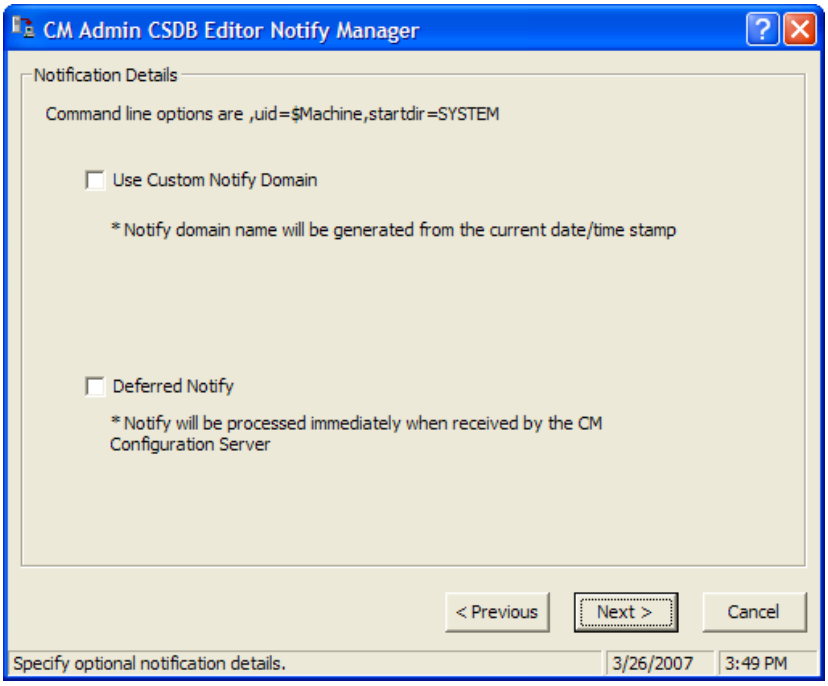
When you choose this option, the **Prompt for deletion on agent** check box is enabled. Normally, Notify will remove the application from the selected agent computers in the audience list without requesting permission from the end user. This allows unattended application removal from agent computers.

To require the end user to give permission, select this check box. A dialog box will open on the agent computer requesting permission to remove the application. If the end user agrees, CM removes the application. If the end user denies permission, the application remains intact on the agent computer.

- After you finish composing the e-mail or selecting the notification type, click **Next** to continue.

  The Notification Details dialog box opens.

Figure 33 Notification Details dialog box



Set Notification Details

The top line on the Notification Details dialog box identifies the Notify command line options for UID and STARTDIR that will be used. The Notify tab of the CM Admin CSDB Editor Options dialog box allows you to customize these values.

To reset the Notify command values for UID or STARTDIR, cancel out of this dialog box, set the Notify options for UID and STARTDIR using the Options menu, and then repeat the Notify Subscribers selections. See Notify Tab on page 39 for more information.

The Notification Details dialog box allows you to specify a name for the domain in the NOTIFY File that will contain the results of the current Notify operation. It also lets you specify when the Notify operation will take place.

You can choose to leave the Notification Details check boxes empty. This will cause the (following) default settings to be used.

- The Notify domain name will be generated from the current date/time stamp.

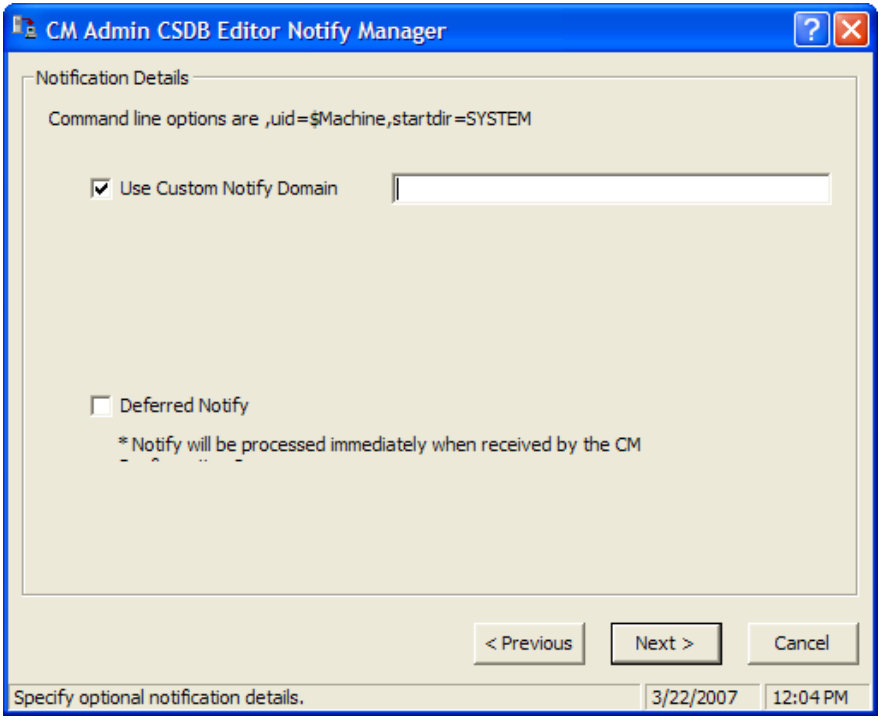
- The deferred notify date/time will be processed immediately when received by the CM Configuration Server.

To specify a domain name

1 Select the **Use Custom Notify Domain** check box.

A text box opens.

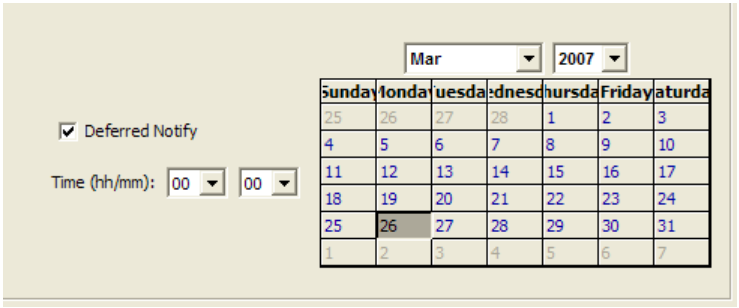
2 Type a domain name.



You might want to do this if you rarely use the Notify feature and you would like your domain names in the NOTIFY File to suggest the purpose of the Notify operation. See Notify File Structure on page 154 for additional information.

To defer notification until a specific date and time

1 Select the **Deferred Notify** check box in the Notification Details dialog box. Additional controls will appear as shown below.



2 Enter the date and time when the Notify operation should occur. The CM Configuration Server must be running in order for the Notify operation to occur. If not, the Notify operation occurs the next time the CM Configuration Server starts, after the designated date and time.

3 Click **Next**.

A summary screen opens.

4 Click **Finish** to start the Notify.

The CM Configuration Server schedules the Notify and generates an object in the NOTIFY File using the date and time of the Notify action. The format of the object name is YYYY_MM_DD_HH_MM_SS.

5 To view the status of the Notify, click **Yes**.

A Notify Status window opens. It provides a snapshot of the progress of the Notify process.

6 Select **Refresh** to update the Notify Status window with the latest information. It could take some time for all selected members of the audience list to be notified successfully or unsuccessfully.

The information provided can be used to correct, reconfigure, and regenerate the Notify request so that all recipients receive the notification. Notify information is also written to the CM Configuration Server log and can be viewed there.

After a Notify has been initiated for the first time, the NOTIFY File will thereafter appear among the files of the CM-CSDB in the CM Admin CSDB Editor tree view.

Use CM Admin CSDB Editor to examine the domains, classes, and instances of the NOTIFY File.

7 Right-click a domain in the NOTIFY File. A shortcut menu opens.

8 Click **Status Display** to view the status of the associated Notify operation.

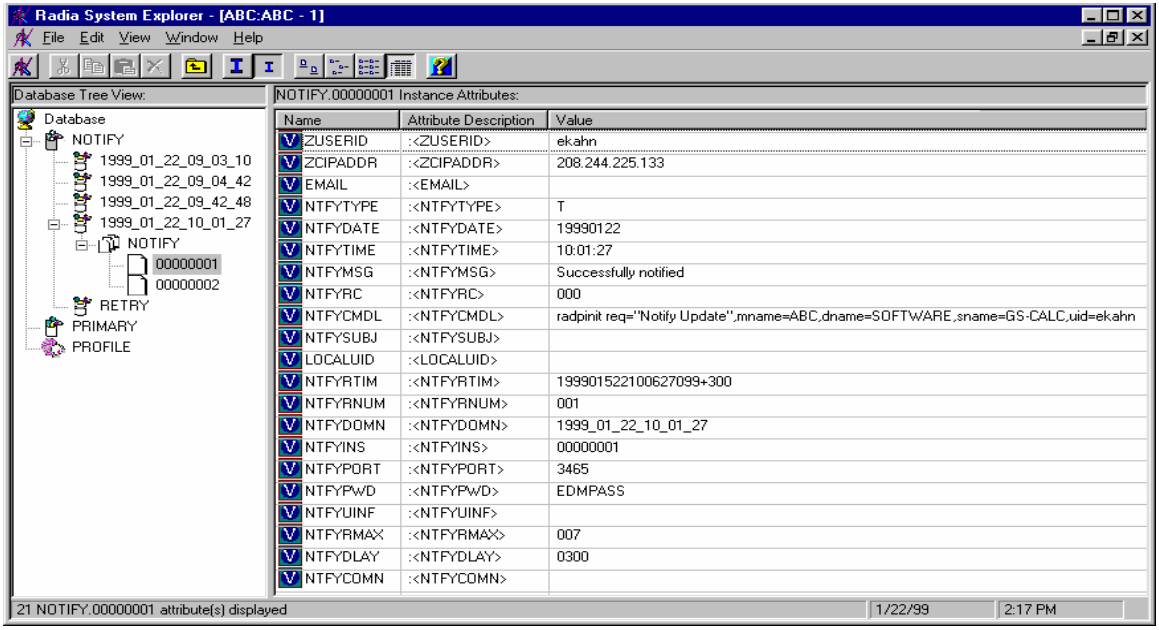
or

Click **Status Delete** to remove from the CM-CSDB the status information that is associated with the Notify operation.

Notify File Structure

The NOTIFY File is divided into domains, where each domain is one Notify operation. The format of the domain name is YYYY_MM_DD_HH_MM_SS, which represents the date and time when the Notify operation was initiated. For example, Figure 34 below shows an opened NOTIFY Domain with the name 1999_01_22_10_01_27. This domain holds the results of a Notify operation that was initiated on January 22, 1999 at 10:01:27.

Figure 34 Sample NOTIFY domain entries following an initial notification



Each NOTIFY File domain has one NOTIFY Class. Each NOTIFY Class contains an instance for each selected member of that Notify operation's

audience list. The instances are named with eight-digit numbers starting with 00000001 and running sequentially up to the total number of selected members in that Notify operation's audience list.

Each instance contains attributes that identify the selected audience list member, the kind of Notify operation (see the req= parameter of the command line in the NTFYCMDL attribute of Figure 34 on page 154), and the results of the Notify operation with respect to that member (see the NTFYMSG and NTFYRC attributes in Figure 34).

Retrying a Notify Operation

Often, not every selected audience list member can be notified successfully when the Notify operation is initiated. At that time, one or more selected agent computers might be turned off, might not be running the CM Notify module, might not be accessible via the normal communication channel, or might be broken, etc. An unsuccessful Notify attempt creates an instance in the RETRY Domain of the NOTIFY File.

CM can automatically retry the Notify operation for failed Notify attempts. To do so, the CM Configuration Server must be started with the Notify Retry Manager (the ZRTRYMGR module), as indicated by the last line in the following excerpt from the CM Configuration Server `edmprof` file.

```
[MGR_ATTACH_LIST]
ATTACH_LIST_SLOTS = 15
RESTART_LIMIT = 7
CMD_LINE=(zutilmgr) RESTART=YES
CMD_LINE=(zrexxmgr) RESTART=YES
CMD_LINE=(zsnmpmgr) RESTART=YES
CMD_LINE=(zsmtrmgr) RESTART=YES
CMD_LINE=(zsmtsmgr) RESTART=YES
CMD_LINE=(zntytmgr) RESTART=YES
CMD_LINE=(ztcpmgr)  RESTART=YES
CMD_LINE=(zrtrymgr) RESTART=YES
```

The Notify Retry Manager periodically examines the NOTIFY File's RETRY Domain, and re-attempts the Notify operation for each instance it finds there.

To see the status of retry attempts, use CM Admin CSDB Editor to examine the contents of the RETRY Domain. Right-click **RETRY** Domain, and click **Status Display** from the shortcut menu.

Drag-and-Drop Notify for Wake-On-LAN Agents

CM can issue a **wake-up** packet that will remotely power-on agent computers that have been configured for Wake-On-LAN (**WOL**). The ease and straightforwardness of the drag-and-drop functionality simplifies the assigning of notify eligibility to WOL agents. Using the CM Admin CSDB Editor, it is easy to connect Notify instances to a USER, WORKGRP, DEPT, and SERVICE.

In order to use the drag-and-drop feature to assign agents for WOL Notify, use the CM Admin CSDB Editor. After you have accessed the CM Admin CSDB Editor, click and hold a Notify instance, drag it to the instance to which you want it connected, and release (drop) it, thereby creating the connection.

Refer to the *CM Application Manager Guide* for information on how to configure a WOL agent for drag-and-drop notify. Refer to the *CM Configuration Server Guide* for information on changes to the Configuration Server Settings file.

Filtering—Viewing Portions of the CM-CSDB

As your desired state grows, you will find it convenient to restrict the CM Admin CSDB Editor to view only a portion of the CM-CSDB. This saves effort in scrolling through the window to locate the domain, class, or instance you need to inspect or edit. The CM Admin CSDB Editor offers the ability to set filters to provide user-defined restricted views of the CM-CSDB. This is particularly valuable for classes that can have thousands of instances such as the USER Class or the FILE Class.

The filter specification tells the CM Admin CSDB Editor which domains, classes, or instances to display. Filtering is based upon matching the name of the domain, class, or instance with the filter specification you provide.

Filter specifications can contain one or more wildcards, represented by an asterisk, and/or one or more placeholder characters represented by a question mark. An asterisk matches any number of characters in the name of the component being filtered. Each question mark represents any one character in the name of the component being filtered.

Here are a few examples of valid filter expressions, and what they select.

Table 30 **Examples of Valid Filter Expressions**

Filter Specification	Selects
*	Wildcard to select all; removes any filter currently in effect.
_*	Names whose first character is an underscore.
_A	Names that have a prefix, and any level of the prefix is followed by an **A**.
*.?LL	Names ending in a period, followed by any one character, followed by **LL**.
-??WORD.HP?	Names containing any number of characters preceding a dash, followed by any two characters, followed by **WORD**, followed by any number of characters up to a period, followed by **HP**, followed by any one character.

You can apply filters that remain in effect for an entire CM Admin CSDB Editor session (*permanent filters*), or filters that apply only to the current CM Admin CSDB Editor window (*temporary filters*).

Permanent Filters

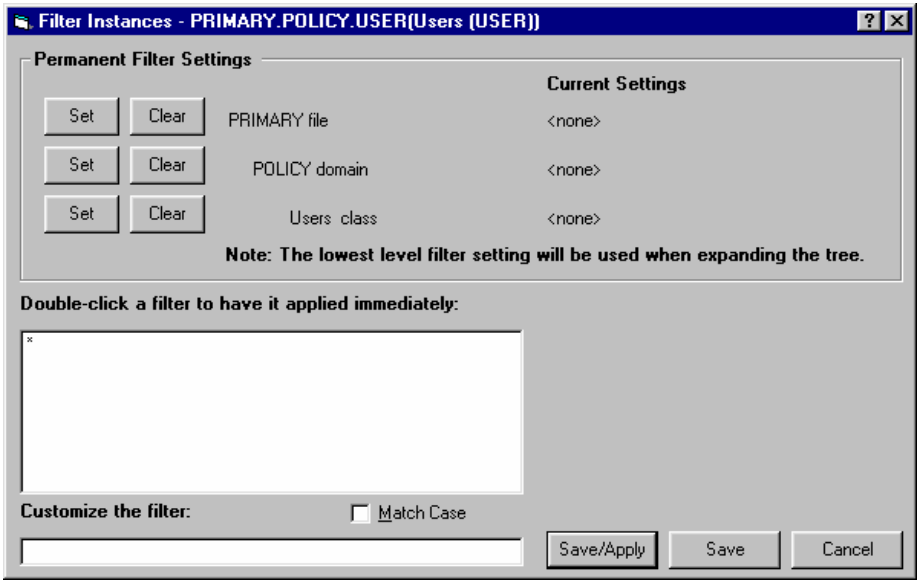
Permanent filters are applied every time you open any CM Admin CSDB Editor window during the current CM Admin CSDB Editor session. Each class can have its own permanent filter specification during a CM Admin CSDB Editor session.

To set or clear a permanent filter use the Permanent Filter Settings group box at the top of the Filter dialog box.

To set a permanent filter to filter instances of the USER Class on the prefix ESB_

1. Expand the POLICY Domain in the tree view.
2. Scroll down the tree view until you reach the USER Class.
3. Right-click the **USER Class**.

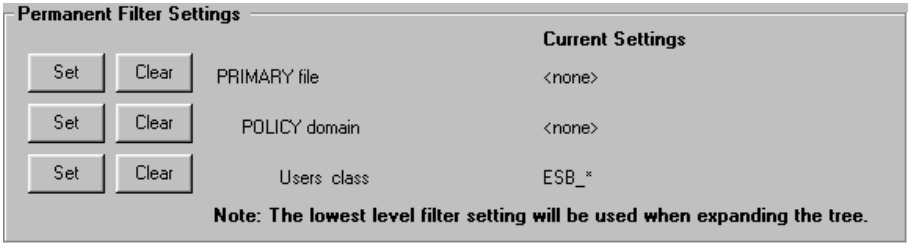
   A shortcut menu opens.
4. Click **Filter Instances** to open the Filter Instances dialog box.



5 Type **`ESB_*`** in the Customize the filter text box.

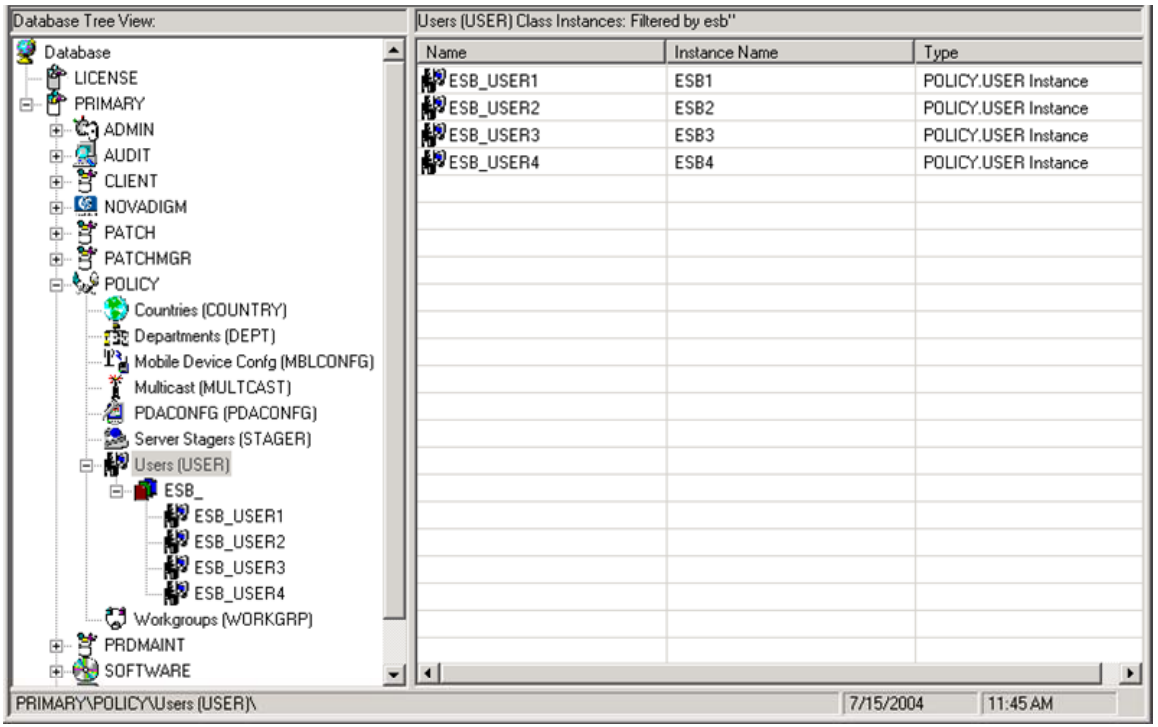
6 In the Permanent Filter Settings group box, click **Set** next to Users class.

