HP OpenView Configuration Management Application Manager

for HP-UX, RedHat Linux, SuSE Linux, Solaris, and AIX operating systems

Software Version: 5.00

Installation and Configuration Guide

Document Release Date: April 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 lists new features added for the Configuration Management 5.0 release.

Table 1 New features added for Configuration Management 5.0

Chapter	Version	Changes
2	5.00	Page 30, System Requirements: Updated System Requirements for version 5.0.
2	5.00	Page 34, To install the CM agent: Updated installation: • New panel to prompt user when to start CM Daemons. • New panel to prompt for WBEM server libraries link creation and search path.
3	5.00	Page 50, System Requirements: Updated System Requirements for version 5.00.
5	5.00	Page 94, Published Owner, Group, and Permission Considerations. Added DIRPERMS attribute information for the UNIXFILE class.
10	5.00	Page 243, Controlling Default Permissions for Directories and Objects. Added information for controlling the default permissions of objects and directories created by CM.

Table 2 indicates changes made to this document for earlier releases.

Table 2 Document changes

Chapter	Version	Changes
6	4.1	Configuring Client Operations Profiles added. Client Operations Profiles allow you to create redundancy and fail over capabilities for CM Configuration Servers and CM Proxy Servers, control hardware scans, diagnostic settings, and user interface options.
8	4.1	Page 188, Table 29: The following parameters have been added to radskman: catexp to filter applications, machfreq for thin clients, and mnt to control when Self Maintenance is applied.
8	4.1	Page 191, Table 30: The following parameters have been added to radskman for use with Client Operations Profiles: cop, datauri, product, and rcsuri.
8	4.1	Page 192, Table 31: added the upd parameter, which prevents updates to applications during the Agent Connect session.
8	4.1	Page 198, Table 32: NETAVAIL, RETRYRC, RETRYFLG, RETRYINT, RETRYLMT attributes have been added to the TIMER class to allow for retry if a timer event fails.
8	4.1	Page 204, Table 33: MONTHLY, MONTHDAY, and STARTUP values have been added to the ZSCHDEF attribute in the Scheduler (TIMER) class.
8	4.1	Self-maintenance is now supplied to the customer in the form of export decks. The 4.0 agents use the PRDMAINT domain.
10	4.1	Page 234, Table 39: added a row for SMINFO.
10	4.1	Page 240, Radskman Execution (PREFACE): The following client objects are documented in this guide: ZCONFIG, SAPSTATS, SYNOPSIS, PREFACE, and SMINFO. Radskman Execution (PREFACE) was introduced in version 3.1. SAPSTATS and SYNOPSIS are used with Client Operations Profiles. Systems Management Information (SMINFO) is a new object that includes unique computer information taken from the client computer's BIOS tables.

Chapter	Version	Changes
10	4.1	Page 242, Systems Management Information (SMINFO): new section.

Support

Please visit the HP OpenView support web site at:

www.hp.com/managementsoftware/support

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

HP OpenView 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 log 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

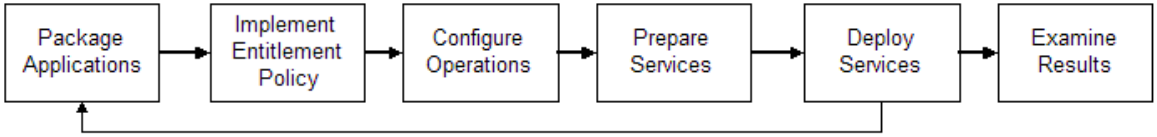
At the end of this chapter, you will:

- Understand the components of Configuration Management (CM).
- Be familiar with the structure of the CM Configuration Server DB (CM-CSDB).
- Understand suggested deployment strategies.

About This Guide

This guide discusses the *suggested* implementation for the HP OpenView Configuration Management Application Manager (CM Application Manager). Although you will modify this strategy to meet your organization's needs, we recommend that you review this guide for a comprehensive understanding of the CM Application Manager. At the start of each chapter, you will find the following diagram to help you locate where you are in the implementation. The appropriate area will be shaded. Before you can manage software, you must install the CM agent and the CM Administrator.

Figure 1 Tasks completed in this guide



The *HP OpenView Configuration Management Application Manager Installation and Configuration Guide* (*CM Application Manager Guide)* covers the following:

- Chapter 2, Installing the CM Agents
  This chapter describes how to install the CM agents.
- Chapter 3, Installing the CM Administrator for UNIX
  This chapter describes how to install the CM Administrator for UNIX.
- Chapter 4, Installing the CM Administrator for Windows
  This chapter describes how to install the CM Administrator for Windows.
- Chapter 5, Packaging Applications and Content
  This chapter describes how to package applications using Component Selection Mode.
- Chapter 6, Implementing Entitlement Policy
  This chapter shows you how to define users and groups, and how to connect them to the appropriate applications.
- Chapter 7, Configuring CM Client Operations Profiles
  This chapter explains how to configure your clients to use the most appropriate CM Configuration Servers and CM Proxy Servers, provide for fail over capabilities, and configure your CM agents.

- Chapter 8, Preparing Services
  This chapter describes services options such as restarting the agent computer.
- Chapter 9, Deploying Services
  This chapter explains how to deploy applications to your agent computers.
- Chapter 10, CM Agent Objects and Directories
  This chapter shows you where to find and how to examine the results of your CM implementation.

About CM Technology

CM technology provides high levels of adaptability, flexibility, and automation. Adaptability comes from the embedded intelligence of platform-independent object-oriented technology. Flexibility is provided by the media-independence of CM technology that enables content to be easily revised and customized. And our solutions automate digital asset management across virtually any kind of network. The following bullets detail each of these distinctive capabilities that are essential to CM technology:

- **The Embedded Intelligence of Object-Oriented Technology**
  Object-oriented technology transforms software and content from file-based media into self-aware, platform-independent, intelligent objects that automatically assess the environment into which they are deployed, and personalize, install, update, and repair themselves accordingly. In other words, as intelligent objects, they know what they need for a particular device or user, where to get what they need, when they need to change, how to change themselves, and how to repair themselves.
- **Revisable Packaging for Revisable Content**
  CM technology enables revision and customization of software and content at any midstream point in the publisher-to-subscriber deployment process. Because CM technology transforms software and content into objects, these objects can be easily modified midstream – subtracted from, added to, reconfigured – simply by packaging them with other objects or new configuration information. With revisable packaging, value-added service providers and IT administrators can customize standard published software offerings for the needs of their particular users without having to unpack and repackage everything.