The filter specification is displayed.



7 To apply the filter, click **Save/Apply** at the bottom of the dialog box.

The dialog box will close, and the tree display will be updated with the filter applied.



This filter will be applied to the USER Class in any additional CM Admin CSDB Editor windows you open during the current CM Admin CSDB Editor session.

To remove a permanent filter

1. Click the appropriate **Clear** button in the Filter Instances dialog box.
2. Click **Save/Apply** to remove the filter.

Temporary Filters

The Filter Instances dialog box also allows you to apply *temporary filters.* Temporary filters remain in effect only as long as the current CM Admin CSDB Editor window is open. To create a temporary filter, enter a filter specification in the Customize the filter field, and then click **Save/Apply**.

To set a temporary filter to filter instances of the USER Class on the prefix ESB_

1. Expand the POLICY Domain in the tree view.
2. Scroll down the tree view until you reach the USER Class.

3 Right-click the **USER Class**.

4 Select **Filter Instances** from the shortcut menu.

The Filter Instances dialog box opens.

5 Type **`ESB_*`** in the Customize the filter text box.

6 Click **Save/Apply**.

To remove a temporary filter

1 Apply ***** as a filter specification.

This matches any domain, class, or instance name, effectively removing the filter. The ***** filter specification is frequently found in the *express filter* list.

2 Click **Save/Apply** to remove the filter.

Viewing the Base or Null Instance

The Filter Instances dialog box also allows you to restrict your view to only the Base Instance or the Null Instance of the class.

To view the base or null instance

1 Expand the POLICY Domain in the tree view.

2 Scroll down the tree view until you reach the USER Class.

3 Right-click the **USER Class**.

4 Select **Filter Instances** from the shortcut menu.

The Filter Instances dialog box opens.

5 Click **Get Base Instance** to view the _BASE_INSTANCE_ of the USER Class, or click **Get Null Instance** to view the _NULL_INSTANCE_ of the USER Class.

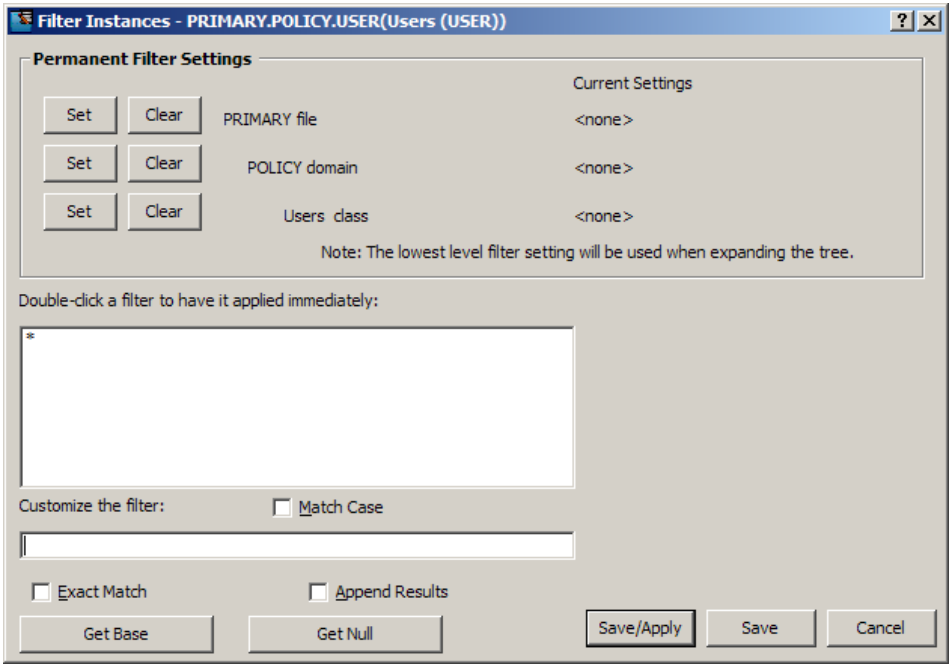
6 Click **Save/Apply**.

Appending Filter Results

You might want to continue viewing the current filter's result, and add an additional filter. To view both results at the same time, select the **Append**

Results check box (shown in Figure 35 below) when you perform the next filter.

Figure 35 Filter Instances dialog box

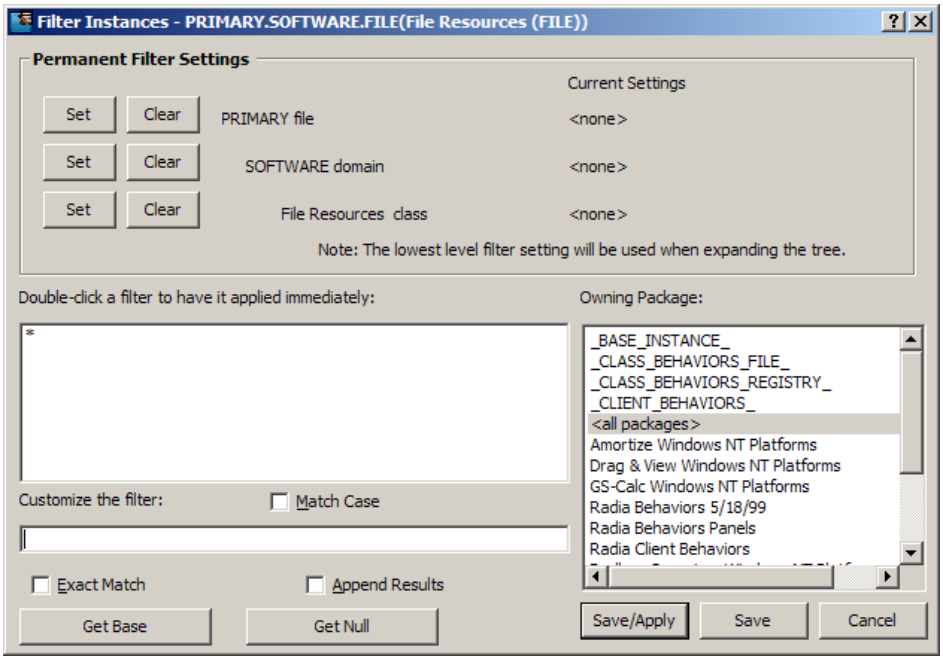


Filtering Component Class Instances

Component class instances (such as those of the FILE and DESKTOP Classes) can be filtered based on the configuration class instance to which they are connected.

When filtering component classes (such as the FILE class of the SOFTWARE Domain in this example), the Filter Instances dialog box presents a scrolling list labeled Owning Package, as shown in Figure 36 on page 162. The package instances are listed in this scrolling list.

Figure 36 Filtering component classes



To filter FILE instances to select only those connected to a particular PACKAGE instance

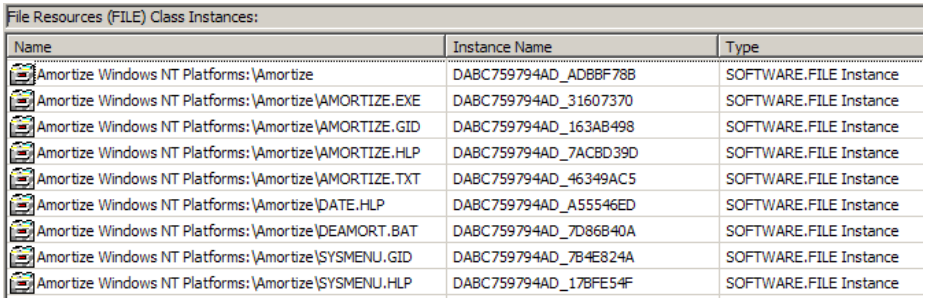
- Double-click a package listed in the Owning Package list.

  or

- Click the package in the Owning Package list, and click **Save/Apply**.

For example, select **Amortize Windows NT Platforms** in the Owning Package list, and then click **Save/Apply**. The Filter Instances dialog box will close, and the selected instances will appear in the list view of the CM Admin CSDB Editor window.

Figure 37 Owning Package FILE class instances



File Resources (FILE) Class Instances:

Name	Instance Name	Type
Amortize Windows NT Platforms:\Amortize	DABC759794AD_ADBBF78B	SOFTWARE.FILE Instance
Amortize Windows NT Platforms:\Amortize\AMORTIZE.EXE	DABC759794AD_31607370	SOFTWARE.FILE Instance
Amortize Windows NT Platforms:\Amortize\AMORTIZE.GID	DABC759794AD_163AB498	SOFTWARE.FILE Instance
Amortize Windows NT Platforms:\Amortize\AMORTIZE.HLP	DABC759794AD_7ACBD39D	SOFTWARE.FILE Instance
Amortize Windows NT Platforms:\Amortize\AMORTIZE.TXT	DABC759794AD_46349AC5	SOFTWARE.FILE Instance
Amortize Windows NT Platforms:\Amortize\DATE.HLP	DABC759794AD_A55546ED	SOFTWARE.FILE Instance
Amortize Windows NT Platforms:\Amortize\DEAMORT.BAT	DABC759794AD_7D86B40A	SOFTWARE.FILE Instance
Amortize Windows NT Platforms:\Amortize\SYSMENU.GID	DABC759794AD_7B4E824A	SOFTWARE.FILE Instance
Amortize Windows NT Platforms:\Amortize\SYSMENU.HLP	DABC759794AD_17BFE54F	SOFTWARE.FILE Instance

Express Filters

The box beneath the label Double-click a filter to have it applied immediately contains a list of potentially useful filter specifications that a CM administrator with appropriate authority can customize.

These are express filter specifications. As the label indicates, simply double-click the specification of your choice to apply it immediately. The filter is temporary, applying only to the currently open CM Admin CSDB Editor window.

Customizing Express Filters

You can specify the lists of express filters that appear for selection in the Filter Instances dialog box by storing them in the CM-CSDB, and making any needed connections. The CM Admin CSDB Editor retrieves the list of express filters for a particular Filter Instances dialog box from the CM-CSDB as the dialog box opens.

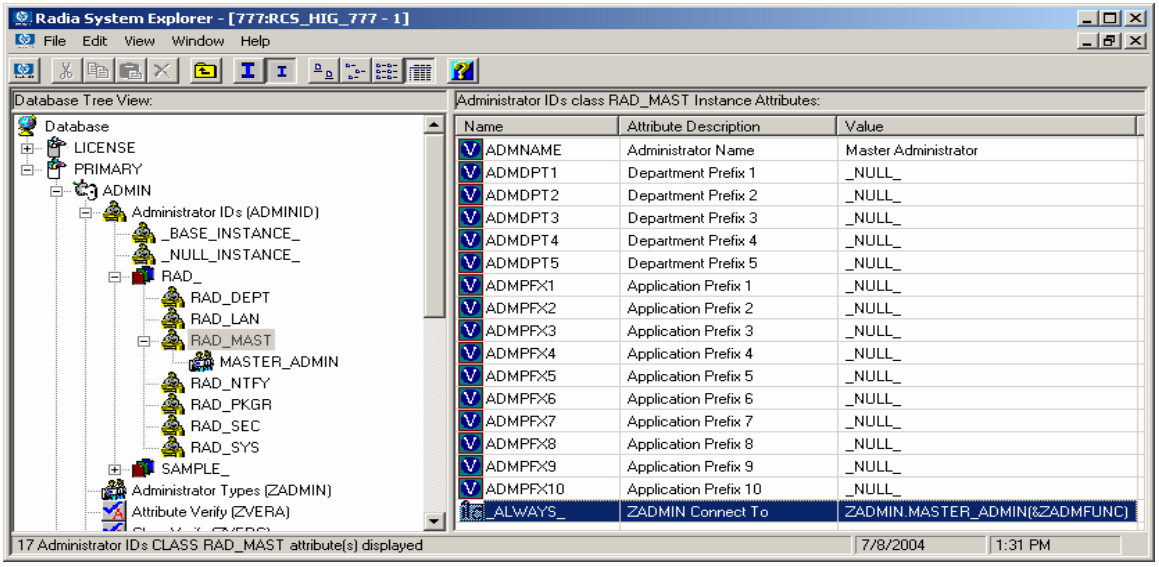
The CM Admin CSDB Editor locates the correct list in the CM-CSDB by executing a resolution process based on the user ID that was entered when the CM Admin CSDB Editor session was started, and the CM component (i.e., domain, class, or instance) being filtered. See Chapter 5, Understanding the Resolution Process for more information.

The list of express filters for a particular domain is taken from an instance of the ZLIST32 class of the ADMIN Domain in the PRIMARY File. Here's how.

Assume you have logged on as user ID RAD_MAST. The authority and defaults associated with an administrator's user ID are stored in the ADMIN Domain of the PRIMARY File. The ADMINID Class contains an instance for each administrator's user ID (in this case, the instance name is RAD_MAST).

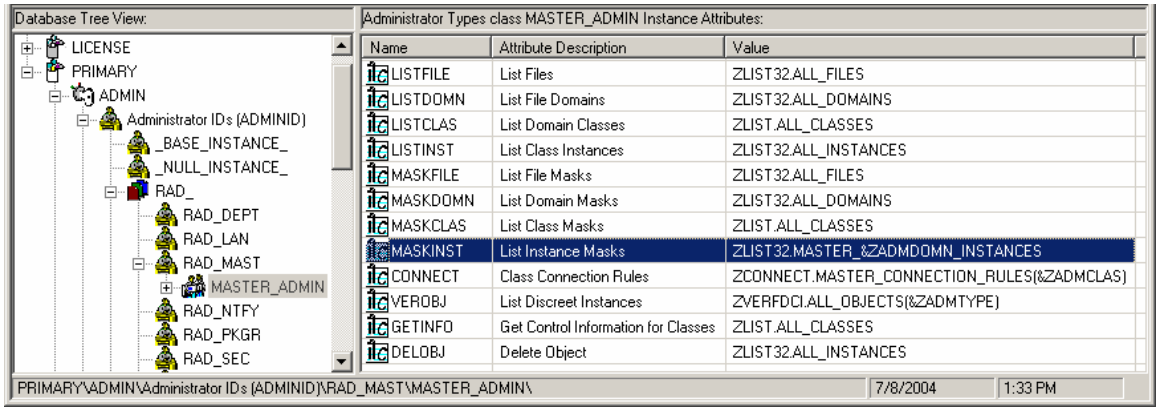
The RAD_MAST instance contains a connection to an instance of the ZADMIN class, in this case ZADMIN.MASTER_ADMIN.

Figure 38 The RAD_MAST instance connections



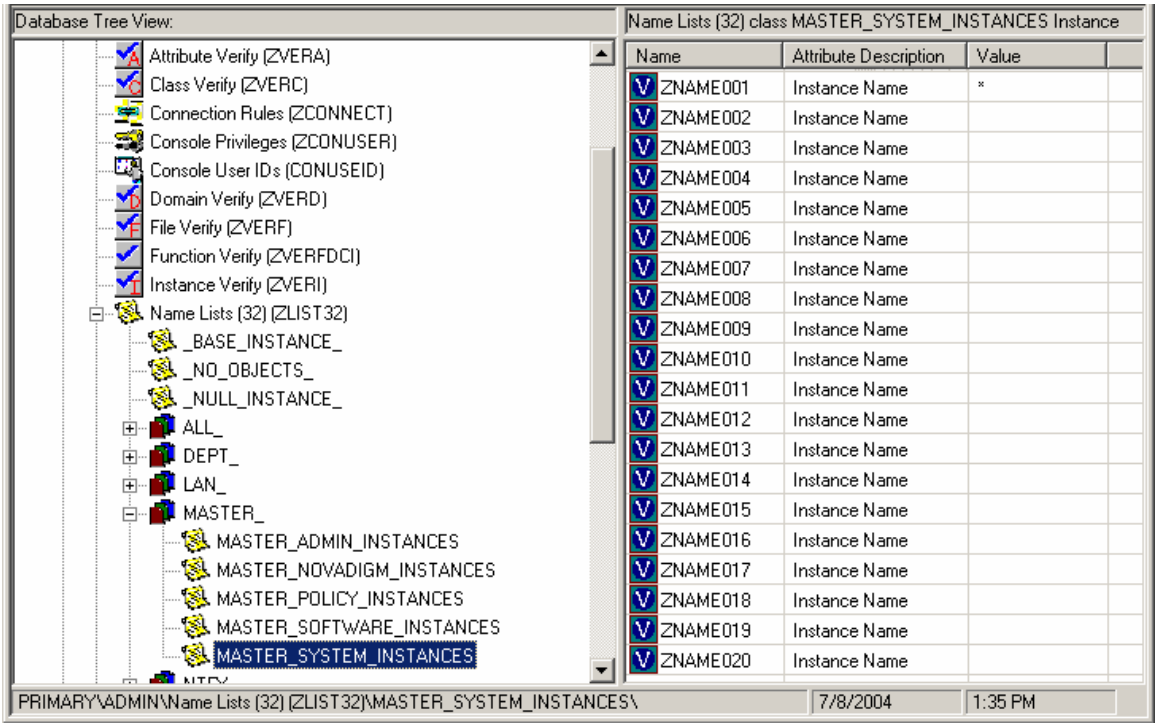
The connection to ZADMIN.MASTER_ADMIN passes a parameter in the system message (&ZADMFUNC) that indicates which of the connections contained in the ZADMIN.MASTER_ADMIN instance should be made. Only those connections whose names match the value of the system message (or whose name is _ALWAYS_) are actually made during a particular resolution. The CM Admin CSDB Editor sets the parameter's value before initiating the resolution. In this case, the CM Admin CSDB Editor sets the value of the ZADMFUNC variable (and thus, after symbolic substitution, the value of the system message) to MASKINST, because the Filter Instances dialog box needs to retrieve a list of instance masks.

Figure 39 ZADMFUNC variable set to MASKINST



This causes a connection to an instance of the ZLIST32 class, ZLIST32.MASTER_& ZADMDOMN_INSTANCES, based upon the domain to be filtered (SYSTEM). The domain is dynamically set in the ZADMDOMN variable by the CM Admin CSDB Editor. This is determined by which class the administrator right-clicked to access the Filter Instances dialog box. In our example, the connection will be made to ZLIST32.MASTER_SYSTEM_INSTANCES (after the resolution process performs symbolic substitution), which contains the desired list of default express filter specifications.

Figure 40 Connection made to LIST32.MASTER_SYSTEM_INSTANCES



To change the list of express filters for a domain, edit the appropriate instance of the ZLIST32 class. The changes will affect all administrators whose user IDs connect to the ZLIST32 instance you modify.

To set custom express filters for particular administrators, use the information detailed above to connect their user ID instances in the ADMIN.ADMINID Class to the appropriate ZLIST32 instance containing the desired list of express filters.

See Adding an Instance on page 70 and Editing an Instance on page 71 for information on how to use the CM Admin CSDB Editor to make these changes.

Summary

- The New Application wizard automates the creation of an Application (ZSERVICE) Class instance from a PACKAGE Class instance.
- Use the Version Group Editor to create, edit, and delete version instances associated with the Version Group. It controls the Version Group's deployment sequence.
- Service Optimization uses byte-level differencing and its ability to generate patches to recreate original data.
- CM manages products that need more than one service package to establish full product installation for operation. Create a CM Service Group when a product requires other service packages or has dependencies on other services.
- CM Notify enables the CM administrator to cause one or more agent computers to connect to the CM Configuration Server and to update or remove a specified application.
- CM can issue a wake-up packet that will remotely power on agent computers that have been configured for Wake-On-LAN.
- Use the CM Admin CSDB Editor to set filters, and append filter results, to provide user-defined restricted views of the CM-CSDB.

6 Understanding the Resolution Process

At the end of this chapter, you will:

- Understand the resolution process of the HP OpenView Configuration Management Configuration Server (CM Configuration Server).

The Resolution Process

The CM Configuration Server uses a procedure called the **resolution process** to accomplish a unit of work in response to a service request. The unit of work is defined by the contents of the HP OpenView Configuration Management Configuration Server Database (CM-CSDB) and parameters included in the request. In other words, what Configuration Management (CM) does depends on the information that is stored in its database, and the information that accompanies the request for CM to perform an action.

For example, the CM agent connect submits service requests to the CM Configuration Server, and the CM Configuration Server performs a resolution process in response to each such request.

Heaps and Objects

One way that the CM Configuration Server recognizes a service request is when it receives an **object**. An object is simply a storage structure, or a container, for a record or a record set. An object consists of **instances**, also known as **heaps**. An instance is an occurrence of a set of variables. Objects (such as ZMASTER) can consist of a single instance or several instances, as does the FILE object.

You can inspect and even modify desktop objects using the CM Admin Agent Explorer.

CM Admin Agent Explorer

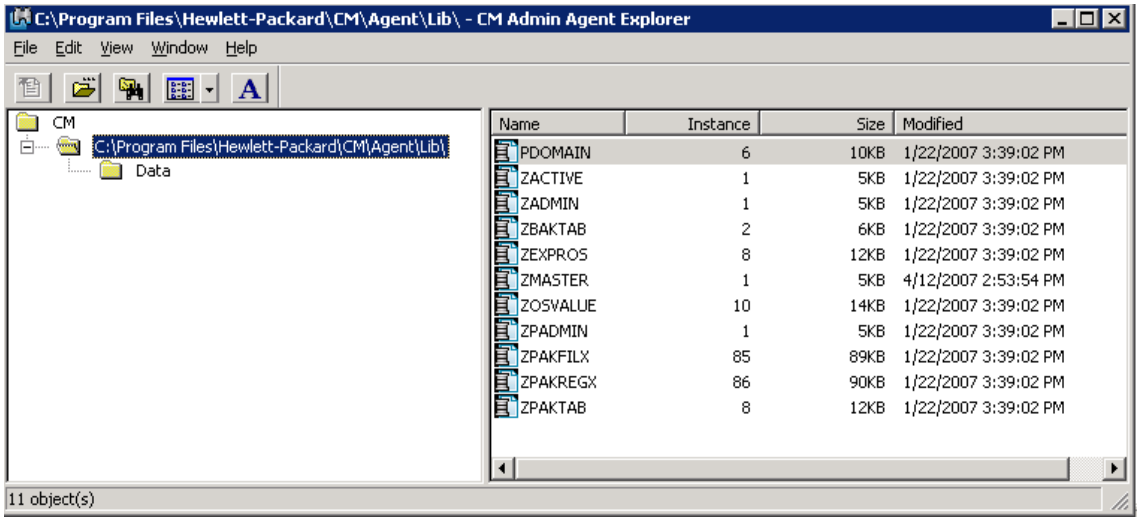
The CM Admin Agent Explorer is used to view or edit objects. A user with the CM Administrator installed can use the CM Admin Agent Explorer as a diagnostic utility to view local objects on the desktop, edit objects, or create new objects.

To access the CM Admin Agent Explorer

- Click **Start → Programs → HP OVCM Administrator → CM Admin Agent Explorer**.

  The CM Admin Agent Explorer opens.

Figure 41 List view in CM Admin Agent Explorer

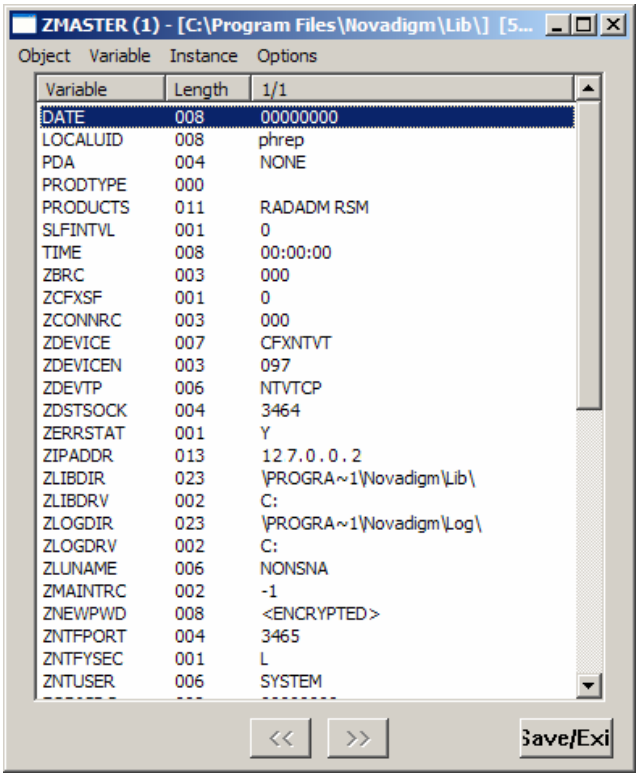


The figure above displays the CM Admin Agent Explorer at its highest level. The list view (the right side of the window) in the CM Admin Agent Explorer displays the:

- Instance name,
- Number of Instances,
- Size of the instance, and
- Date and time of the last modification.

To view an instance, double-click it. The instance information will be displayed as shown in the following figure.

Figure 42 Double-click on an instance to display objects inside.



The resolution process uses the objects that are stored for each agent to bring them to the **desired state**. The desired state embodies the content that CM manages for each end-user computer, and is defined for each end-user computer in the CM-CSDB. The **desired state** is created and managed using the CM Admin CSDB Editor.

Resolution Process—Where Does It Begin?

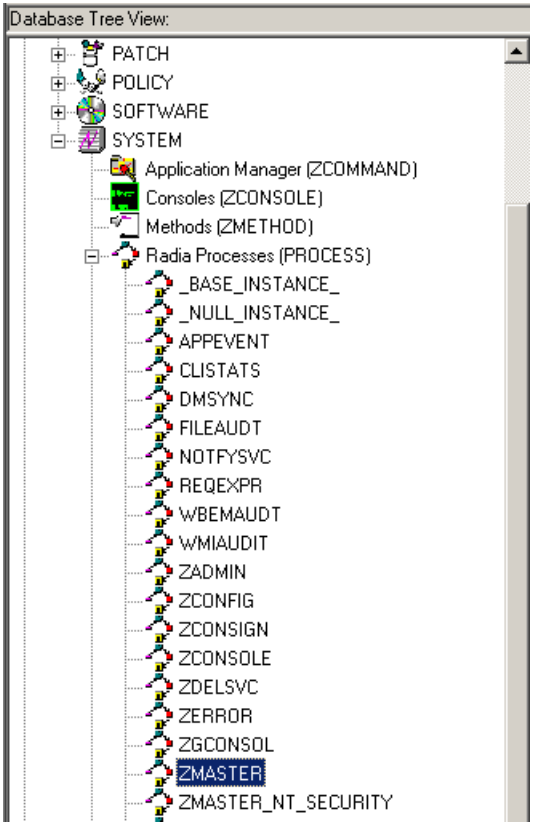
As previously stated, the resolution process begins when an agent computer connects to the CM Configuration Server. The ZMASTER is the first object sent to the CM Configuration Server during the agent connect. It contains information about the agent computer that is needed to run CM, such as the identity of the subscriber and the IP address of the computer.

The CM Configuration Server stores the ZMASTER object in a transient storage area called **global memory**. The resolution process defines variable attributes into objects contained in global memory, and their current values

are maintained there. The current value of a variable attribute is the value it holds as of its most recent reference during the resolution process.

After storing ZMASTER in global memory, the CM Configuration Server finds the PROCESS Instance for the ZMASTER. This is known as the **process entry point**. Its location is SYSTEM.PROCESS.ZMASTER.

Figure 43 SYSTEM.PROCESS.ZMASTER

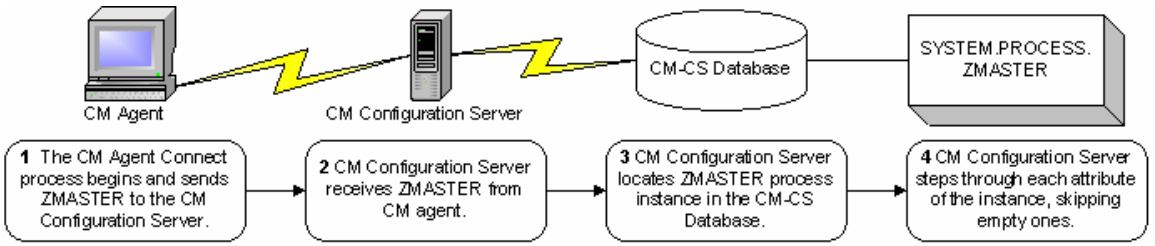


The CM Configuration Server reads each attribute of SYSTEM.PROCESS.ZMASTER. As a result of reading an attribute, the CM Configuration Server might:

- Set variable values.
- Evaluate an expression.
- Execute a method.
- Connect to other instances.

If there is a connection to another instance, the CM Configuration Server processes the connected instance. Then the resolution process resumes with the referring instance at the next attribute after the connection attribute.

Figure 44 CM Configuration Server performing resolution



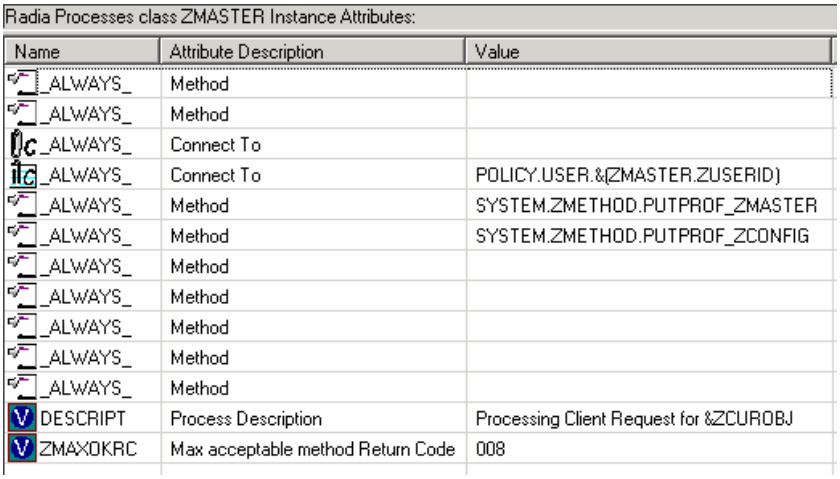
Resolution Example

Let's look at how an agent's ZMASTER object is resolved.

In Figure 45 on page 175 the first connection instance links to POLICY.USER.&(ZMASTER.ZUSERID).

After processing this, resolution will return to PRIMARY.SYSTEM.PROCESS.ZMASTER and will process the next attribute which is a connection instance to SYSTEM.ZMETHOD.PUTPROF_ZMASTER.

Figure 45 PRIMARY.SYSTEM.PROCESS.ZMASTER instance



Radia Processes class ZMASTER Instance Attributes:

Name	Attribute Description	Value
ALWAYS	Method	
ALWAYS	Method	
ALWAYS	Connect To	
ALWAYS	Connect To	POLICY.USER.&(ZMASTER.ZUSERID)
ALWAYS	Method	SYSTEM.ZMETHOD.PUTPROF_ZMASTER
ALWAYS	Method	SYSTEM.ZMETHOD.PUTPROF_ZCONFIG
ALWAYS	Method	
ALWAYS	Method	
ALWAYS	Method	
ALWAYS	Method	
ALWAYS	Method	
DESCRIPT	Process Description	Processing Client Request for &ZCUROBJ
ZMAXOKRC	Max acceptable method Return Code	008

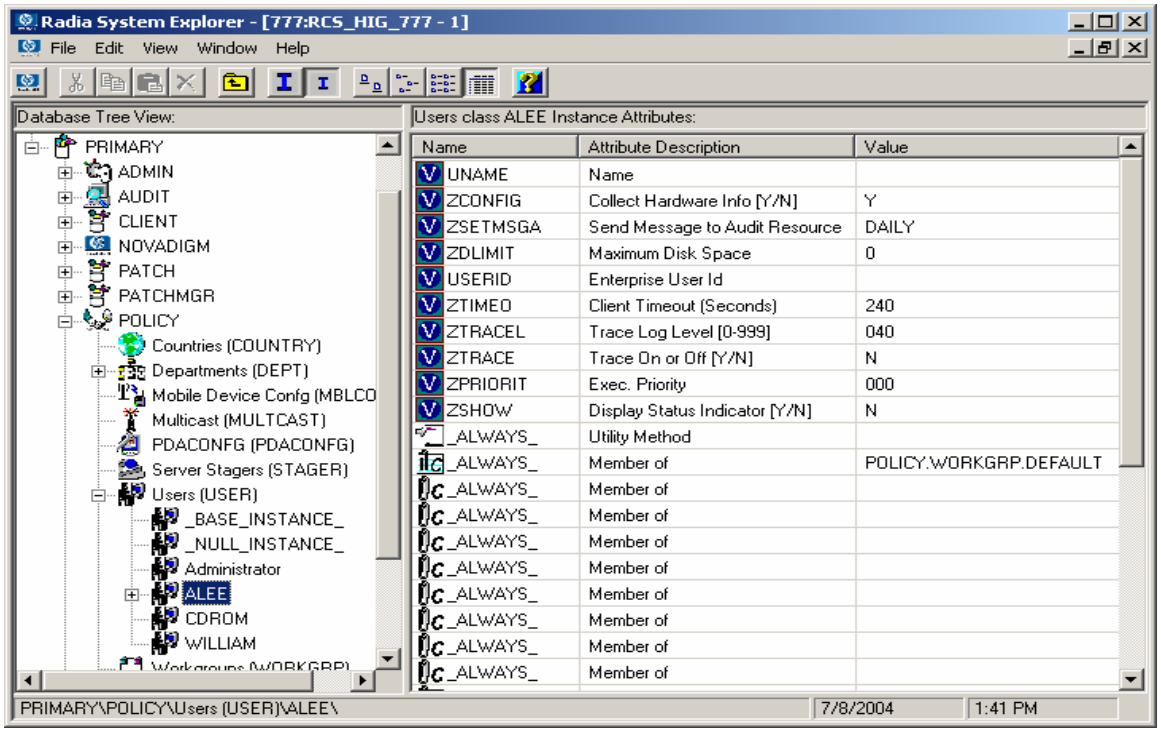
Resolution Process and Symbolic Substitution

During resolution, the CM Configuration Server performs **symbolic substitution** to set values and connect to other instances. When processing symbolic substitutions, CM will substitute the value of the second variable to replace the reference in the first variable.

References to be processed with symbolic substitution are specified using an initial ampersand.

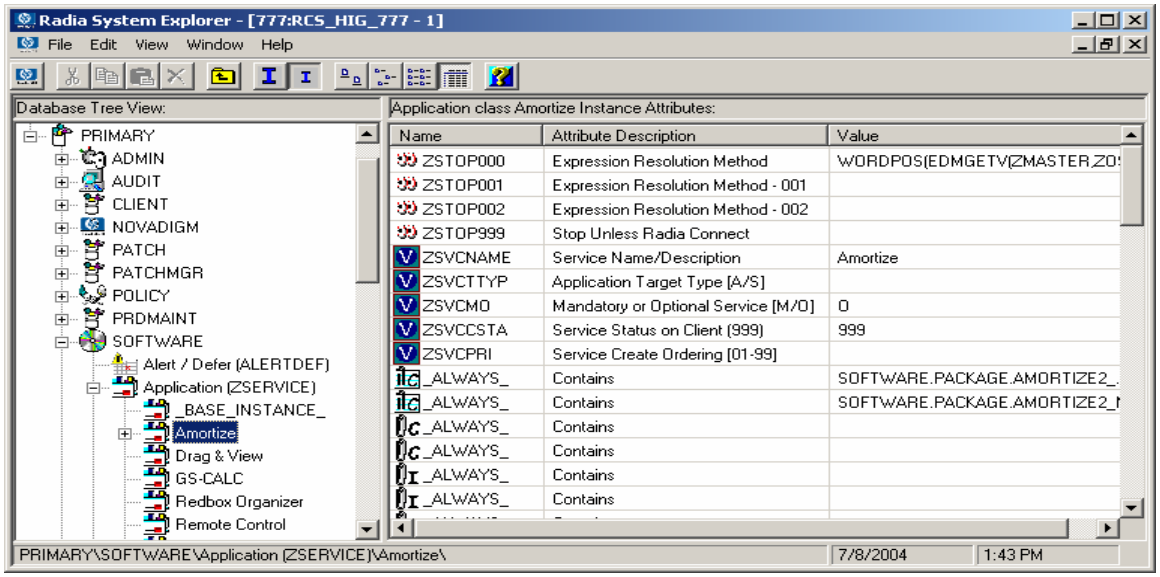
For example, in SYSTEM.PROCESS.ZMASTER there is an always connection to POLICY.USER.&(ZMASTER.ZUSERID). The CM Configuration Server substitutes the value of ZUSERID from the ZMASTER object that is in global memory. In Figure 46 on page 176 the value of ZUSERID is ALEE. The resolution process will connect to POLICY.USER.ALEE to begin resolving the ALEE instance.

Figure 46 POLICY.USER.ALEE instance



In Figure 47 on page 177 after setting a number of variables, the first connection attribute is to SOFTWARE.ZSERVICE.AMORTIZE. In your implementation, the POLICY instance might connect to a workgroup or a department that would then connect to a service.

Figure 47 ZSERVICE.AMORTIZE instance

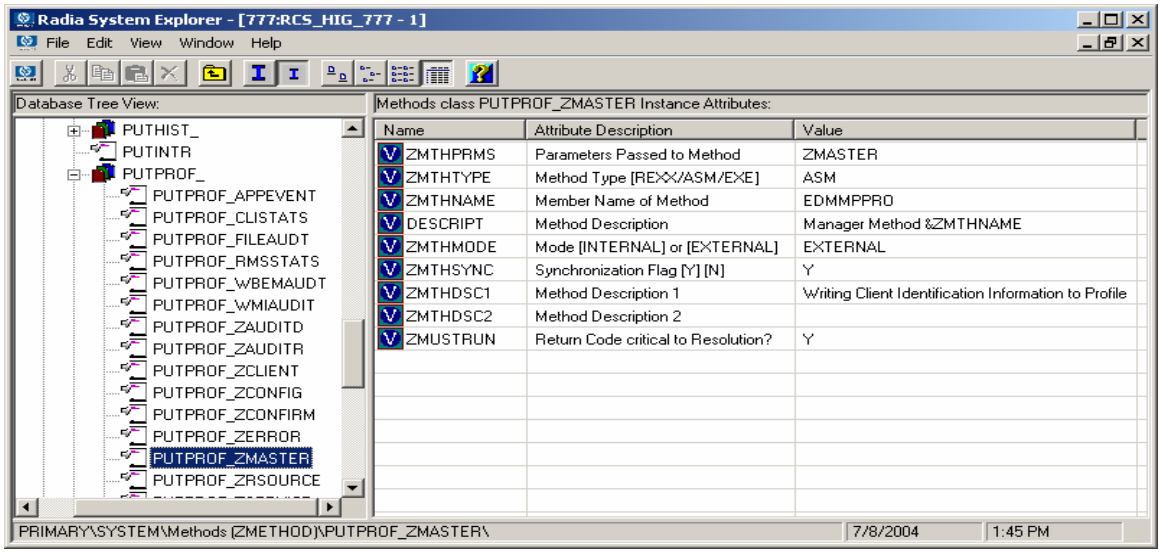


The resolution proceeds to the service instance that is indicated in the first connection of the USER as shown in Figure 47 above. The Amortize service resolution begins with a ZSTOP expression variable. This expression variable contains statements that, if evaluate to *true*, stop the resolution of the current instance. An expression allows alternative paths to be taken in a given resolution, based on variable data. In this case, the expression checks to be sure that the operating system of the agent computer is allowed for the Amortize software. If the agent computer's operating system is Windows 2000, Windows 2003, Windows XP, or Vista, or, the resolution process continues with this instance, connecting the file instances, registry entries, path instances, and shortcuts. If the agent has a different operating system, resolution returns to SOFTWARE.ZSERVICE.AMORTIZE, and to the next connection instance.