- **Self-Managing Infrastructure**
  The object-oriented intelligence of CM technology incorporates a self-managing infrastructure. This capability begins with network-independence, with CM technology flexibly supporting any deployment environment, whether client/server, local, wide or virtual area network, intranet, extranet, or the Internet. Furthermore, we support whatever distribution media make sense for the target audience and the provider (which might be a software publisher, application service provider (ASP), Internet service provider (ISP), provider of enterprise application integration (EAI) services, e-business integrator, e-commerce component provider, or in-house IT administrator).

In the Internet age in which software is fundamental to the ability of businesses to compete, change is a constant state, and audience diversity has grown beyond the capacity of older technologies to manage. CM technology provides the necessary automation, adaptability, and flexibility to solve the software management challenge.

Desired State

CM manages the distribution of data based on your **desired state**. A desired state records the identities and intended configurations of the desktop computers whose configurations are managed by CM. The desired state can be simple or complex. At a minimum, a CM desired state includes the following five elements:

- **Users**
  The identity of the computers being managed.

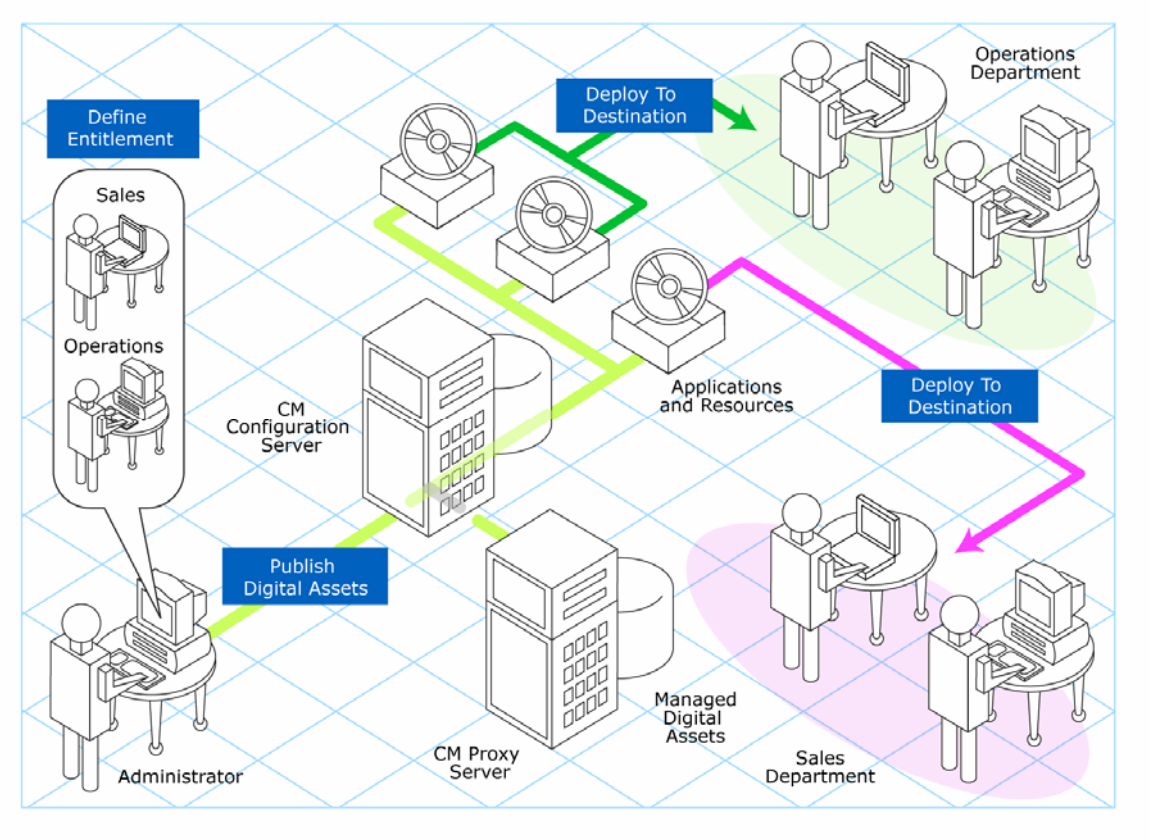
  

  The term **computer** is used to refer to a workstation or server.

- **Applications**
  The data that are being managed.
- **Application Files**
  The components that make up the data.
- **Deployment Source**
  The location where the application components are centrally stored, such as on a CM Staging Server or CM Configuration Server, so they can be deployed to the users.

- **Deployment Destinations**
  The location to which the application and its files will be distributed, such as desktop computers, PDAs, and laptops.

Figure 2 Elements in a desired state



Use CM to manage all of these components. You will publish **packages** of data, determine entitlement policy, and define how the packages will be deployed.

A package is a unit of distributable software or data.

The CM Configuration Server DB

The CM Configuration Server DB, located on the CM Configuration Server, stores the information needed to create the 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 policies determining which subscribers are assigned to which packages.
- Security and access rules for CM administrators.

Use the CM Administrator Configuration Server Database Editor (CM Admin CSDB Editor) to view and manipulate the CM Configuration Server DB. The CM Configuration Server DB is hierarchically structured, and its components consist of files, domains, classes, instances, and attributes.

Elements of the Database

file	A logical partition in the CM Configuration Server DB that groups similar domains. It is the highest level in the hierarchical structure of the database. Example: The PRIMARY File is used to define and maintain the desired state. This is one of the pre-configured files distributed with CM.
domain	A logical partition in the CM Configuration Server DB that groups similar classes. It is the second level in the hierarchical structure of the database. Example: The POLICY Domain contains the classes needed to create users and groups.
class	A logical partition in the CM Configuration Server DB that groups similar instances. It is the third level in the hierarchical structure of the database. Example: The USER Class of the POLICY Domain defines subscribers of CM-managed applications. It defines all of the attributes necessary to identify the agent computer to be managed by CM.

class instance	A specific occurrence of a class. Each instance of a particular Class inherits the attributes defined for that Class. Also called instance. Example: A USER instance is an object created from the USER Class, containing the information needed to identify a subscriber's agent computer.
class attribute	A property of a class that defines its type. There are four types of classes: expression, variable, connection, and method. Example: The NAME attribute of a User Class contains the name of the subscriber, and the USERID attribute contains the User ID, as specified by the CM administrator.

Files and Domains

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

- The LICENSE File is read-only and used for CM Configuration Server processing. This file should only be used by HP, and should not be modified.
- The PRIMARY File is where you will find most information regarding software management. Within the PRIMARY File, there are seven default domains.
  - Use the ADMIN Domain to define administrative rights and rules for connecting classes.
  - Use the AUDIT Domain to configure tasks that will audit agent computers' data. Refer to the *HP OpenView Configuration Management Inventory Manager Guide (CM Inventory Manager Guide)* for more information.
  - Use the CLIENT Domain to configure CM Client Operations Profiles. This includes defining which CM Configuration Servers, CM Proxy Servers, and CM Staging Servers the agent computer can use. For more information, see Chapter 7, Configuring CM Client Operations Profiles.
  - Use the PRDMAINT Domain to store packages for self-maintenance that are supplied by HP. This domain should be used only for the deployment of CM agent maintenance packages. See CM Self Maintenance on page 225 for more information.

- Use the CM PATCH Domain to store information for binary patching of files associated with Service Optimization. Refer to the *HP OpenView Configuration Management Administrator Configuration Server Database Editor Guide (CM Admin CSDE Editor Guide).*

  The CM Patch Manager uses a different domain called PATCHMGR for managing security patches.

- Use the POLICY Domain to create users and groups, and to assign users to groups. See Chapter 6, Implementing Entitlement Policy for more information.
- The SOFTWARE Domain contains information about the software being managed and the methods used to deploy the software. See Chapter 8, Preparing Services and Chapter 9, Deploying Services.
- The SYSTEM Domain contains administrative and process control definitions.

- As you begin to use CM, the PROFILE File appears. This file contains information collected from agent computers. The file appears after the first agent computer has registered with the CM Configuration Server. This information is used to connect to computers to deploy software managed by CM, and to see the configuration of the agent computer. See Chapter 10, CM Agent Objects and Directories for more information about the PROFILE File.
- The NOTIFY File contains information about attempts by the Notify function to update, remove or e-mail subscribers. This file appears after the first attempted Notify. For more information about Notify, see Chapter 9, Deploying Services.

CM Infrastructure

Use infrastructure components to take full advantage of the ability to manage your enterprise's computing environment. Depending on your configuration, your infrastructure may be enhanced by any combination of these components. The CM components can be divided into four categories.

- CM Management Applications
- CM Management Infrastructure
- CM Extended Infrastructure
- CM Management Extensions

Some of the basic CM Infrastructure components are described below. For more information on all of the CM products, refer to the *HP OpenView Configuration Management Getting Started Guide (CM Getting Started Guide)* or the HP OpenView web site.

HP OpenView Configuration Management Configuration Server

The HP OpenView Configuration Management Configuration Server (CM Configuration Server) is part of the CM Management Infrastructure, and resides on a single server or across a network of servers. Applications and information about the subscribers and agent computers are stored in the CM Configuration Server DB. The CM Configuration Server distributes packages based on policies established by the CM administrator. Refer to the *HP OpenView Configuration Management Configuration Server User Guide (CM Configuration Server Guide)* for more information.

HP OpenView Configuration Management Portal

The HP OpenView Configuration Management Portal (CM Portal) is a web-based interface that you can use to manage your infrastructure. The CM Portal is part of the extended infrastructure. Whether you are already using CM, or are just beginning, you can use the portal to create a graphical representation of your infrastructure. Refer to the *HP OpenView Configuration Management Portal Installation and Configuration Guide (CM Portal Guide)* for more information.

HP OpenView Configuration Management Proxy Server

If you want to reduce the load on the CM Configuration Server, or store your data closer to your agent computers, consider using the HP OpenView Configuration Management Proxy Server (CM Proxy Server). The CM Proxy Server stores a copy of the data that are available to subscribers connected to the CM Proxy Server. The CM Proxy Server is also part of the CM extended infrastructure. Evaluate the potential benefits for each server and its attached subscribers individually. For more information, refer to the *HP OpenView Configuration Management Proxy Server Installation and Configuration Guide (CM Proxy Server Guide).*

Contact your HP representative for details on the CM Portal and the CM Proxy Server.

HP OpenView Configuration Management Administrator

CM comes with a set of tools used to carry out software management functions. You should become very familiar with these tools. This is part of the CM management infrastructure. These include:

- **CM Admin Packager**
  Use the CM Admin Packager to create groups of components, called **packages**, and promote them to the CM Configuration Server. See Chapter 5, Packaging Applications and Content for more information.
- **CM Admin CSDB Editor**
  Use the CM Admin CSDB Editor to view and to manipulate the CM Configuration Server DB. In addition to this publication, refer to the *CM Admin CSDB Editor Guide* for more information.
- **CM Admin Agent Explorer**
  Use the CM Admin Agent Explorer to view and to manipulate CM objects on the agent computer.
- **CM Admin Screen Painter**
  Use the CM Admin Screen Painter to create custom dialog boxes.
- **CM Admin Publisher**
  Use the CM Admin Publisher to publish Windows Installer files. Refer to the *HP OpenView Configuration Management Administrator Publisher Guide (CM Admin Publisher Guide)* for more information.

CM Management Applications

Management applications (clients) allow you to automate deployment, update, repair, and deletion activities, and inspect hardware and software. Install the CM management applications onto the subscriber's computer.

There are three types of management applications available for communicating with the CM Configuration Server. Install only those agents for which you have a license. The agent software is located in the Management Applications folder.

- **HP OpenView Configuration Management Application Manager** (CM Application Manager)
  Use this agent to distribute mandatory applications throughout the enterprise. This agent is described in this book.

- **HP OpenView Configuration Management Application Self-service Manager** (CM Application Self-service Manager)
  Subscribers install, remove, and update optional applications that are available in a service list. For more information, refer to the *HP OpenView Configuration Management Application Self-service Manager Installation and Configuration Guide (CM Application Self-service Manager Guide).*
- **HP OpenView Configuration Management Inventory Manager (CM Inventory Manager)**
  Use this agent to collect hardware information and send it to the CM Inventory Manager for collection and reporting. See the *HP OpenView Configuration Management Inventory Manager Installation and Configuration Guide (CM Inventory Manager Guide)* for details.
- **HP OpenView Configuration Management Patch Manager** (CM Patch Manager)
  The CM Patch Manager analyzes and manages security patches. See the *HP OpenView Configuration Management Patch Manager Installation and Configuration Guide (CM Patch Manager Guide).*
- **HP OpenView Configuration Management OS Manager** (CM OS Manager)
  The CM OS Manager controls the provisioning of operating systems. See the *HP OpenView Configuration Management OS Manager User Guide (CM OS Manager Guide)*.
- **HP OpenView Configuration Management Solutions for Servers** (CM Solutions for Servers)
  CM Solutions for Servers has the ability to control server applications and analyze settings and baselines. See the HP Openview Configuration Management Solutions for Servers guides.