Eventually, the resolution process will return to the USER instance, finish resolving it and return to the process entry point, SYSTEM.ZPROCESS.ZMASTER. Referring back to Figure 45 on page 175 the next attribute connects to the PUTPROF_ZMASTER method.

A **method** is a program that controls, customizes, and reports on the deployment of CM-managed applications.

Figure 48 ZMETHOD.PUTPROF_ZMASTER instance



The CM Configuration Server executes the EDMMPPRO method, passing ZMASTER as a parameter. This causes the contents of the ZMASTER object in global memory to be written to the PROFILE File of the CM-CSDB.

After completing the processing of all attributes in the SYSTEM.PROCESS.ZMASTER instance, resolution terminates.

Summary

- The CM Configuration Server uses a procedure called the resolution process to accomplish a unit of work in response to a service request.
- One way that the CM Configuration Server recognizes a service request is when it receives an object.
- The CM Admin Agent Explorer is one of two tools that can be used to view or edit objects.
- The resolution process uses the objects that are stored for each agent to bring them to the desired state.
- The ZMASTER is the first object sent to the CM Configuration Server during the agent connect.
- During resolution, the CM Configuration Server performs symbolic substitution to set values and connect to other instances.

7 Maintaining Connection Rules in the CM-CSDB

At the end of this chapter, you will:

- Understand connection rules.
- Know how to enable connection rules.

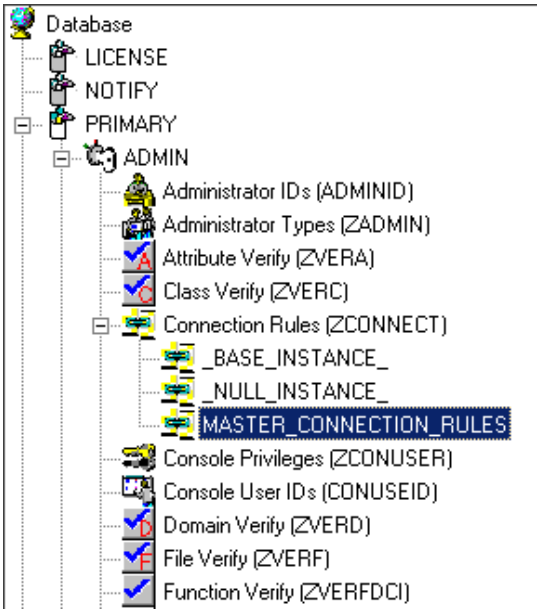
Introduction

The CM Admin CSDB Editor provides easy and reliable drag-and-drop connections between instances of classes. These connections, which contain connection rules, are stored in the CM-CSDB.

Connection Rules Overview

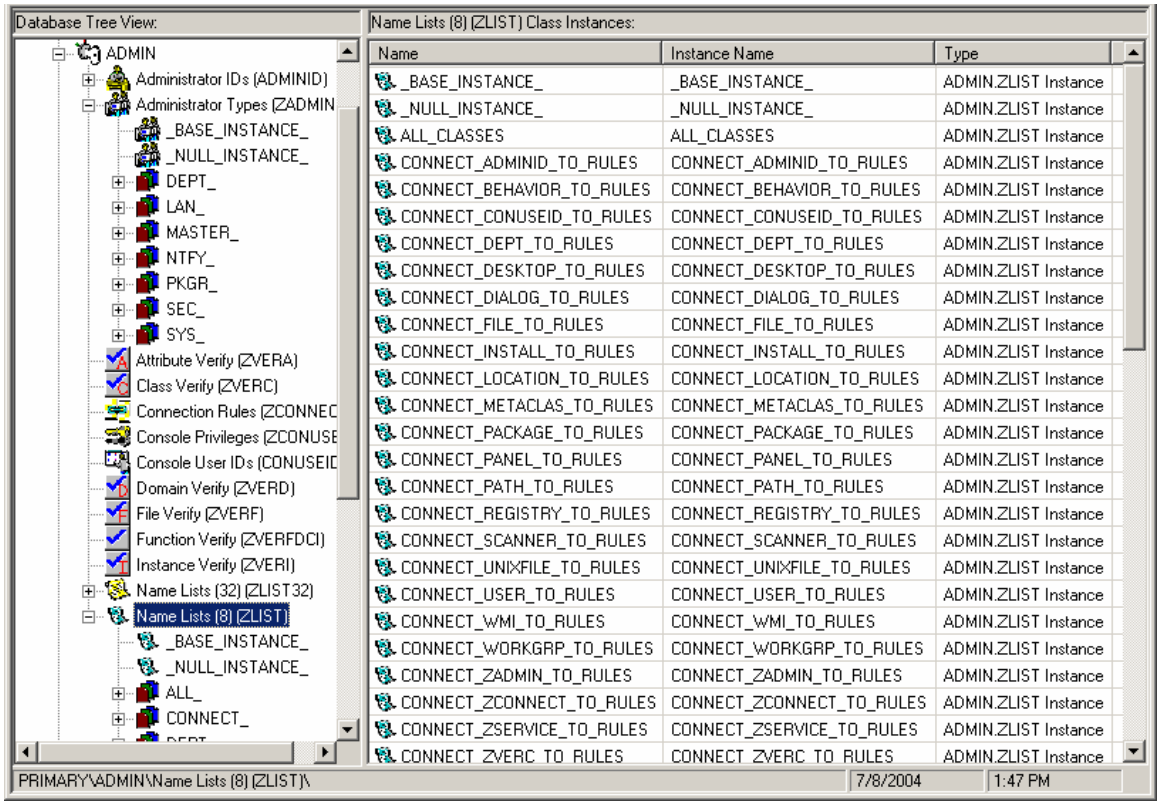
In order to allow for drag-and-drop capabilities for an added class, it is necessary to establish rules for the new class to follow. These connection rules are stored in the PRIMARY File, ADMIN Domain, ZCONNECT Class, MASTER_CONNECTION_RULES Instance.

Figure 49 MASTER_CONNECTION_RULES instance



While the ZCONNECT Class houses the connection rules, the Name List (ZLIST) Class contains all of the rules for all of the classes. In order to create a **connection rule**, you will have to first create the rule in the ZLIST Class. Then you will add the rule to the ZCONNECT Class.

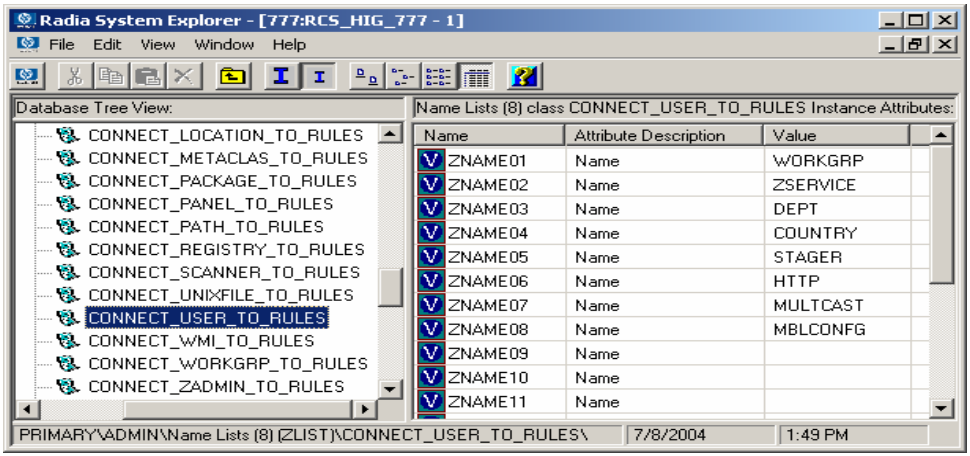
Figure 50 Name Lists (ZLIST) Class in ADMIN Domain



The CM Admin CSDB Editor then uses these rules to populate the class list in the Show Connections dialog box, and to control drag-and-drop connections.

For example, let's look at the connection rules that have been established for the USER Class. To do this, examine the ZLIST.CONNECT_USER_TO _RULES Instance as shown in Figure 51 on page 184.

Figure 51 ZLIST.CONNECT_USER_TO_RULES Instance



An instance of this particular USER Class can contain a connection to any of the following classes:

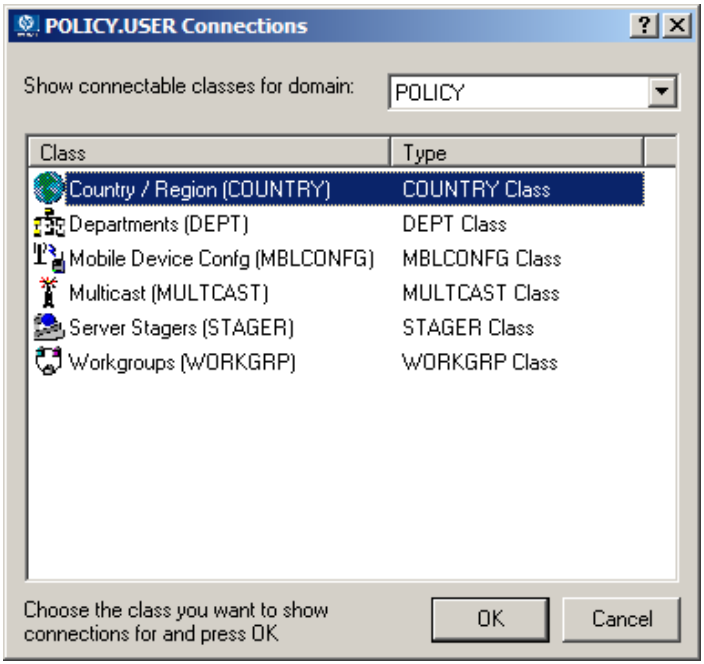
WORKGRP	ZSERVICE	DEPT	COUNTRY
STAGER	HTTP	MULTCAST	MBLCONFG

The ZLIST instance contains a list of up to 20 classes to which the class, represented by the ZLIST instance (in this case, USER), can connect. You can edit this list to include additional classes, or remove classes from a connection rule. If any class in the CM-CSDB needs to be able to connect to more than 20 different classes, you can edit the ZLIST Class template and add additional ZNAME*nn* variables as needed.

After the rules for connection have been established in the Name List (ZLIST) Class, the connection type can be created. Since we are creating a connection rule for drag-and-drop, the class variable is added to the Connection Rules (ZCONNECT) Class. You will then define the type of connection that can be made to the USER Class by editing the Rules variable.

When completed, drag-and-drop connection can be made to the new class. Figure 52 on page 185 shows the connectable classes for the POLICY Domain where the USER Class is located.

Figure 52 Connectable classes for the POLICY Domain



Enabling Connections in New Classes

If you add to the CM-CSDB a new class that can validly contain connections to other classes, you will need to:

- Create an instance in the ZLIST Class to represent the new class.
- Set the new ZLIST Class instance to allow the connection rules.
- Add the new class to the ZCONNECT Class template.
- Set the new class' variable in the MASTER_CONNECTION_RULES.

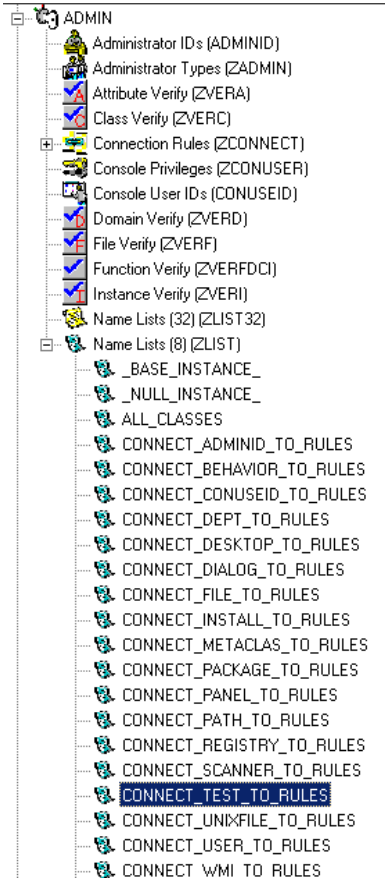
For example, the steps below show how to add the connections for a new class called TEST.

- First, create an instance of the ZLIST Class named CONNECT *_newclass*_TO_RULES where *newclass* is the name of the new class, for example, TEST.

1 Open the CM Admin CSDB Editor and navigate to the PRIMARY.ADMIN.Name Lists (8) (ZLIST) Class.

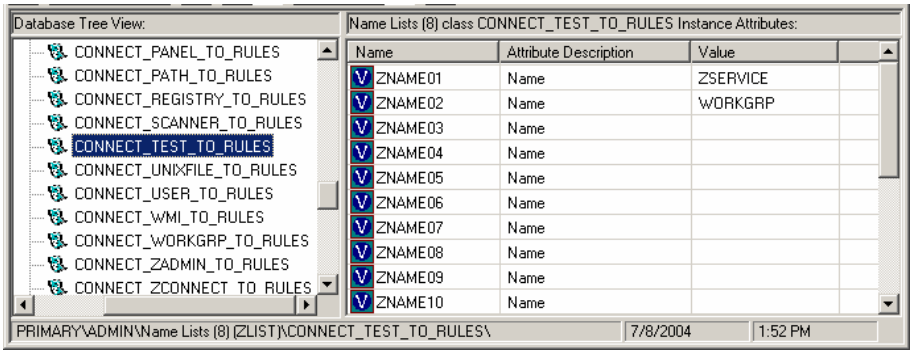
2 Right-click **ZLIST** and select **New Instance**.

3 Name the new instance as suggested above, CONNECT_TEST_TO RULES. This figure shows the added CONNECT_TEST_TO_RULES in the ZLIST Class.



4 Edit the CONNECT_TEST_TO_RULES Instance, setting each ZNAME*nn* variable to the name of a class to which the new class can be connected.

For example, set values for ZNAME01 and ZNAME02 to allow connections to the ZSERVICE and WORKGRP Classes, as shown in the following figure.



Now you will need to add the **connection rule** variable to the Connection Rules (ZCONNECT) Class.

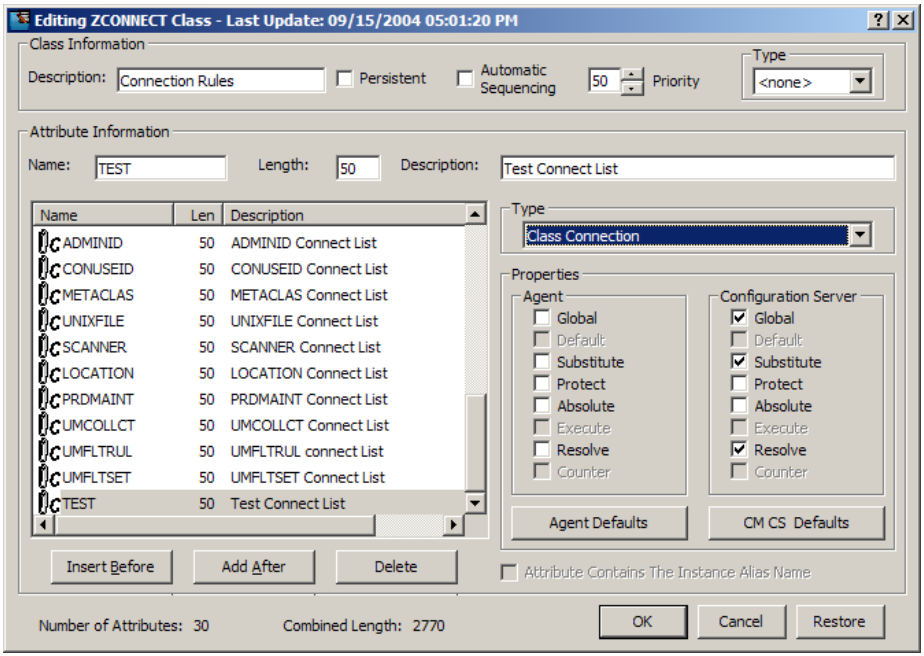
5 Navigate to the PRIMARY.ADMIN.ZCONNECT Class.

6 Right-click on the **ZCONNECT** Class and select **Edit Class** from the shortcut menu.

7 Add the new class variable (TEST) to the class template for the ZCONNECT Class. The length of the attribute will be 50, and the description is Test Connect List.

When adding an attribute to the ZCONNECT Class, be sure to change the attribute type to Class Connection. See the figure below.

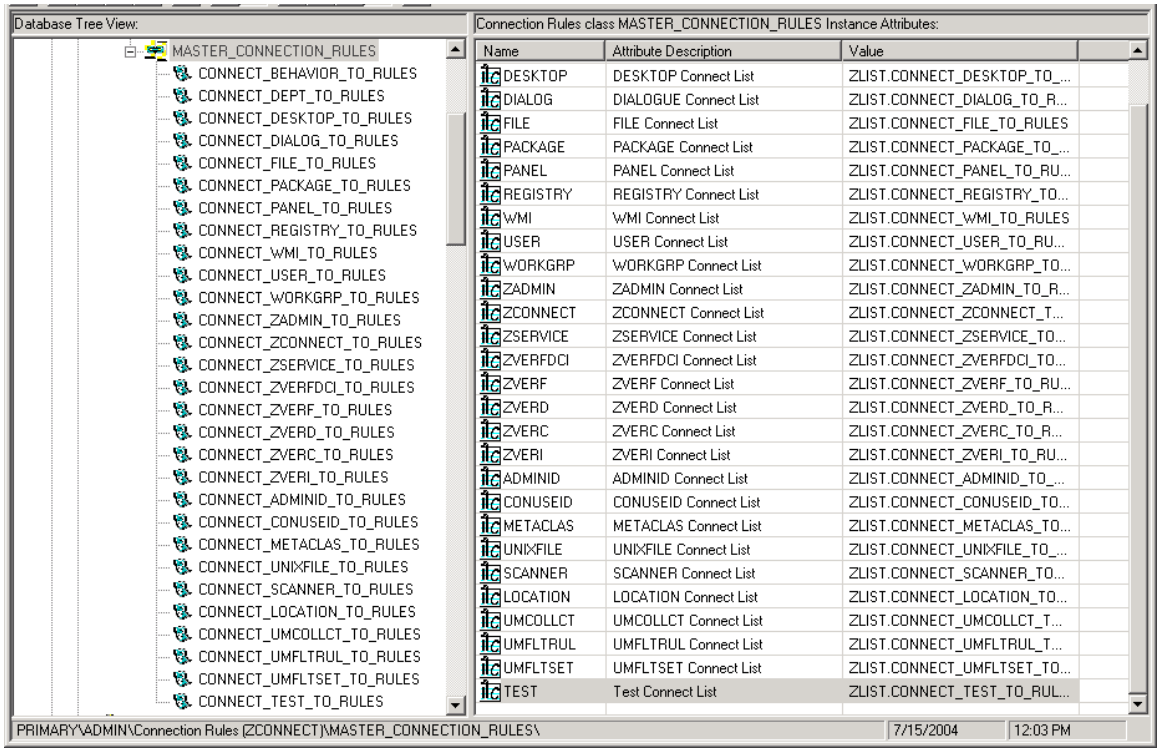
For information on editing a class, see Editing a Class beginning on page 57.