If you install both the CM Application Manager and CM Application Self-service Manager feature sets, you decide if an application is mandatory or optional, and specify who controls the installation of the application. By adding the CM Inventory Manager, you can also discover the hardware and software configurations of the agent computer.

Summary

- Configuration Management gives you the flexibility and control to manage desktop software efficiently.
- The CM Configuration Server DB includes all the information needed to manage your software.
- We provide suggested deployment strategies that you should tailor to your organization's needs.

2 Installing the CM Agents

At the end of this chapter, you will:

- Understand the system requirements and permissions necessary to deploy the CM agents.
- Be able to install the CM agents using either the graphical or non-graphical mode.

Install only the CM agents for which you have licenses. If you do not have a license, the CM agent will not authenticate with the CM Configuration Server.

System Requirements

- TCP/IP connection to a computer running CM Configuration Server.
- CM agent requires 20 MB free disk space.

Platform Support

For detailed information about supported platforms, see the release note document that accompanies this release.

Prerequisites

- We strongly recommend installing the CM agents as root. Root authority is required to apply owner and group designators to managed resources.
- Install the CM agent on a local file system.
- The installation program must be run from within UNIX. Although you can continue to work within UNIX (performing other tasks and operations) while the installation program is being executed, we strongly recommend that you do not.
- If you intend to run any of the graphical components of the CM agent software, make sure the UNIX environment variable DISPLAY is set in your environment. If it is not, you will need to set this variable to indicate the hostname or IP address to which you would like to redirect the graphical display.

Table 3 [PROPERTIES] Section of INSTALL.INI

In a.....	Type....
C shell	setenv DISPLAY IP address or hostname:0.0
Bourne, Bash, or Korn shell	DISPLAY=IP address or hostname:0.0 export DISPLAY

If there is an existing installation in the current working directory, we urge you to relocate it before beginning installation. You will be prompted for this during the installation. If you choose to overwrite your existing agent, all your customized data will be lost.

When installing the CM agent, you must know the subscribers' operating systems. After setup and configuration, CM executables and library files will not be changing with the same frequency as that of your site's user files.

To successfully run CM applications, standard UNIX environment variables are required. Minimally, these environment variables should include the fully qualified path of the installed client executables, the path to the operating system-specific Motif libraries, and the standard UNIX operating system paths for operating system executables and shared libraries. We recommend these be included as part of the logon scripts of the UNIX user ID who installs, and will maintain the CM agents.

Table 4 Environment Variables

Platforms	Examples
Solaris	LD_LIBRARY_PATH=/lib:$IDMSYS:$MOTIF:$LD_LIBRARY_PATH PATH=/bin:/usr/bin:$IDMSYS:$MOTIF:$PATH
HP-UX	SHLIB_PATH=/lib:$IDMSYS:$MOTIF:$SHLIB_PATH PATH= /bin:/usr/bin:$IDMSYS:$MOTIF:$PATH
AIX	LIBPATH=/lib:$IDMSYS:$MOTIF:$LIBPATH PATH=/bin:/usr/bin:$IDMSYS:$MOTIF:$PATH
Linux	LD_LIBRARY_PATH=/lib:/usr/lib:$IDMSYS:$LD_LIBRARY_PATH PATH=/bin:/usr/bin:$IDMSYS:$PATH

In Table 4 above, `$IDMSYS` represents the fully-qualified path to the CM agent executables, often referred to as the `IDMSYS` location. MOTIF represents the fully-qualified path to the Motif libraries installed with the operating system.

The inclusion of the MOTIF libraries is required only when running CM agent or CM Administrator graphical tools such as the CM Admin Publisher, the CM Admin Agent Explorer, and the presentation of the CM agent logon panel.

After the CM agent is installed, the file `.nvdrc` is placed in the `HOME` directory of the UNIX user ID who performed the installation. This file aids you in setting the required environment variables needed to use the CM agents. We recommend adding a line to the appropriate logon scripts to invoke this shell script:

```
. $HOME/.nvdrc
```

Recommendations

- After you perform an installation, make sure the CM Application Manager is successfully connected to the CM Configuration Server. This registers the subscriber in the CM Configuration Server DB. Once registered, the subscriber appears in the PROFILE File. Make sure to verify that all ports are active and that you have full connectivity to the CM Configuration Server.

Before you install the CM agent, consider the following:

- You can perform a local installation of the CM agents.
- Your CM systems administrator can perform a Remote Installation Setup. This process stores the installation media in a selected directory path. Later agent installations can be initiated from any number of intended agent workstations providing they have access to the directory path selected during the Remote Installation Setup.
- Performing an installation from a customized configuration file provides a number of benefits.
  - Replication of precise installation details on multiple clients.
  - Ability to use a pre-installation method that runs any script or executable before the CM agent installation.
  - Ability to use a post-installation method, which runs any script or executable after the CM agent is installed.
  - You can configure the installation to force a client connection to the CM Configuration Server immediately after the installation.
  - You can pre-configure the IP address and port number of the CM Configuration Server that the CM agent will be connecting to.
  - Ability to use an object update text file that can be used to update CM objects after the installation.

Installation Methods

You can install the CM agents by:

- Executing the installation procedure directly from the CM media.

- Copying the files from the CD media into a temporary directory and executing the installation procedure.

Several parameters can be used on the command line when installing the CM agents. These parameters are used to install the CM agent using the graphical mode, non-graphical mode, plain mode, or silent mode. Table 5 below, describes the installation parameters.

Table 5 Command Line Installation Parameters

Parameter	Example	Description
`-mode plain`	`./install -mode plain`	Installs the CM agent in plain mode. The installation graphics are displayed with no animations. This is useful for remote installations where network bandwidth may be an issue.
`-mode text`	`./install -mode text`	Installs the CM agent in text mode using the non-graphical installation. The installation takes place completely on the command line. The installation will default to text mode if the DISPLAY environment variable is not set.

Including Maintenance Files with the Agent Installation

If additional maintenance files are available, for example, service packs or hot fixes, you can include these files with your agent installation by creating a maintenance tar file.

Within your agent installation media `/ram` directory, create a file called `maint.tar` that includes all updated files.

The agent installation will check for `maint.tar` and if found, the client installation will extract all updated files into the `IDMSYS` directory.

Installing the CM agent

This section describes both the graphical (using a GUI) and non-graphical (using a command line) installations of the CM agent for UNIX.

In order for CM to install correctly on HP-UX platforms, you must mount the media using pfs_mount.

The CM media is created using the Rock Ridge format. Since the HP-UX standard mount procedure is incompatible with the Rock Ridge file system type, HP has made available the PFS package (Portable File System) that allows their workstations to recognize this format. Specific instructions follow:

Insert the CM media and mount by typing:

```
/usr/sbin/pfs_mount -v -x unix /dvdrom/mnt
```

where */dvdrom* is your physical media device.

To un-mount, type:

```
/usr/sbin/pfs_umount /mnt
```

See your local UNIX systems administrator and UNIX man pages for more information.

Graphical Installation

This section describes how to install the CM agents both to a local and to a remote computer using a graphical user interface (GUI).

Local Installation

This section describes how to install the CM agents to a local computer using a GUI.

To install the CM agent to a local computer using a GUI

These instructions will guide you through the local graphical installation of the CM agent. For the non-graphical installation instructions, see Non-graphical Installation on page 43.

1. Depending on your version of UNIX, change your current working directory to the correct Agents platform subdirectory on the installation media.

   Example: For HP-UX, type: **`cd /dvdrom/Agents/hpux`**

2. Type **`./install`**, and then press **Enter**.
3. The Welcome window opens.

At any point during the installation, you can return to a previous window by clicking **Back**. Also, if you would like to exit the installation at any time, click **Cancel**.

4 Click **Next**.

The End User License Agreement window opens.

5 Read the agreement and click **Accept** to continue.

The Select Components to Install window opens.

6 Select the **CM Application Manager** check box.

7 Click **Next**.

The CM Daemons window opens.

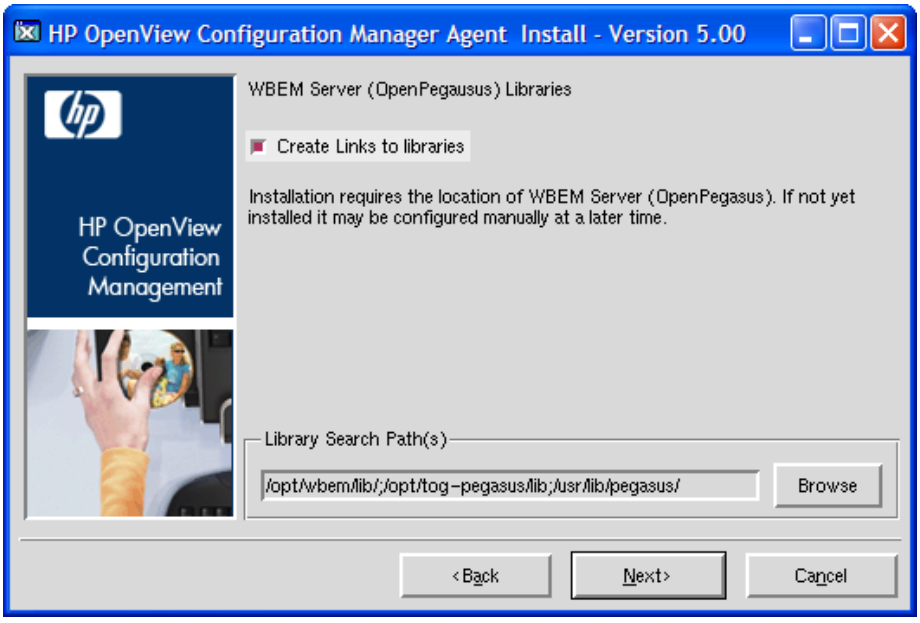
8 Select when you want the CM Daemons to start. The CM Daemons run on the client computer and perform CM management tasks. See About CM Daemons in UNIX on page 45 for additional information.

— Select **Start after installation** to start the daemons after the Agent installation is complete.

— Select **Automatic start after reboot via init scripts** to configure the daemons to start automatically each time the device is restarted.

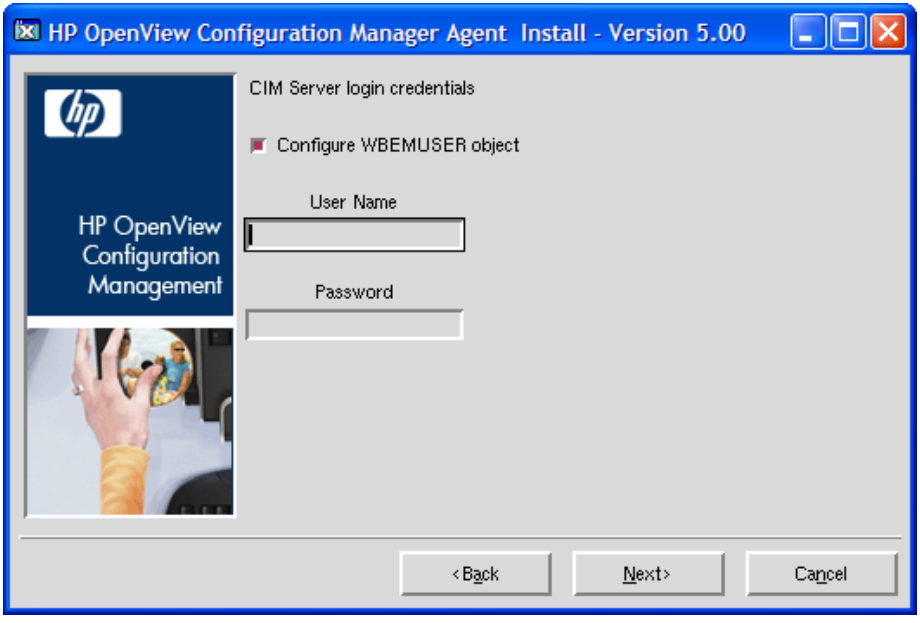
9 Click **Next**.

The WBEM Server (OpenPegasus) Libraries window opens. If you are running the installation on a Solaris device, you will be prompted for CIM server login credentials, see below.



Select **Create Links to libraries** to create a link to existing WBEM Server libraries. Enter the location in the text box. Links can be created after the CM Agent is installed.