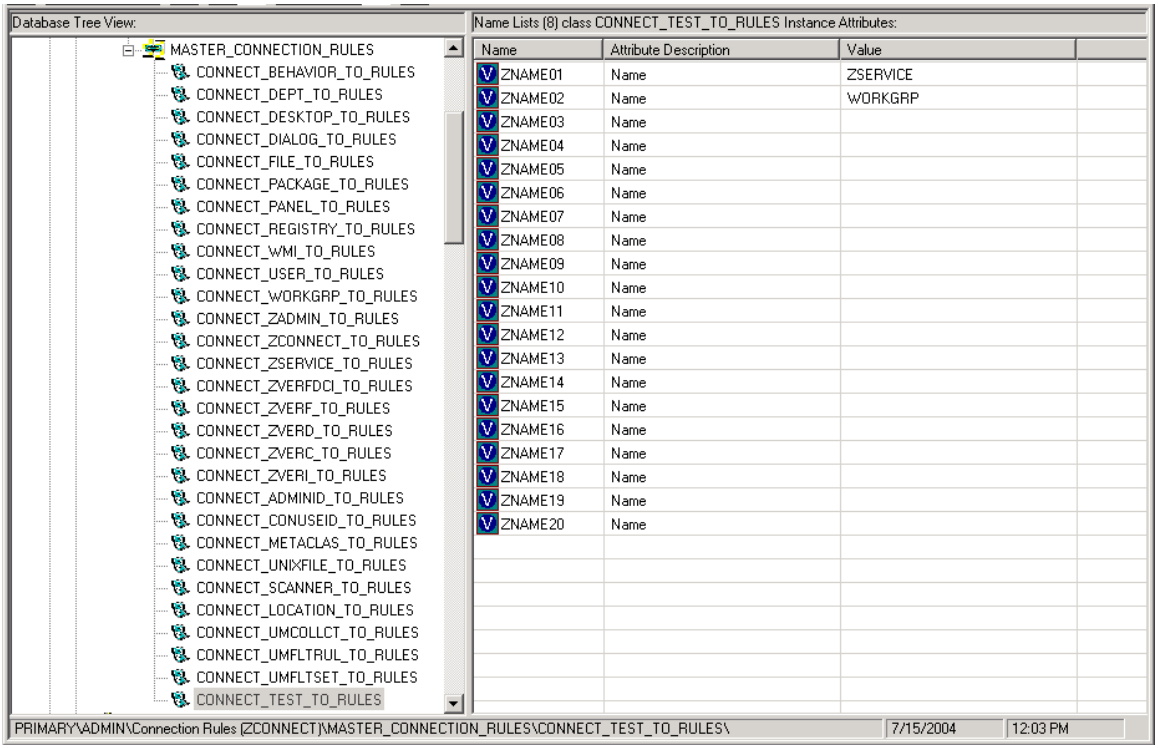
After you have added the new attribute to the class template, you will need to set the value of the new attribute in the ZCONNECT.MASTER _CONNECTION_RULES Instance.

8 Locate the new instance in the list view (right side of the screen), and set the value of the newly added variable to ZLIST.CONNECT_TEST _TO_RULES, where *newclass* is the name of the newly added class.

For example, check the value that has been set for the TEST attribute in the next figure.



The next figure shows that the new connection rule has been added to the MASTER_CONNECTION_RULES, thereby allowing drag-and-drop connections for the new class.


Database Tree View:
MASTER_CONNECTION_RULES
CONNECT_BEHAVIOR_TO_RULES
CONNECT_DEPT_TO_RULES
CONNECT_DESKTOP_TO_RULES
CONNECT_DIALOG_TO_RULES
CONNECT_FILE_TO_RULES
CONNECT_PACKAGE_TO_RULES
CONNECT_PANEL_TO_RULES
CONNECT_REGISTRY_TO_RULES
CONNECT_WMI_TO_RULES
CONNECT_USER_TO_RULES
CONNECT_WORKGRP_TO_RULES
CONNECT_ZADMIN_TO_RULES
CONNECT_ZCONNECT_TO_RULES
CONNECT_ZSERVICE_TO_RULES
CONNECT_ZVERFDCL_TO_RULES
CONNECT_ZVERF_TO_RULES
CONNECT_ZVERD_TO_RULES
CONNECT_ZVERC_TO_RULES
CONNECT_ZVERI_TO_RULES
CONNECT_ADMINID_TO_RULES
CONNECT_CONUSEID_TO_RULES
CONNECT_METACLAS_TO_RULES
CONNECT_UNIXFILE_TO_RULES
CONNECT_SCANNER_TO_RULES
CONNECT_LOCATION_TO_RULES
CONNECT_UMCOLLCT_TO_RULES
CONNECT_UMFLTRUL_TO_RULES
CONNECT_UMFLTSET_TO_RULES
CONNECT_TEST_TO_RULES
Name Lists (8) class CONNECT_TEST_TO_RULES Instance Attributes:
Name
Attribute Description
Value
ZNAME01 Name ZSERVICE
ZNAME02 Name WORKGRP
ZNAME03 Name
ZNAME04 Name
ZNAME05 Name
ZNAME06 Name
ZNAME07 Name
ZNAME08 Name
ZNAME09 Name
ZNAME10 Name
ZNAME11 Name
ZNAME12 Name
ZNAME13 Name
ZNAME14 Name
ZNAME15 Name
ZNAME16 Name
ZNAME17 Name
ZNAME18 Name
ZNAME19 Name
ZNAME20 Name
PRIMARY\ADMIN\Connection Rules (ZCONNECT)\MASTER_CONNECTION_RULES\CONNECT_TEST_TO_RULES\
7/15/2004
12:03 PM

Summary

- In order to enable drag-and-drop capabilities for an added class, it is necessary to establish rules for the new class.
- These connection rules are stored (in the ADMIN Domain of the PRIMARY File) in the MASTER_CONNECTION_RULES Instance of the Connection Rules (ZCONNECT) Class.
- The Name List (ZLIST) Class contains the rules for all of the classes.
- The ZCONNECT Class houses the rules that enable connections between classes.
- CM Admin CSDB Editor uses these rules to populate the class list in the Show Connections dialog box, and to control drag-and-drop connections.

8 Feature Set Editor

At the end of this chapter, you will:

- Know what the Configuration Management (CM) Feature Set Editor is and what it does.
- Know how to access the editor.
- Understand the features of the editor.
- Know how to manage feature sets in native MSI mode.

Feature Set Editor Overview

The CM Admin CSDB Editor includes a Feature Set Editor, with a wizard type interface, to guide you through the selection and configuration of the features of a Windows Installer-enabled software package managed by CM. This allows you to use the features of Windows Installer while taking advantage of the packaging and deployment technologies available in CM.

With the release of Office 2000, Microsoft introduced new technology (Microsoft Windows Installer) for installing and maintaining software products on the Windows platform. CM products can install and manage Windows Installer-enabled applications as follows:

The CM Admin Publisher allows for *basic* and *advanced* methods of publishing and configuring Windows Installer-enabled applications.

For information regarding the packaging of Windows Installer-enabled applications, see the *HP OpenView Configuration Management Extensions for Windows Installer Getting Started Guide (CM Extensions for WI GSG*).

For information regarding the publishing of Windows Installer-enabled applications, see the *HP OpenView Configuration Management Administrator Publisher Guide* (*CM Admin Publisher Guide*). Both guides are available from the HP OpenView Support web site.

The CM-CSDB enables HP products to store Windows Installer feature sets.

CM provides policy administration and state data gathering.

CM provides the ability to control the customization of MSI packages without the need to create Windows Installer Transform (MST) files.

CM provides the ability to discover and manage previously installed Windows Installer-enabled applications that were not deployed using CM.

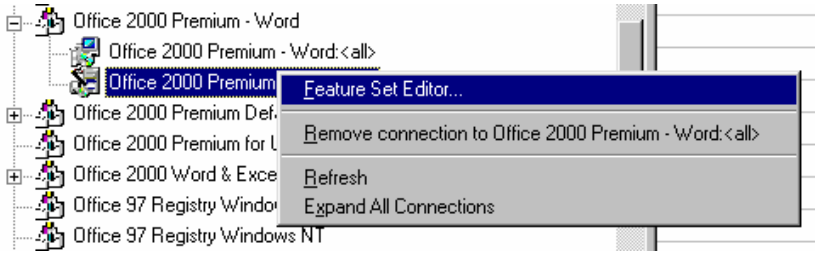
Accessing the Feature Set Editor

To access the Feature Set Editor

1 Locate the Application Packages instance that represents the software package in the CM-CSDB whose features you wish to edit.

2 Double-click this instance to display its MSI and MSIFEATS connections.

3 Right-click the MSIFEATS connection in the tree view, as follows:



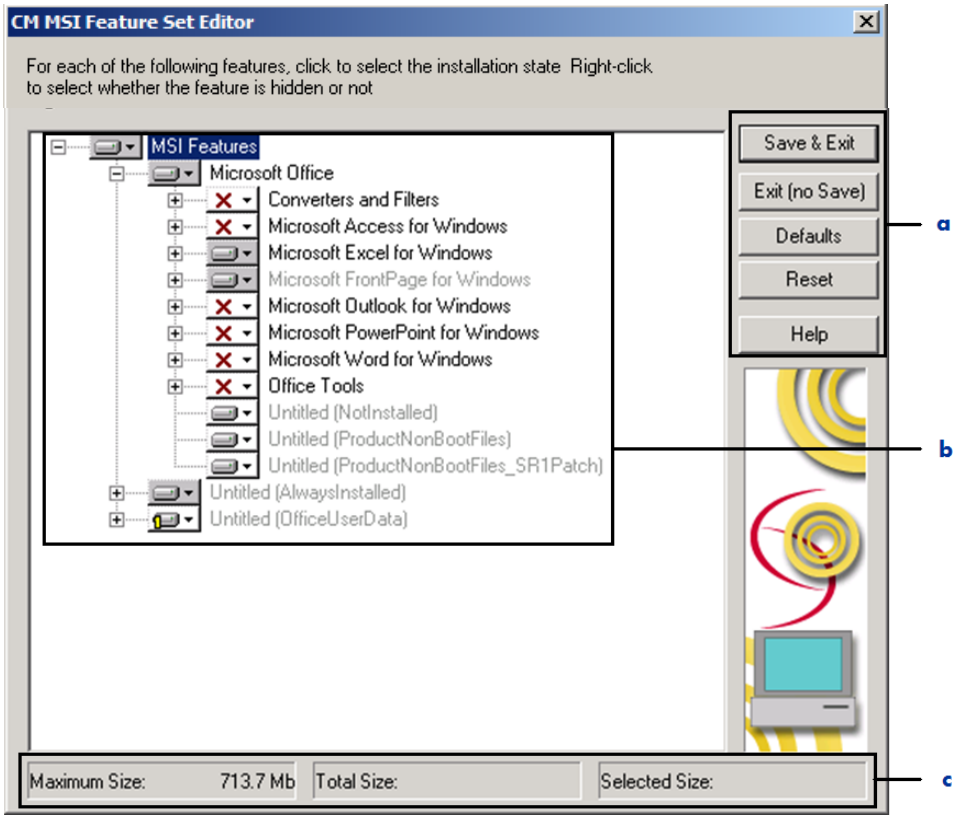
4 Select **Feature Set Editor** from the shortcut menu.

The Feature Set Editor dialog box opens.

About the CM MSI Feature Set Editor

Figure 53 on page 196 shows that the Feature Set Editor interface is divided into three areas, the **tree view** on the left, a **button list** on the right, and **package size information** on the bottom.

Figure 53 Feature Set Editor dialog box



Legend

a Buttons

b Tree view

c Package size information

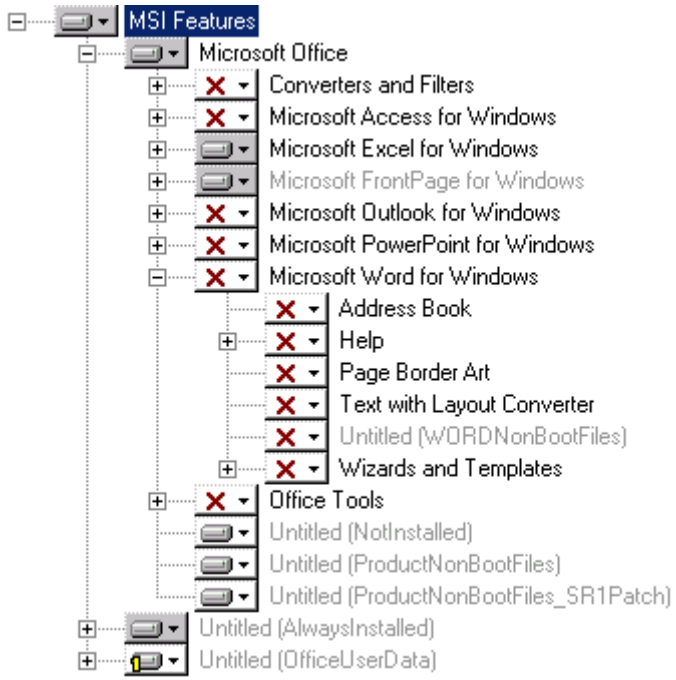
Tree View

Features are grouped into a hierarchy that you navigate using the tree control. Higher-level features consist of two or more lower-level, child features.

To view the child features comprising any particular feature, click on the small plus sign in the box next to the feature. The plus sign changes to a

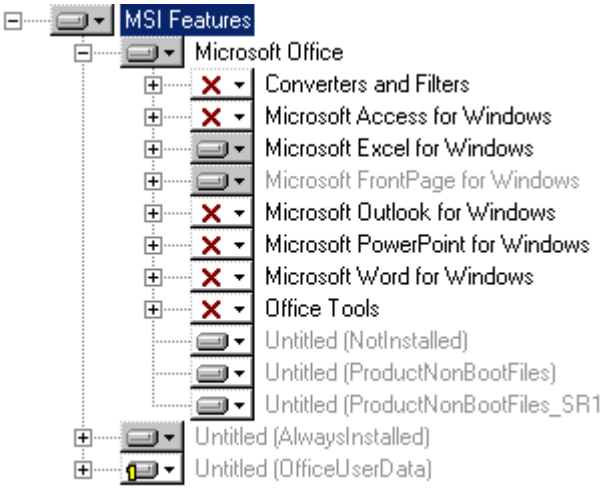
minus sign, and the next lower level features are displayed beneath the selected feature in the tree view, as follows:

Figure 54 Child features displayed



To close the display of child features beneath a particular feature, click on the minus sign in the small box next to the selected feature:

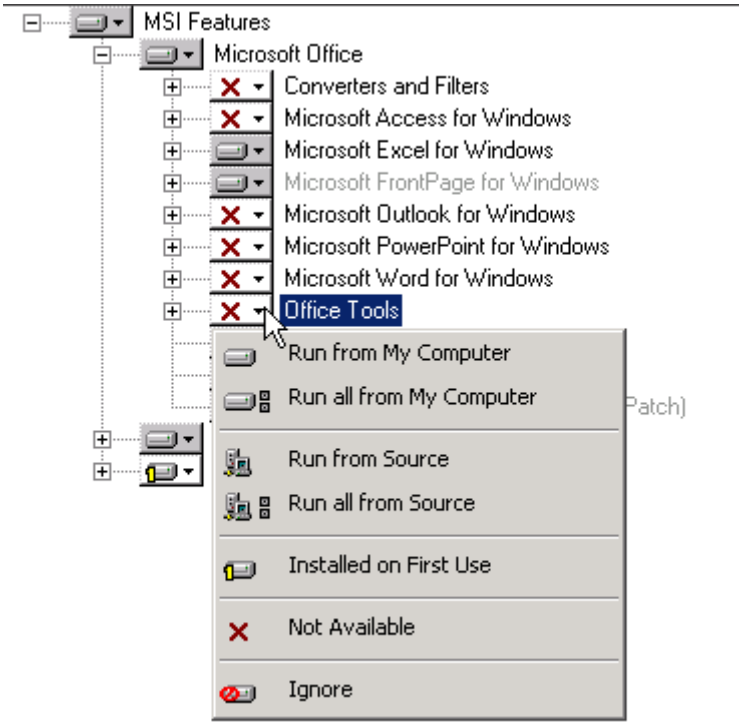
Figure 55 Child features closed under Microsoft Word



Features Menu

When you click on the drop-down button next to a feature, the Features menu opens.

Figure 56 Features menu



Each of the features has an attribute that indicates how that feature is to be installed and maintained. The image on the button that appears next to each feature reflects the attribute assigned to the feature.

Names of features whose attributes have been changed during the current session of the CM MSI Feature Set Editor appear in boldface in the tree.

Normally, by setting an attribute for a higher-level feature, the setting is automatically applied to all of the child features that comprise it. See the attribute descriptions in Table 31 on page 200 for exceptions to this rule.

Table 31 Features Menu Options

Feature Icon and Name	Description of Function
 Run from My Computer	Features with this attribute are physically installed on a hard drive on the agent computer, and therefore perform best. This attribute applies to all child features, unless the child feature's attribute is already set to Installed on First Use or Not Available. The selected feature's children appear on branches of the tree, and are beneath and connected to the selected feature.
 Run all from My Computer	Features with this attribute are physically installed on a hard drive on the agent computer. All features that are children of the selected feature have their attributes changed to Run from My Computer, and they, too, are physically installed on the agent computer. Features with this attribute perform best because they are run from the end user's own machine. Mobile computer users should have all of the features they require while traveling set to Run from My Computer or Run all from My Computer. Otherwise, when they go to use a feature that is not installed on their computer's hard drive, they will be prompted for the original installation media, which might not be available at that time.
 Run from Source	Features with this attribute are not physically installed on a hard disk on the agent computer; rather they are run from the Administrative Installation Point (AIP). This is normally a network share persistently accessible to the agent computer. When you select this attribute for a feature, the attribute is automatically applied to all features that are children of the selected feature, unless the child feature's attribute is already set to Not Available. The Run from Source and Run all from Source attributes save local hard drive space at the expense of slower performance and the requirement of having an active connection to the AIP.

Feature Icon and Name	Description of Function
 Run all from Source	Features with this attribute are not physically installed on a hard disk on the agent computer, rather they are run from the AIP. This is normally a network share persistently accessible to the agent computer. When you select this attribute for a feature, the Run from Source attribute is applied automatically to all features that are children of the selected feature, regardless of the attribute currently assigned to the child feature. The Run from Source and Run all from Source attributes save local hard drive space at the expense of slower performance and the requirement of having an active connection to the AIP.
 Installed on First Use	Features with the Installed on First Use attribute are not installed on the agent computer until the end user tries to use the feature through a shortcut or menu selection. Shortcuts for shortcut-accessible features are deployed when the package is initially installed. When the end user first invokes the feature, files that are required to support the feature are installed on the end user's hard disk, where they remain. Future uses of the feature are run from the agent computer. In order to install such a feature, the agent computer must be running the CM Redirector service, and it must be able to connect to the CM Configuration Server computer. Using this attribute minimizes the use of hard disk space on agent computers by allocating space only to features that they actually use. On the other hand, end users will be subject to the delay required to install the feature at the time when they first invoke it, which might surprise and displease them. When you select this attribute for a feature, the Installed on First Use attribute is applied automatically to all features that are children of the selected feature, unless the child feature's attribute is already set to Not Available. • See the Important Note on page 203 for additional information.