If you are running the installation on a Solaris device, the CIM Server login credentials window opens.



10 Select to configure the WBEMUSER object by adding a user name and password (Solaris only).

11 Click **Next**.

The Select Installation Type window opens.

Select **Local Install** to install the CM agent onto a local computer, and then click **Next**.

The CM Agent Location window opens.

12 Type the name of the directory where you want to install the CM agent, or click **Browse** to navigate to it.

13 Click **Next**.

If the specified directory already exists you will be prompted to verify this location.

— If you would like to update the existing directory, click **OK**.

— If you want to specify a different location, click **Cancel**.

The Lib Directory window opens.

14 Type the name of the directory where you would like to store proprietary information created by CM (the `lib` directory), or click **Browse** to navigate to it.

15 Click **Next**.

The Log Directory window opens.

16 Type the name of the directory where you would like to store the log files generated by CM, or click **Browse** to navigate to it.

17 Click **Next**.

The CM Configuration Server IP Address window opens.

18 Type the IP address (format: xxx.xxx.xxx.xxx) of the CM Configuration Server to which the CM agent will connect. Specify a valid IP address or hostname recognized by the agent workstation.

19 Click **Next**.

The CM Configuration Server Port Number window opens.

20 Type the CM Configuration Server's port number (default is 3464).

21 Click **Next**.

The Package Settings window opens.

22 Review the settings displayed in the Package Settings window. If you would like to change any of the settings, click **Back** until you get to the appropriate window.

23 When you are satisfied with the settings, click **Install** to install the CM agent with these settings.

24 When the installation is complete, click **Finish** to exit the program.

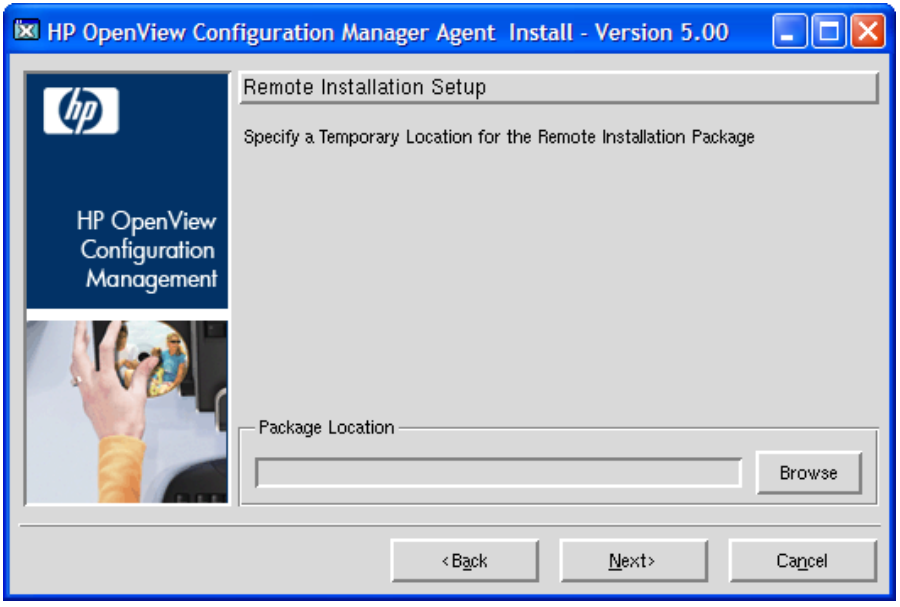
The CM agent has been successfully installed.

Remote Installation Setup

This section describes how to create a CM Agent installation configuration file that can be used to install the CM Agent in silent mode or to a remote computer.

After the Remote Installation Setup is finished, a configuration file is saved in a directory you specify. Use the –cfg installation option to use the configuration file you created.

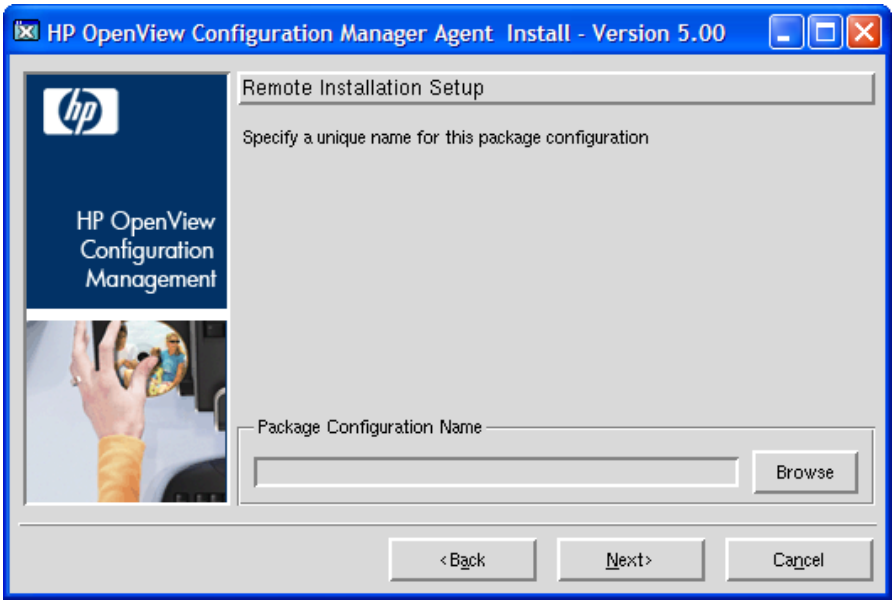
The remote installation is identical to the local install with the exception of two additional steps required for creating the remote installation package. Follow the steps for a local install, above, and when prompted, enter the required information for creating the remote installation package.



— Type the fully qualified path to a directory where you would like to store the CM agent installation media for future client installations, or click **Browse** to navigate to it.

— Click **Next**.

The Package Configuration Name window opens.



— Type the fully qualified path to a configuration file that you would like to use for silent installations, or click **Browse** to navigate to it. The configuration file you specify will contain the installation information you chose during the Remote Installation Setup.

After a remote installation is complete, the CM agent installation media is stored on disk for future installations.

Once the media has been stored for other computers to use for remote installations, you should become familiar with the variables in the configuration file.

Customizing the Installation Configuration File

A configuration file supplies the default responses for silent CM agent installations. These responses would normally be provided during an interactive CM agent installation. When performing silent installations, additional installation options are also available in the configuration file.

The variables available in the configuration file are described in Table 6, below.