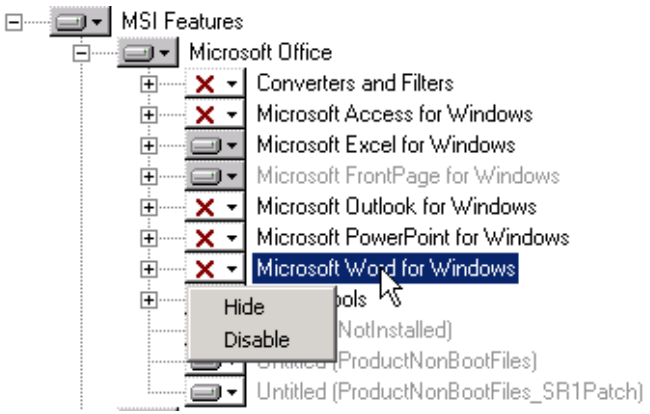
Feature Icon and Name	Description of Function
 Not Available	Features set to Not Available are not installed, are unavailable to the end user, but could affect other services. The attributes for all child features are also set to Not Available. For example, one service is defined to deploy MS Word, and all other features of Office are set to Not Available. Another service is defined to deploy MS Excel, and all other features of Office are set to Not Available. Now, imagine that the end user installs the MS Word service. It installs successfully and does not install any of the other features of Office. A bit later, the end user installs the Microsoft Excel service. The MS Excel service indicates that all other features of Office are Not Available. Therefore, MS Excel is installed, but MS Word is uninstalled. If the end user selects an option that requires the feature, the software instructs the end user to run the setup again and change the installation state of the feature. However, end users whose software is HP-managed generally do not have access to the setup. Assign the Not Available attribute only to features that must be unavailable to end users. Provide information to the end user about what to do if they receive a message.
 Ignore	**Note**: The Ignore feature is available for backward compatibility. You do not need to determine which features need to be ignored by a service. Features set to Ignore are not installed, are unavailable to the end user, but will not affect other services. CM management products act as though the feature were not a component of the managed software at all. The attributes for all child features are also set to Ignore. For example, one service is defined to deploy only MS Word, with all other features of Office set to Ignore. Another service is defined to deploy MS Excel and PowerPoint, setting all other features of Office to Ignore. If an end user installs both services, the features installed by the first-installed service will not be affected by the installation of the second-installed service. Both services can be created from one package published to the CM-CSDB. The package is cloned in the CM-CSDB by copying it with another name. You can then use the Feature Set Editor to edit the features of each package.

Names of features whose attributes have been changed during the current session of the CM MSI Feature Set Editor appear in boldface in the tree.

Hide and Disable Features

When you right-click the name of a feature (in the following example, Microsoft Word for Windows), the shortcut menu for Hide and Disable opens:

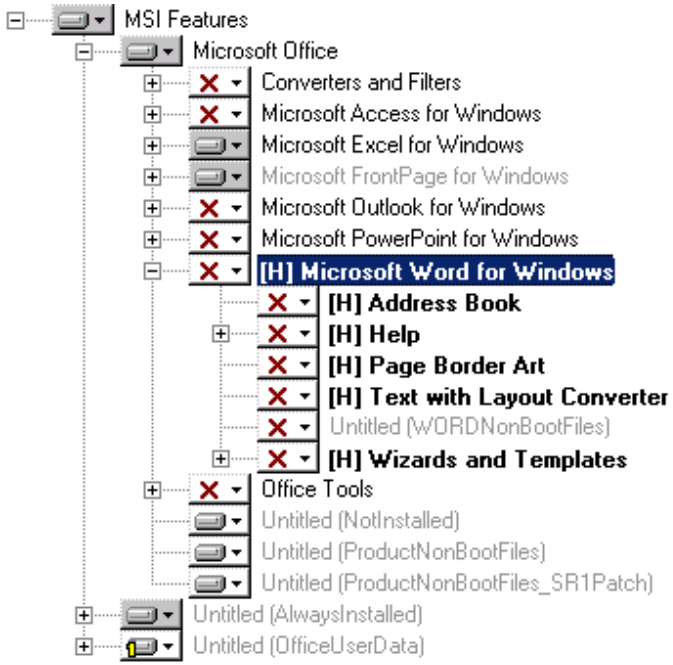
Figure 57 Shortcut menu for Hide and Disable



When you select **Hide**, the selected feature and all of its child features are prevented from appearing in the native Microsoft Windows Installer Update Features dialog box, which is accessible to the end user via the Control Panel's Add/Remove Programs applet.

All features that are hidden appear in the CM MSI Feature Set Editor tree with the prefix (H) before the name of the feature, as follows:

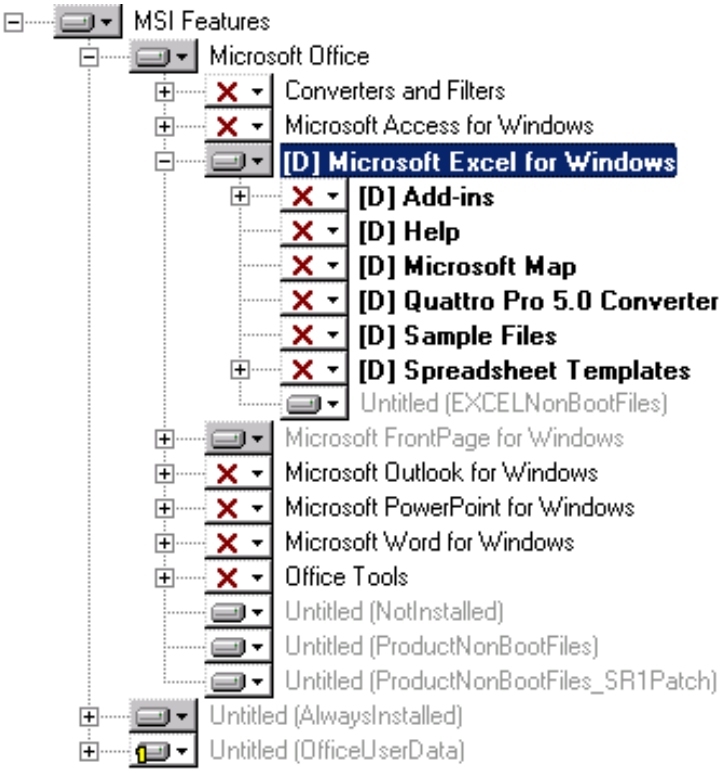
Figure 58 Hidden features have the prefix (H) before their names



When you select **Disable**, the selected feature and all of its child features are ignored by the CM agent. A disabled feature's value is determined by Windows Installer, from the combination of MSI and MST files, instead of the value in the CM instance.

All features that are disabled appear in the CM MSI Feature Set Editor tree with the prefix [D] before the name of the feature, as follows:

Figure 59 Disabled features have the prefix [D] before their names



Feature Set Editor Buttons

The Feature Set Editor buttons are located on the right side of the Feature Set Editor window. The following table describes these buttons.

Table 32 Feature Set Editor Buttons

Button	Description of Function
Save & Exit	Click **Save & Exit** to accept the changes that were made to the attributes of the file, and to then exit the Feature Set Editor. It is also possible to exit the Feature Set Editor by clicking the **X** on the top right corner.

Button	Description of Function
Exit (no Save)	Click **Exit (no Save)** to discard any changes that were made, and to then exit the Feature Set Editor. It is also possible to exit the Feature Set Editor by clicking on the **X** on the top right corner.
Defaults	Click **Defaults** to reset all features' attributes to the settings contained in the MSI file distributed with the software by the software vendor. The default setting will not be permanently applied to the feature set until the feature set is saved by clicking **Save & Exit**.
Reset	Click **Reset** to discard any changes made to feature settings since opening the current session of the CM MSI Feature Set Editor. Settings for all features are refreshed by re-reading them from the CM-CSDB.
Help	Click **Help** to access the CM Admin CSDB Editor Help topics.

Package Information

Use the package information to view the size of the current Windows Installer-enabled features that you are configuring.

Maximum Size

This is the total number of bytes that would be required on the agent computer to install all features in the entire feature tree, if they were all given the Run from My Computer or Run All from My Computer attribute.

Total Size

This is the total number of bytes required on the agent computer to install all features in the entire feature tree that have the Run from My Computer or Run All from My Computer attribute.

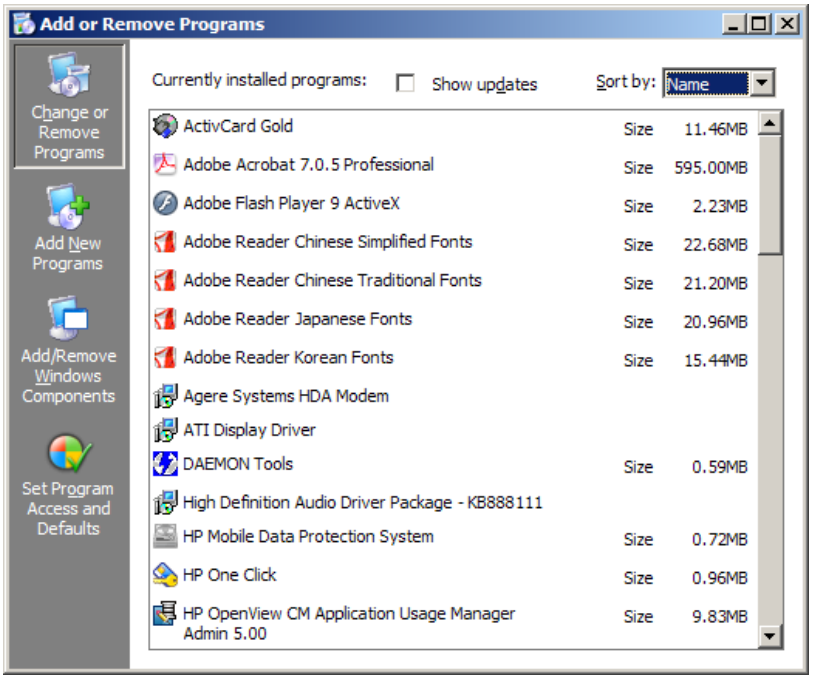
Selected Size

This is the total number of bytes required on the agent computer to install the selected (i.e., highlighted) feature and its child features. The size calculation includes only features with the Run from My Computer or Run All from My Computer attribute.

Windows Add or Remove Programs Settings

When a Windows Installer-enabled software package, such as MS Office, is installed on an agent computer, it is normally added to the list of applications that can be manipulated by the Add or Remove Programs application of Windows Control Panel, as seen here:

Figure 60 Add or Remove Programs dialog box

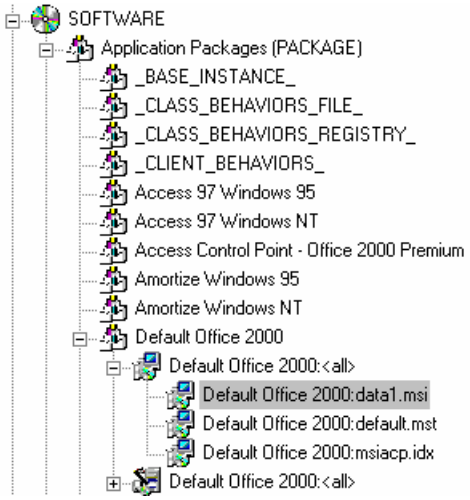


CM gives you the option of not listing Microsoft Office in the Add or Remove Programs application of Windows Control Panel. This prevents the end user from interfering with CM products' centralized management of the application. Alternatively, CM also allows you to selectively hide individual features from access by the Add or Remove Programs application.

Eliminating an Application's Listing in Add or Remove Programs

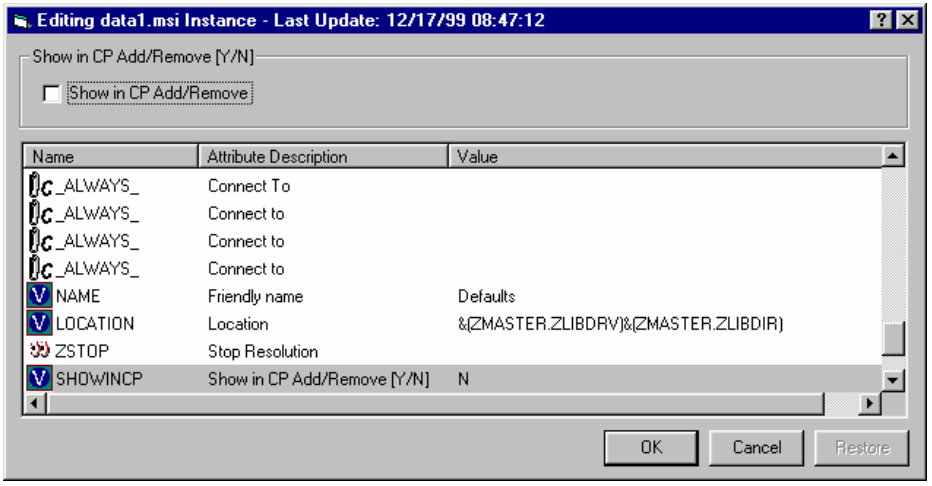
To completely prevent listing the application in the Add or Remove Programs application, use CM Admin CSDB Editor to change the setting of the SHOWINCP variable of the MSI class instance for the DATA1.MSI file that is connected to the Application Packages Class instance:

Figure 61 DATA1.MSI instance



Change the value of the SHOWINCP variable to N, as follows:

Figure 62 Editing DATA1.MSI instance



Hiding Features displayed by Microsoft Windows Installer Update Features

When the end user selects the Windows Installer-enabled software package in the Add or Remove Programs application and clicks **Change**, Microsoft Windows Installer is invoked, and presents its native user interface for manipulating features of the installation on the agent computer. If the end

user clicks **Add or Remove Features**, Microsoft Windows Installer displays its Update Features windows.

Use the CM MSI Feature Set Editor's Hide function to prevent individual features from appearing in the Microsoft Windows Installer Update Features dialog box. This enables you to maintain control over the end user's installation of the application software's features via your CM management product's capabilities, and to avoid possible end-user interference.

Copying a Windows Installer Enabled Package

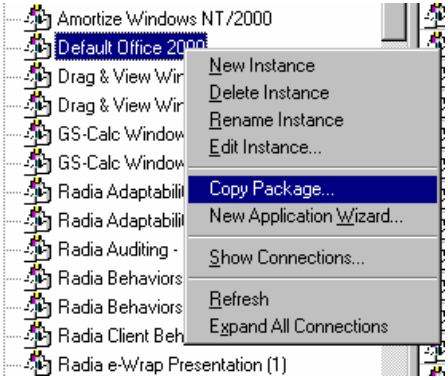
One of the key advantages of using CM products to manage Windows Installer-enabled applications is the ability to easily specify variants of the software package, and make those variants available to targeted populations of users.

Using the advanced capabilities of CM, you need only create one comprehensive AIP, and promote it to the CM-CSDB. All variant sets of features are encapsulated in individual packages in the CM-CSDB.

To make a new variant set of features

Clone an existing package, and edit it to meet its unique requirements.

1 Right-click on the Application Packages instance and select **Copy Package** from the resulting shortcut menu.



A message opens that says that copying a package instance will result in a duplication of all the package's component instances, and that this may take a while.

2 Select **Yes** to continue the Copy Package function.

Select **No** to cancel the Copy Package function.

In this example, **Yes** was selected.

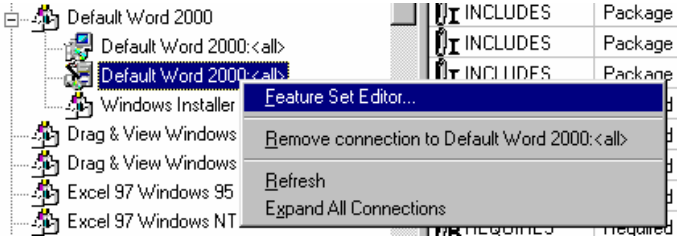
Copying the Package

Copying the package copies the package instance and all of the component class instances that belong to the package instance. The resulting package is a complete clone of the original. This operation maintains the integrity of the CM-CSDB, because each package retains control over its own components.

3 In the Copy Package and Components dialog box, type the names for the newer copied package:

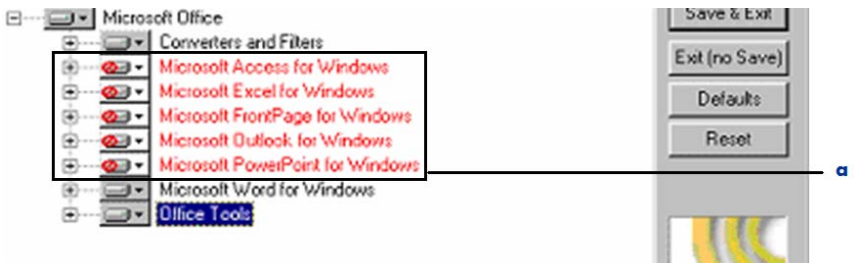
4 Click **OK** to continue. The CM Configuration Server takes a few moments to clone the package in the CM-CSDB.

5 When the newly cloned package appears in the CM Admin CSDB Editor's tree view, right-click on the MSIFEATS instance beneath the package in the tree view, and select **Feature Set Editor** from the shortcut menu:



The CM MSI Feature Set Editor opens.

6 Set the properties of the applications that should not be deployed to Ignore:



Legend

a Ignored applications

7 Click **Save & Exit**, and the feature set for the package is updated. You can now connect this package to the appropriate policy (USER, DEPT,

WORKGRP) instances to deploy and manage a single feature as a stand-alone application.

Managing Feature Sets in Native Mode

In some enterprises, packaging applications and administering CM are separate functions. You might not want to use the CM MSI Feature Set Editor to edit the transform (MST) files. To manage feature sets in native MSI mode, publish the package and MST files using the following procedure:

1. Use the CM Admin Publisher in Advanced mode to create the ACP and publish the Windows Installer Package.

   After completing the publishing process, use CM Admin CSDB Editor to remove the connection to SOFTWARE.MSIFEATS.packageprefix_* from the Application Packages (PACKAGE) instance.

2. If you have multiple MST files, use the CM Admin Packager in Component Selection Mode to promote each of the MST files into individual packages. This will create a new PACKAGE instance for each MST file.

3. Connect the MSI PACKAGE instance to each of the transform file's PACKAGE instances.

4. Create one Application (ZSERVICE) instance for each of the transform packages.

5. Assign subscribers to the appropriate transform package's service.

Any changes to or replacement of the `*.MST` file will cause CM to call MSIEXEC to uninstall and reinstall the application.

When the service is installed or modified, CM will detect the absence of the MSIFEATS object, and will use the native MSI mode, MSIEXEC. CM will call MSIEXEC natively with the proper location of the data source. If CM calls MSIEXEC and detects the presence of a single MST file, it will add the `.MST` file to the command line that invokes MSIEXEC. If CM calls MSIEXEC and detects more than one MST file, it will run MSIEXEC for the MSI file only. The MSI native behavior will be used for update and repair.

If you are using an MST file to include a new feature that is not already a feature available in the CM Feature Set Editor (MSIFEATS), the new feature from the MST file will be included in the MSI service. However, if there is a feature in the MST file that already exists in the MSIFEATS object, the feature state indicated in MSIFEATS will be used, and the one in the additional MST file will be disregarded. If you want to manage the MSI package using only your MST file, use the procedure outlined in this section.

Summary

- Use the Feature Set Editor to configure each of the features of a Windows Installer-enabled software package.
- Features are grouped in a hierarchy. High-level features consist of two or more lower-level features.
- Each of the features has an attribute that indicates how that feature should be installed and maintained.
- Alternative procedures allow you to manage feature sets in native MSI mode, if desired.

9 Registry File Editor

At the end of this chapter, you will:

- Understand the Registry File Editor.
- Know how to edit data in a registry key.

The Registry File Editor

Use the **Registry File Editor** to edit the properties or values for registry keys associated with a CM package.