Table 6 Configuration File Variables

Variable	Sample Value	Description
REMOTE	0	0 designates a local installation. 1 designates a Remote Installation Setup.
INSTDIR	`/opt/HP/CM/Agent`	The default installation directory.
IDMLOG	`/opt/HP/CM/Agent/log`	This can be defined to designate a directory for IDMLOG other than the default INSTDIR/log.
IDMLIB	`/opt/HP/CM/Agent/lib`	This can be defined to designate a directory or IDMLIB other than the default INSTDIR/lib.
PREPROC		The fully qualified name of a script or executable to run pre-installation.
PREPARM		Any parameters that may be required by the pre-installation method specified in the variable PREPROC.
POSTPROC		The fully qualified name of a script or executable to be run post-installation.
POSTPARM		Any parameters required by the post-installation method specified in the variable POSTPROC.
MGRIP	192.168.123.40	The default IP address for connection to the CM Configuration Server.
MGRPORT	3464	The default port number for connection to the CM Configuration Server.
NTFYPORT	3465	The default Notify port used.

Variable	Sample Value	Description
CONNECT	Y	Connects to the CM Configuration Server immediately after the installation. Default behavior is N. Set to Y if you want your CM agent to connect to the CM Configuration Server automatically after the installation.
OBJECTS	`./object.txt`	The file that is used to create or update CM attributes after the installation.
DUAL	1	0 designates RAM only selected. 1 designates more than one component selected.

Using a Pre- or Post-Installation Script

You can create and run custom executables or shell scripts prior to or after the silent installation of a CM agent. For example, your post-installation script can initiate a connection to the CM Configuration Server in order to process mandatory applications. The example below is part of a shell script that initiates the connection to the CM Configuration Server and processes mandatory applications.

```
#!/bin/sh
#
cd /opt/HP/CM/Agent

# ZIPADDR is the IP address or hostname of the manager
ZIPADDR="xxx.xxx.xxx.xxx"
# ZDSTSOCK is the TCP port the manager is running on
ZDSTSOCK="3464"

# To manage the machine
# 1. .edmprof must exist in root's home directory
# 2. The connect must be run as root

/opt/HP/CM/Agent/radskman mname=NVDM,dname=SOFTWARE,ip=$ZIPADDR,
port=$ZDSTSOCK,cat=prompt,ind=y,uid=\$MACHINE,startdir=SYSTEM,ulo
gon=n
```

Customizing Installed Object Variable Content

The configuration file option OBJECTS allows you to specify the fully qualified path to a filename that contains data in the form:

```
OBJECT_NAME VARIABLE_NAME VARIABLE_VALUE
```

```
An example of a valid object file is:
```

```
ZMASTER ZTRACE N
```

```
ZMASTER ZTRACEL 000
```

When creating an object text file:

- A pound sign (#) at the beginning of a line indicates a comment.
- A pound sign (#) on any other part of a line will be considered data.
- The format is OBJECT_NAME followed by VARIABLE_NAME. Everything after the VARIABLE_NAME is considered VARIABLE_VALUE.
- The VARIABLE_VALUE text should not be enclosed by any special characters.

Performing a Silent Installation of a CM Agent

We recommend that you install the agent as root.

Performing a silent installation of the CM agent using stored CM agent installation media requires that:

- your CM system administrator has already run the Remote Installation Setup installation method.
- the workstation running the silent installation is able to access the directory path where the installation media was stored.

Several parameters can be used on the command line when performing a silent installation of the CM agent. Table 7 on page 43 describes these.

Table 7 Silent installation command line parameters

Parameter	Example	Description
`-cfg`	`./install -cfg install.cfg`	The file name specified after `-cfg` is the name of the configuration file to be used during the installation. For information about configuration files, see Customizing the Installation Configuration File on page 39.
`-mode silent`	`./install -mode silent -cfg install.cfg`	Installs the CM agent in silent mode based on the parameters set in the configuration file specified after the `-cfg` parameter. For information about configuration files, see Customizing the Installation Configuration File on page 39.

Non-graphical Installation

This section describes a non-graphical (using a command line) installation of the CM agent for UNIX.

To install the CM agent for UNIX using a command line

These instructions guide you through the local non-graphical installation of the CM agent for UNIX. For the graphical installation, see Graphical Installation on page 34.

1. Depending on your version of UNIX, change your current working directory to the correct Agents subdirectory on the installation media.

   Example: For HP-UX, type: **`cd /cdrom/Agents/hpux`**

2. Type **`./install -mode text`**, and then press **Enter**.

   The CM agent installation begins.

3. Type **`C`**, and press **Enter**.

4. Read the license agreement, type **`Accept`** and press **Enter**.

5. In the next few steps, select which Agents to install. Type Y or N and press **Enter** at each prompt.

   - CM Application Manager
   - CM Inventory Manager
   - CM Application Self-Service Manager
   - CM OS Manager
   - CM Patch Manager
   - CM Server Management

6. You are prompted to start the CM Daemons after installation. Press **Enter** to accept the default (Y) and start the CM Daemons after install or type N and press **Enter** to start them later.

7. You are then prompted to automatically start the CM Daemons after a reboot via init scripts. Press **Enter** to accept the default (N) and **not** start the CM Daemons each time the device is restarted or type **`Y`** and press **Enter** to allow CM Daemons to automatically start when the device is rebooted.

8. If you are installing to a Solaris device, you will be prompted to configure the WBEMUSER object. If you select Y you will then be prompted to supply a user name and password for the WBEMUSER object.

9. Select the type of installation. The default is 1, a local installation.

   Type **`1`**, and then press **Enter** to install the CM agent locally.

   or

   Type **`2`**, and then press **Enter** to set up remote installation media.

   For this example, we accepted the default.

10. Specify the installation location for the CM agent, and then press **Enter**.

11 Specify the location for the CM proprietary objects (IDMLIB), and then press **Enter**.

12 Specify the location for the log files created by CM (IDMLOG), and then press **Enter**.

13 Specify the IP address of the CM Configuration Server, and then press **Enter**.

14 Specify the port number for the CM Configuration Server, and then press **Enter**.

15 Review the installation settings you have chosen.

16 If you would like to install the CM agent with these parameters, press **Enter** to accept the default answer of **`Y`**.

   If you want to change any of these settings, type **`N`** to re-enter the installation information.

17 When you are satisfied with the settings, press **Enter** to install the CM agent.

   The CM agent is installed.