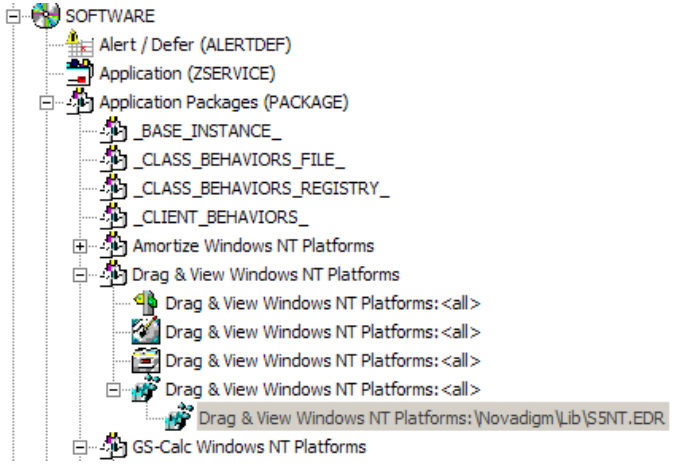
When using CM Administrator Packager (CM Admin Packager) to create a package, all of the registry changes for that package are stored in a file (with an `.EDR` extension) in the `LIB` directory. Use the Registry File Editor to modify this file. Then, the file is sent (or re-promoted) to the CM Configuration Server and the instance in the CM-CSDB is updated.

See the *HP OpenView Configuration Server Application Manager Installation and Configuration Guide (CM Application Manager Guide)* for information on creating and promoting packages.

Accessing the Registry File Editor

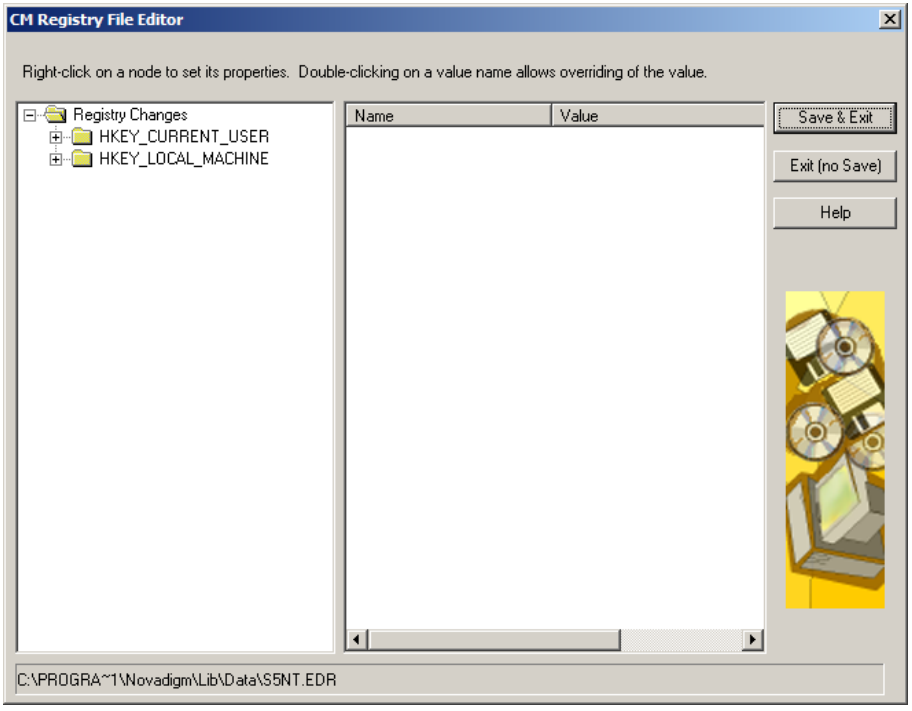
To access the Registry File Editor

1. Locate the Registry Resource instance that represents the software package whose registry settings you wish to change in the CM-CSDB.
2. Right-click the instance in the tree view. A shortcut menu opens. For example, locate the registry instance for the Drag & View Windows NT Platforms package and right-click on it.



3. Select **Edit Registry Resource**.

The Registry File Editor dialog box opens.



- The tree view (on the left) displays nodes that represent keys in the registry file.
- The list view or value name (in the middle) displays the contents of a selected registry key.
- The buttons (on the right) allow you to Save & Exit, Exit without saving, or access the Help function in the Registry File Editor.

Expanding and Collapsing the Tree

Nodes are grouped into a hierarchy. When you set a property to a node, the Registry File Editor applies changes to all nodes within that node's branch in the tree view.

Use the plus and minus signs to expand and collapse the tree view.

Set Properties Shortcut Menu

To set properties for the Registry File Editor, right-click on any node of the tree view in the Registry File Editor. A shortcut menu opens.

For specific information regarding the options available through this shortcut menu, see the following table.

Table 33 Registry File Editor shortcut menu options

Action	Description
Active	Select **Active** to include the registry keys in the package.
Inactive	Select **Inactive** to prevent the registry keys from being included in the package. Inactive nodes appear with a symbol covering its folder.
Deploy/Verify	Click **Deploy/Verify** to access the following submenu: • **Unconditional/Enforced** CM deploys the registry key every time the subscriber connects to the CM Configuration Server, even if it already exists on his desktop. This setting repairs the registry key if it is modified or deleted. Use this setting for registry keys that must remain invariant in order for the application software to function properly. • **Unconditional/Exist** CM deploys the registry key upon initial deployment or if it has been deleted. However, the value of the registry key is not reset if the user modifies it. This is the default setting. • **Exist/Exist** CM deploys the registry key only if it does not exist on the subscriber's desktop, during both the initial deployment and on each subsequent connection. This gives users full control over the registry key value, but reinstalls the key if it is accidentally deleted.
New Key	Click **New Key** if you would like to add a new value to your registry. Clicking **New Key** will display a dialog box to enter the New Key name. Click **OK** to accept the new key, or **Cancel** to return to the Registry File Editor.

Action	Description
New Value	Click **New Value** to add a value to a newly created key or to add a value to an existing registry key. Clicking **New Value** opens a dialog box where you will be required to enter the following information: **Value Name** Enter the name of the value you will be adding to the registry file. This name will appear in the Value Name (center) of the Registry File Editor window. **Data Type** Use the drop-down menu to select from the list of available data types. The type selected is as follows: • **REG_SZ** A text sequence most typically used for descriptive information. This is the default setting. • **REG_DWORD** Is data represented by an integer that is 4 bytes long and is displayed in the binary, hexadecimal, or decimal format. • **REG_EXPAND_SZ** Is an expandable data string. This text contains a variable that is replaced when called by an application. • **REG_MULTI_SZ** A string containing multiple text entries of user readable text.
Delete Always	Click **Delete Always** if you want all values under a Registry key to be deleted during a Radreg32 Delete operation. When this option is turned on, all values under the key are deleted *without* restoring their original registry values in the backup EDR file.
Delete Never	Click **Delete Never** if you want to retain all values under a key during a Radreg32 Delete operation.
Delete (Default)	Click **Delete (Default)** if you want the Radreg32 Delete operation to exhibit normal Delete behavior with regards to deleting or restoring values.

Editing a Registry Key

To edit a registry key's value access the key through the Registry File Editor. Follow the example in this section to edit the information in a registry key.

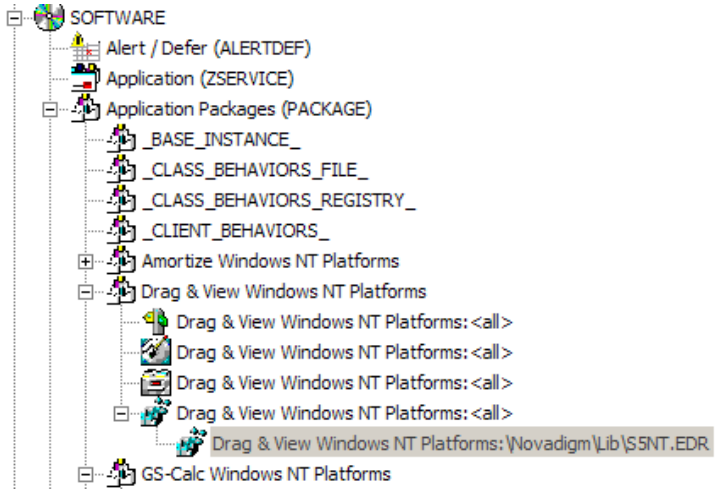
Use extreme caution when you use the Registry File Editor. Editing registry keys without thorough understanding of their functions could render a package or service useless.

To edit the data in a registry key

1. From the Windows taskbar, click **Start → Programs → HP OVCM Administrator → CM Admin CSDB Editor**. The CM Admin CSDB Editor Security Information dialog box appears.

   The factory set user ID is RAD_MAST. No password is necessary. Your CM security administrator might have changed this during installation. Check with him or her to obtain your own user ID and password, if necessary.

2. If necessary, type a user ID and password, then click **OK**. The CM Admin CSDB Editor splash screen appears briefly.
3. Double-click **PRIMARY**.
4. Expand the SOFTWARE Domain and then the Application Packages (PACKAGE) Class.
5. Expand the package Drag & View Windows NT Platforms.
6. Expand the Registry instance.

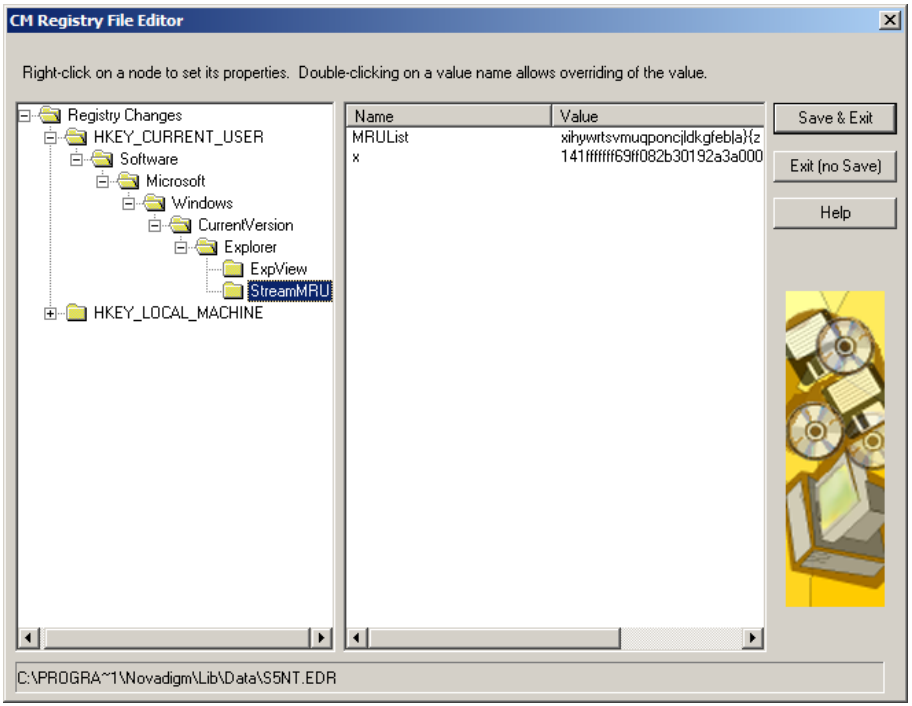


7. Right-click the registry file and select **Edit Registry Resource** from the shortcut menu.

   The CM Registry File Editor opens.

8 Click the node to expand and display the contents of the registry key in the list view.

9 Double-click the name of the registry key in the list view.



The Override Value dialog box opens.

10 Edit the data in the New Value text box.

11 Click **OK** to save the changes and close the dialog box.

or

Click **Cancel** to close the dialog box without saving the changes.

To restore the data in a registry key

1 Double-click the name of the registry key in the list view.

2 Click **Reset**.

3 Click **OK** to save the changes and close the dialog box.

or

Click **Cancel** to close the dialog box without saving the changes.

Closing the Registry File Editor

Close the Registry File Editor when you are finished making changes to instances of the Registry file.

To close the Registry File Editor

1. Click **Save & Exit** to save the changes and close the dialog box.

   or

   Click **Exit (no Save)** to discard the changes and close the dialog box.

2. Click **Yes** to confirm that you want to save the changes and close the dialog box.

   or

   Click **No** to return to the Registry File Editor.

Summary

- Use the Registry File Editor to edit the properties or values for registry keys associated with a CM package.
- The Registry File Editor applies changes to all nodes within that node's branch in the tree view.
- After you have edited a registry key's data, you can restore it to its original value.

A Modifying Component Instances (Non-Production Only)

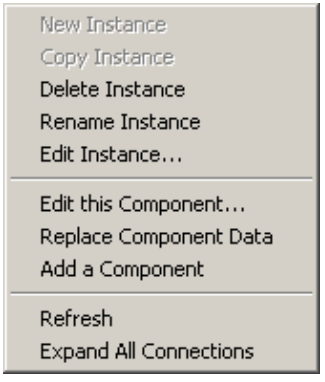
At the end of this appendix, you will:

- Understand the restrictions on using the advanced options to Add, Edit, and Replace file-type components.
- Be able to dynamically edit the distributed contents of a file from within the CM Admin CSDB Editor.
- Be able to replace the entire contents of the data distributed for a file within a published package in the CM Admin CSDB Editor.
- Be able to add a component instance for a file or behavior to a software package, without having to republish the package.

HP does not support this feature in production environments, and assumes no responsibility if a customer uses this feature where packages are actively being deployed to agents in a production environment. Using this feature in such an environment would likely result in massive distributions of unnecessary package updates, since CM would detect a package change and automatically distribute the change to all authorized agents.

Figure 63 below shows the shortcut menu commands for component modifications.

Figure 63 Instance shortcut menu



When to Use these Commands

Examples of when you might want to use this option to modify component instances are listed in the following table.

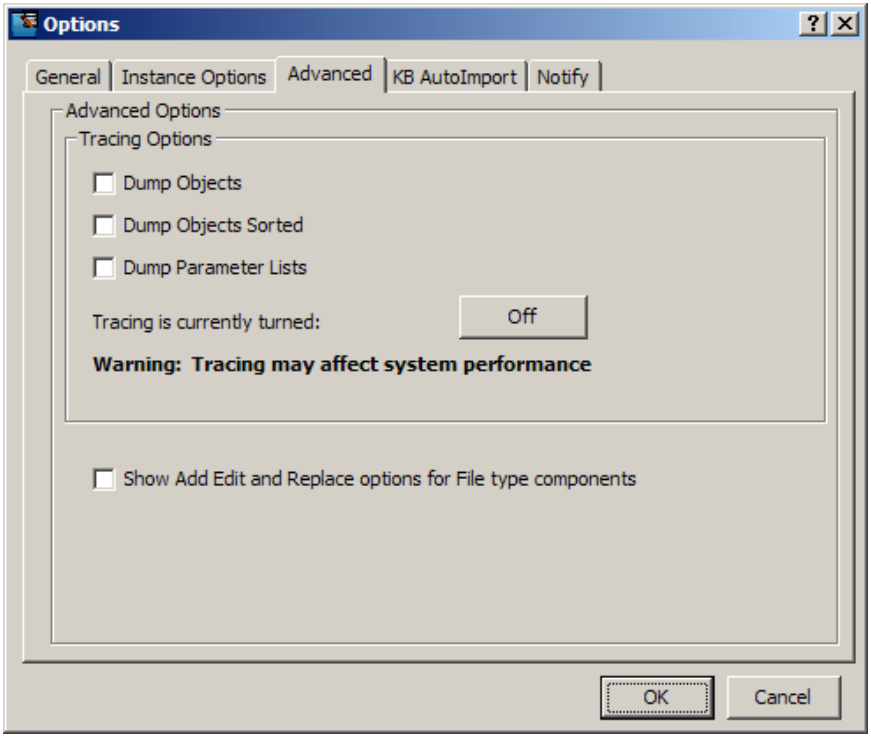
Table 34 When to Use the Advanced Edit Component commands

Command	Typical Use
Edit this Component	Change a serial number deployed with a file. Edit the contents of a `readme` file, document, spread sheet, logo image, or PDF.
Replace Component Data	Replace the entire contents of an existing file in a package with contents of another file (outside of CM-CSDB).
Add a Component	Add a new file instance to a package, using another packaged file as a template.

Accessing the Add, Edit, and Replace Component Commands

The following setting is available on the Advanced tab of the Options dialog box: Show Add Edit and Replace Options for File type components.

Figure 64 Advanced tab of the Options window

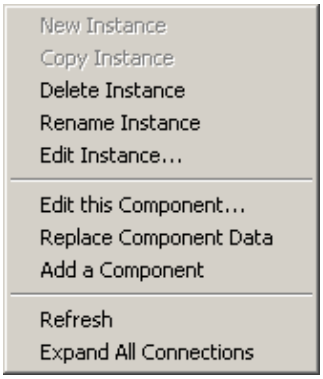


For non-production environments only, select **Show Add Edit and Replace options for File type components** to enable these commands on the shortcut menu for applicable instances.

HP does not support this feature in production environments, and assumes no responsibility if a customer uses this feature where packages are actively being deployed to agents in a production environment. Using this feature in such an environment would likely result in massive distributions of unnecessary package updates, since CM would detect a package change and automatically distribute the change to all authorized agents.

Selecting this option displays the following component modification commands on the shortcut menus for component instances.

Figure 65 Instance shortcut menu



Editing a Component

The **Edit this Component** option permits you to select an editor of your choice with which to dynamically modify the distributed contents of a file instance. After you save the changes to the file, the modified file is sent (re-promoted) to the CM Configuration Server and the instance in the CM-CSDB is updated.

HP does not support this feature in production environments, and assumes no responsibility if a customer uses this feature where packages are actively being deployed to agents in a production environment. Using this feature in such an environment would likely result in massive distributions of unnecessary package updates, since CM would detect a package change and automatically distribute the change to all authorized agents.

For example, you can edit the distributable contents of any component instance that has been packaged and is represented in the CM Admin CSDB Editor. You must save the edited file with the same name as the original file. The modified file immediately replaces the existing file in the package. The CM Admin CSDB Editor takes care of the publishing for you, saving you from having to republish the package.

The original component's contents are not retained in the CM Admin CSDB Editor.

Refer to the *HP OpenView Configuration Management Application Manager Installation and Configuration Guide* for information on creating and promoting packages to the CM-CSDB.

To edit a component

The component editor does not know a component's target operating system. Before you edit an instance, HP recommends that you note (or export) the values of the following attributes, as well as any other attributes, whose values should not change due to your edits:

ZRSCMMEM (PDS Member Name)
ZRSCCRC (Resource CRC)
ZRSCPADM (PDS AdminID)
ZPUBVER (Packager/Publisher Version)

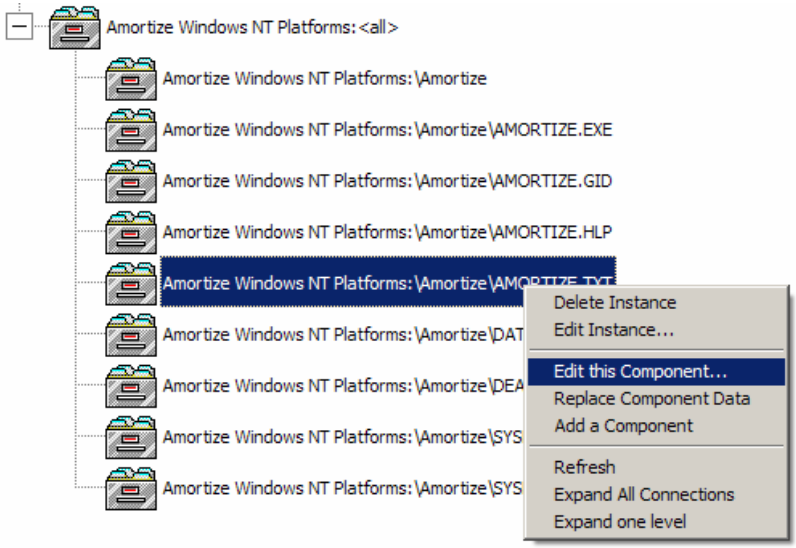
If the values of these attributes change as a result of the editing process, we recommend that you restore the original values.

For example, if a component is targeted for a Windows operating system, the ZRSCMEMM, ZRSCCRC and ZRSCPADM attributes do not apply and should be null. After editing a component, if these attributes show values other than null, you should clear the inserted values. Likewise, the values for other attributes, such as ZPUBVER, should remain unchanged due to an edit. If changed, restore ZPUBVER to its original value.

1. In the CM Admin CSDB Editor, go to **Options** → **Advanced** tab, and select **Show Edit Replace and Add Component options for File type components**.
2. Locate the file instance that belongs to the software package whose contents you want to change in the CM-CSDB.
3. Right-click the instance in the tree view.

   A shortcut menu opens.

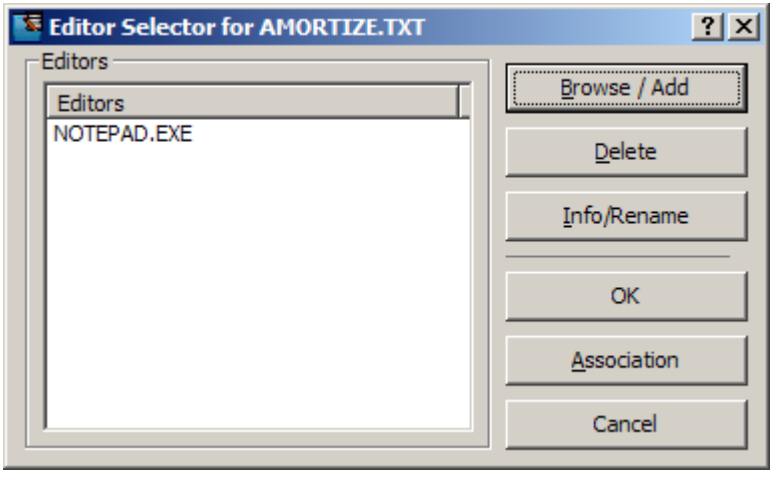
   The following figure shows an example of the shortcut menu for the `Amortize.TXT` readme file instance within the Amortize NT Platforms package. It includes the commands: Edit this Component, Replace Component Data, and Add a Component.



4 Select **Edit this Component** from the shortcut menu.

The Editor Selector for AMORTIZE.TXT dialog box opens.

Use this dialog box to select the resident editor you want to use to make the changes to this file. For example, double-click **Notepad** to open the `AMORTIZE.TXT` file with `Notepad.exe`.



For details on using this dialog box, see Using the Editor Selector Dialog Box on page 232.

5 Double-click an editor from the Editors list box.

The selected editor is launched and opens the previously selected file component instance.

6 Make the required changes to the file's contents and save it with the *same name*.

If you save the file with a different name, the CM component editor cannot make the changes to the file you selected from the CM Admin CSDB Editor.

7 Close the editor.

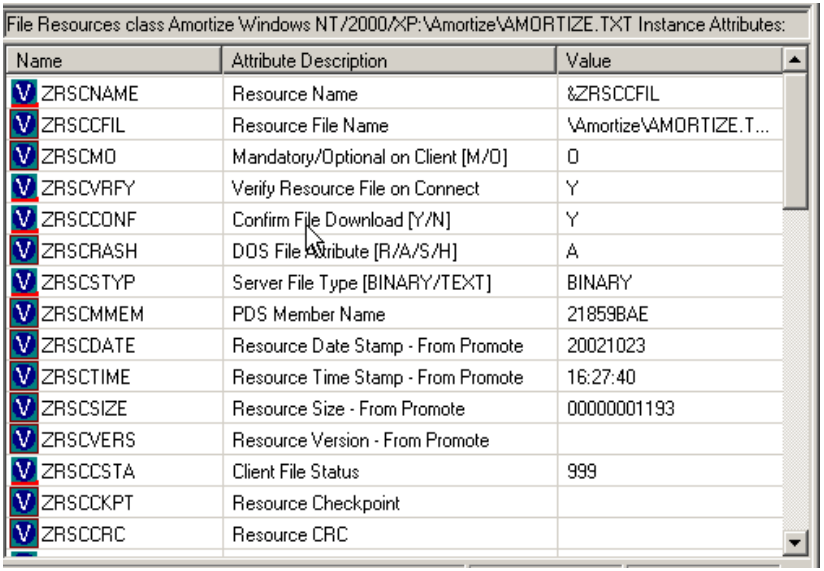
8 If necessary, click **OK** to close the Editor Selector dialog box and save the changes to the instance.

9 The File Re-Promote dialog box opens, and requests a confirmation that you want to commit the changes to the selected instance in the database.

— Click **Yes** to save the changes, perform a mini-publish and re-promote the changed file to the database.

— Click **No** to cancel the edits and the edit session. No changes are made to the database instance.

After you commit the changes, the instance attributes for ZRSCDATE ZRSCTIME, and ZRSCSIZE reflect the edited file size and date.



HP recommends that you compare all attributes' values against

their original ones, and restore any values that changed unnecessarily. For example, you may need to clear values from OS-related attributes whose values were originally null. For details, see the cautionary text at the beginning of this procedure on page 228.

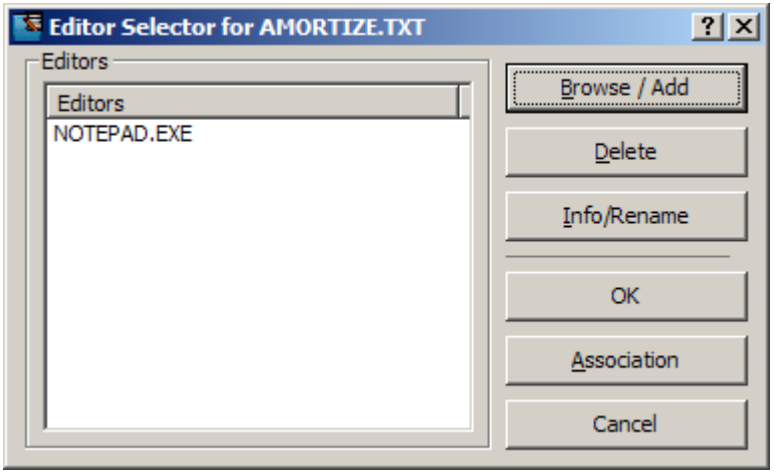
Using the Editor Selector Dialog Box

Use the Editor Selector dialog box to select a resident editor with which to edit the named file.

To select an editor, choose an entry from the Editors list, or click **Association** to use the Windows default editor associated with file's extension.

To add, delete, or rename the editors displayed in the Editors list, use the top three buttons on the right.

Figure 66 Editor Selector dialog box to edit a component



The following table summarizes the Editor Selector dialog box button actions.

Table 35 Editor Selector Dialog Box Buttons

Button	Action
Browse/Add	Opens a File Browse dialog box that you can use to select a program to add to the list of editors. Adds an entry from the Browse dialog box to the Editors list below the highlighted entry.
Delete	Deletes a selected (highlighted) program from the Editors list.
Info/Rename	For the selected editor program, opens a dialog box that displays the

Button	Action
	editor's program name and path, and allows you to assign a friendly name to display on the Editors list.
OK	Opens the selected file (named in the title bar) with the highlighted editor. Alternatively, you can double-click on the Editors list-box entry.
Association	Opens the selected file (named in the title bar) with the default editor associated with that file type through Windows. If no association is available, you will be returned to the Editor Selector dialog box where you can specify an editor or cancel.
Cancel	Cancels the Edit a Component command without making any changes to the selected component or package.

Replacing Component Data

Use the Replace Component Data command when you need to replace the entire contents of a packaged file with the contents of a file that exists outside of the database. For example, suppose there is a new `Readme.TXT`, and you want to completely replace its contents. Prepare a revised `Readme.TXT` file outside of the CM Admin CSDB Editor. Then, use the Replace Component Data command from within the CM Admin CSDB Editor to replace the contents of the existing `Readme.TXT` with the new one. When you use the Replace Component Data command, none of the properties of the packaged file will change, just its distributed contents.

HP does not support this feature in production environments, and assumes no responsibility if this feature is used where packages are actively being deployed to agents in a production environment. Using this feature in such an environment would likely result in massive distributions of unnecessary package updates, since CM would detect a package change and automatically distribute the change to all authorized agents.

To replace [all] component data

1 Prepare a file outside of CM Admin CSDB Editor whose contents represent the replacement data for a packaged component file in the CM-CSDB. The prepared file can have any name.

The entire contents of the prepared file will replace the entire data contents of the file selected from the CM Admin CSDB Editor using the Replace Component Data command. Verify the contents of the files prior to continuing with this procedure.

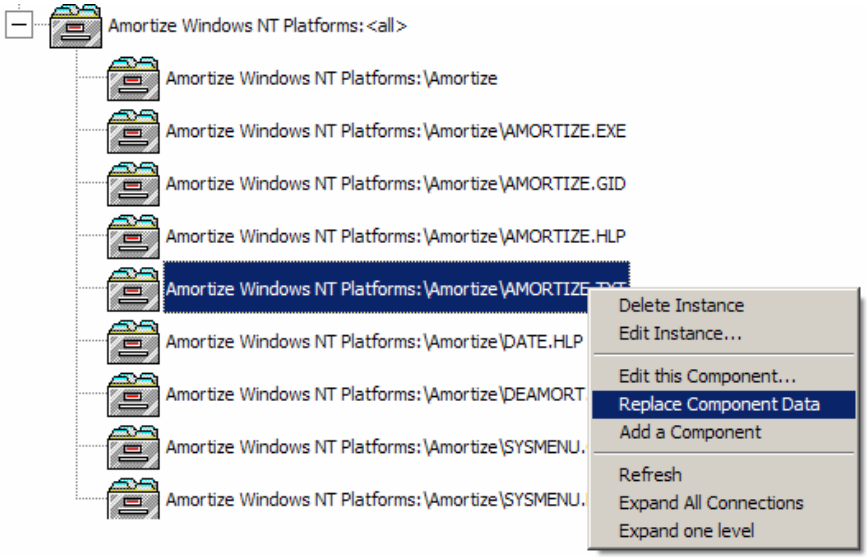
2 In the CM Admin CSDB Editor, go to **Options** → **Advanced** tab, and select **Show Edit Replace and Add Component options for File type components**.

3 Locate the file instance in the software package whose data you want to replace in the CM-CSDB.

4 Right-click the instance in the tree view.

A shortcut menu opens.

The following figure shows an example of the shortcut menu for the `Amortize.TXT` readme file instance within the Amortize NT Platforms package. It includes the commands: Edit this Component, Replace Component Data, and Add a Component.

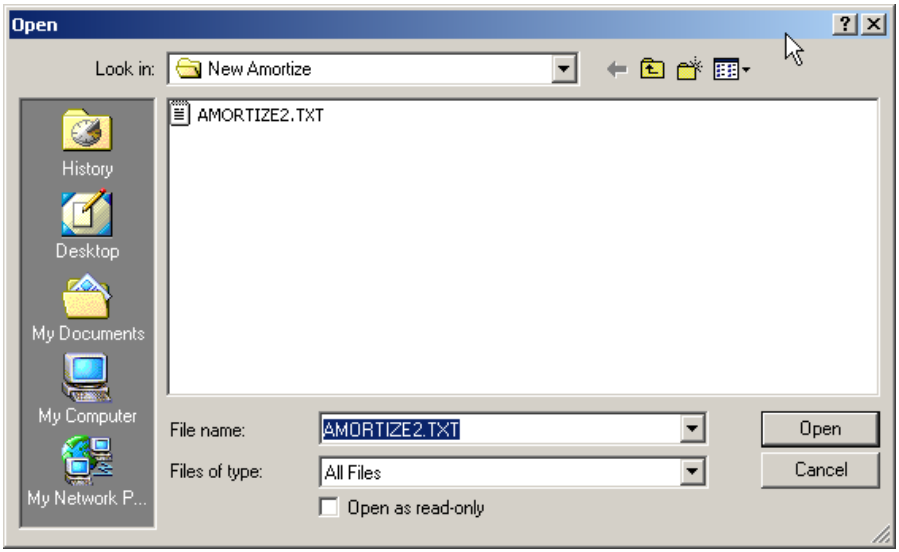


5 Select **Replace Component Data** from the shortcut menu.

A standard Windows Open dialog box opens.

6 Navigate to the prepared file whose contents will replace the data for the file you selected from the CM Admin CSDB Editor.

The following figure shows an example of navigating to a prepared file named `AMORTIZE2.TXT`.



7 Click **Open** to select the file with the replacement data.

A confirmation message opens.

8 Click **Yes** to replace the data; click **No** to abort the data replacement.

If you click **Yes**, the CM Admin CSDB Editor substitutes the entire contents of the data to be distributed with the selected file's data, and republishes the package.

The instance attributes for ZRSCDATE, ZRSCTIME, and ZRSCSIZE reflect the new file size and date, as previously illustrated.

If you click **No**, the replacement command is cancelled.

Adding a Component

The **Add a Component** command is available for file and behavior components only. Files always belong to an owning package, so the most likely place to perform an Add is from the connection under a package. New files can be added by copying the properties of an existing published file; files cannot be created from scratch, that is, from files that have not been promoted from the CM Admin Packager.

HP does not support this feature in production environments, and assumes no responsibility if this feature is used where packages are actively being deployed to agents in a production environment. Using this feature in such an environment would likely result in

massive distributions of unnecessary package updates, since CM would detect a package change and automatically distribute the change to all authorized agents.

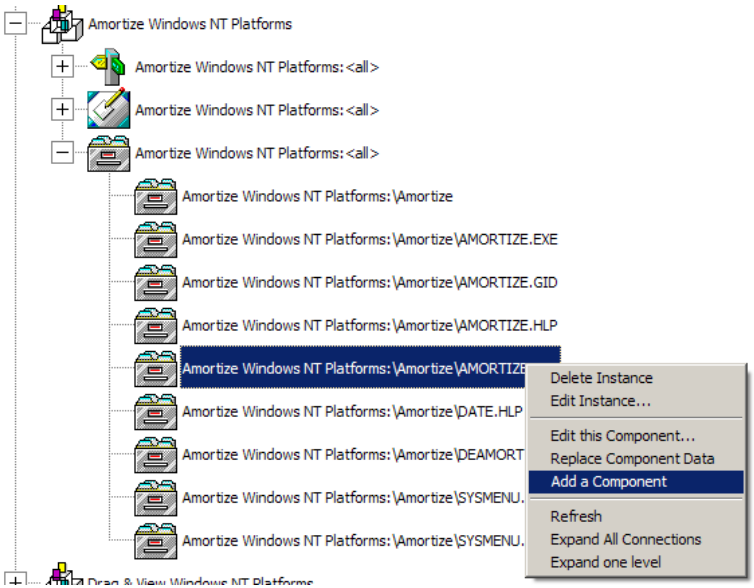
New files are added by selecting an instance to use as a template. The CM Admin CSDB Editor will derive the path for the new instance from the template's LOCATION variable. After a template is selected, right-click it to display the shortcut menu and select **Add Component**. An Open File dialog box opens where you can select the file you want to add to your package. A final confirmation box will allow you to stop the process if you need to. Click **Yes** in the box to promote the file that was selected in the Open File dialog box. The package now contains the new file.

To add a component to a package

1. In the CM Admin CSDB Editor, go to **Options** → **Advanced** tab, and select **Show Edit Replace and Add Component options for File type components**.
2. Locate a file instance in the software package whose LOCATION can serve as a template for the component to be added.
3. Right-click the template file's instance in the tree view.

   A shortcut menu opens.

   The following figure shows an example of the shortcut menu for the `Amortize.TXT` readme file instance within the Amortize Windows NT Platforms package. It includes the commands: Edit this Component, Replace Component Data, and Add a Component.

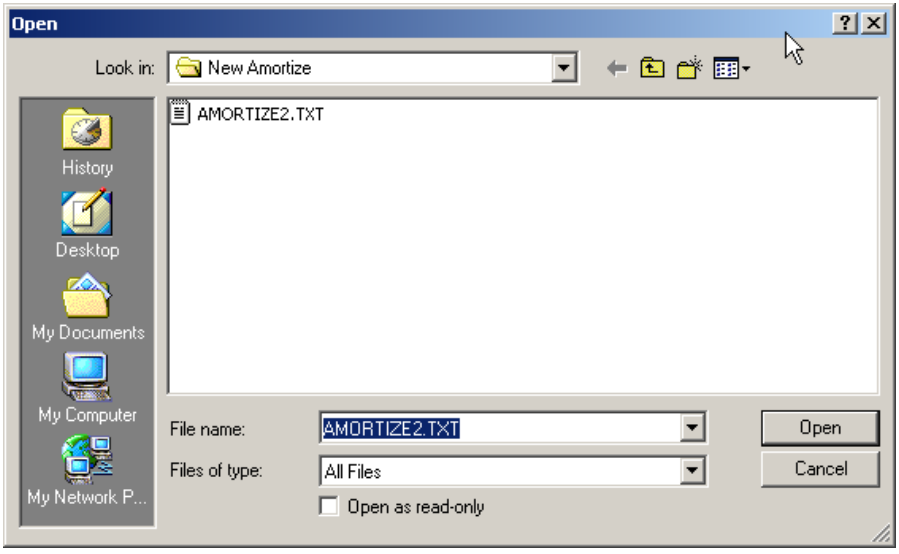


4 Select **Add a Component** from the shortcut menu.

A standard Windows Open dialog box opens.

5 Navigate to the prepared file you want to add to the package.

The following figure shows an example of navigating to a prepared file named `AMORTIZE2.TXT`. In this case, we want to add this component to the package.



6 Click **Open** to select the file to be added with the same LOCATION as the template file.

A confirmation message opens.

7 Click **OK** or **Cancel**.

If you click **OK**, the CM Admin CSDB Editor promotes the added file and republishes the package. The instance attributes for the added file's LOCATION reflect the template file's location.

If you click **No**, the Add a Component command is cancelled.

Summary

- Use the Advanced Options that allow you to add, edit, and replace file-type components in non-production environments only.
- Use Edit this Component to dynamically modify the distributed contents of a component from within the CM Admin CSDB Editor, without having to republish the entire package.
- Use Replace Component Data to replace all of the distributed data in a component instance with the contents of a prepared file outside of the CM-CSDB. You do not have to republish the package using this command.
- Use the Add a Component command to add an instance to a package without having to promote the file and re-publish the package. An existing instance must be able to serve as a template for the added component's path LOCATION.

B Product Name Changes

If you have used Radia in the past, and are not yet familiar with the newly rebranded HP terms and product names, Table 36 below will help you identify naming changes that have been applied to the Radia brand.

Table 36 Product Name and Term Changes

New Name/Term	Old Name/Term
CM agents	Radia clients
HP OpenView Configuration Management Administrator	Radia Administrator Workstation
HP OpenView Configuration Management	Radia
HP OpenView Configuration Management Admin Agent Explorer	Radia Client Explorer
HP OpenView Configuration Management Admin CSDB Editor	Radia System Explorer
HP OpenView Configuration Management Admin Packager	Radia Packager
HP OpenView Configuration Management Admin Screen Painter	Radia Screen Painter
HP OpenView Configuration Management Application Manager	Radia Application Manager
HP OpenView Configuration Management Application Self-service Manager	Radia Software Manager
HP OpenView Configuration Management Application Usage Manager	Radia Usage Manager
HP OpenView Configuration Management Configuration Server	Radia Configuration Server
HP OpenView Configuration Management Configuration Server Database	Configuration Server Database, Radia Database
HP OpenView Configuration Management Extensions for Windows Installer	Radia Extensions for Windows Installer

New Name/Term	Old Name/Term
HP OpenView Configuration Management Inventory Manager	Radia Inventory Manager
HP OpenView Configuration Management Patch Manager	Radia Patch Manager
HP OpenView Configuration Management Reporting Server	Radia Reporting Server
HP OpenView Configuration Management Solutions for Servers	Server Management

Index

D

E

F

G

H

I

K

L

M

N

O

P

R

S

T

U

V

W

Z