About CM Daemons in UNIX

The CM agent installation program installs the following daemon executables:

- **CM Notify (default port 3465)**
  Use CM Notify, **radexecd**, to push updates to subscribers or to remove applications. A Notify message is sent from the CM Configuration Server to this daemon. When the daemon receives the Notify message, the CM Application Manager connects to the CM Configuration Server and performs the action initiated by the Notify operation.

  If you want to send a Notify to subscribers of a particular application, that application *must* be installed on their computers in order for them to be eligible for notification.

- **CM Scheduler**
  Use the CM Scheduler service, **radsched**, to schedule timer-based deployments of applications.

The installation of **radexecd** and **radsched** as services on a UNIX workstation is not automated within the context of the installation. The starting of services on UNIX workstations is operating system dependent. For information about installing CM daemons as system services at boot time, see your local UNIX system administrator or refer to your UNIX operating system's manual.

Sample Shell Scripts

The installation of the CM agent includes a subdirectory called "sample". It contains a sample shell script called **daemons.sh** that may be used to start, stop, and restart the **radexecd** and **radsched** daemons.

- To start the radexecd and radsched daemons, type: `daemons.sh start`
- To stop the radexecd and radsched daemons, type: `daemons.sh stop`
- To stop, then restart the radexec and radsched daemons, type: `daemons.sh restart`

Troubleshooting the Agent Installation

If you encounter any problems while installing the CM Agent, perform the following steps before contacting technical support:

- Enable diagnostic tracing by appending the text **`-loglevel 9`** to the installation command line and re-run the installation.
- Have this log file (`tmp/setup/setup.log`) located in the home directory of the UNIX user ID who ran the install.

The installation option `-loglevel 9` should only be used to diagnose installation problems.

Summary

- We strongly recommend that you install and run the CM agents as root.
- The CM agents can be installed using either the graphical or non-graphical modes.

3 Installing the CM Administrator for UNIX

At the end of this chapter, you will:

- Understand the system requirements and permissions necessary to install the CM Administrator for UNIX.
- Be able to install the CM Administrator using either the graphical or non-graphical mode.

If you are responsible for packaging applications or configuring them for distribution, install the CM Administrator on your administrator computer.

Use the CM Admin Publisher to create software or data packages, and then promote them to the CM Configuration Server DB.

System Requirements

- TCP/IP connection to a computer running CM Configuration Server.

Platform Support

For detailed information about supported platforms, see the release note document that accompanies this release.

Prerequisites

- We strongly recommend installing the CM agents as root.
- Install the CM agent on a local file system.
- If you intend to run any of the graphical components of the CM Administrator software, make sure the UNIX environment variable DISPLAY is set in your environment. If it is not, you will need to set this variable to indicate the hostname or IP address to which you would like to redirect the graphical display.

Table 8 Setting the DISPLAY Variable

In a.....	Type....
C shell	setenv DISPLAY IP address or hostname:0.0
Bourne, Bash, or Korn shell	DISPLAY=IP address or hostname:0.0 export DISPLAY

If the DISPLAY environment variable is not set in your environment, the installation will default to a non-graphical installation.

Troubleshooting

Should you encounter any problems while installing the CM UNIX Agent, please perform the following steps before contacting technical support:

- Enable diagnostic tracing by appending the text **`-loglevel 9`** to the installation command line and re-run the installation.
- Put this log file (`tmp/setup/setup.log`) located in the home directory of the UNIX user ID who ran the install.

The install option `–loglevel 9` should only be used to diagnose installation problems.

Recommendations

- We strongly recommend that you install and run the CM Administrator as root.

Root authority is required to apply owner and group designators to managed resources.

Installation Methods

You can install the CM Administrator by:

- Executing the installation procedure directly from the CD-ROM.
- Copying the files from the CD-ROM into a temporary directory and executing the installation procedure.

Several parameters can be used on the command line when installing the CM Administrator.

Table 9 Command line parameters

Parameter	Example	Description
`-mode plain`	`./install -mode plain`	Installs the CM Administrator in plain mode. The installation graphics are displayed in plain mode (no moving graphics). This is useful for remote installations where network bandwidth may be an issue.
`-mode text`	`./install -mode text`	Installs the CM Administrator in text mode using the non-graphical installation. The installation takes place completely on the command line. The installation will default to text mode if the DISPLAY environment variable is not set.

Installing the CM Administrator for UNIX

This section describes both the graphical (using a GUI) and non-graphical (using a command line) installations of the CM Administrator for UNIX.

Graphical Installation

This section describes how to install the CM Administrator for UNIX using a graphical user interface (GUI).

To install the CM Administrator for UNIX using a GUI

These instructions will guide you through the graphical installation of the CM Administrator. For non-graphical instructions, see Non-graphical Installation on page 54.

If the UNIX user ID of the person performing the CM Administrator installation has previously installed a CM agent, the location of the CM Administrator will default to the location of the CM agent executables.

1 Depending on your version of UNIX, change your current working directory to the correct UNIX subdirectory on the installation media.

Example: For HP-UX:

`/Configuration Server/management_infrastructure/administrator_workstation/hpux/`

2 Type **`./install`**, and then press **Enter**.

The Welcome window opens.

3 Click **Next**.

If you are installing the CM Administrator to a computer with the same UNIX user ID that had previously installed a CM agent, the installation program will prompt you for the CM Configuration Server's IP address next. The next three windows: CM Administrator Location, Lib directory, and Log directory, are only needed if you are installing the CM Administrator to a computer that does not have a CM Agent already installed.

The HP Software License terms window opens.

4 Read the software license terms and click **Accept**.

The CM Administrator Location window opens.

5 Type the name of the directory where you are installing the CM Administrator, or click **Browse** to navigate to it.

6 Click **Next**.

If the specified directory already exists you will be prompted to verify this location.

— If you would like to update the existing directory, click **OK**.

— If you would like to change the directory location, click **Cancel**.

The Lib Directory window opens.

7 Type the name of the directory where you would like to store proprietary information created by CM (the `lib` directory), or click **Browse** to navigate to it.

8 Click **Next**.

The Log Directory window opens.

9 Type the name of the directory where you would like to store the log files generated by CM, or click **Browse** to navigate to it.

10 Click **Next**.

The CM Configuration Server IP Address window opens

11 Type the IP address (format: xxx.xxx.xxx.xxx) or host name of the CM Configuration Server you will be publishing to.

12 Click **Next**.

The CM Configuration Server Port Number window opens.

13 Type the port number of your CM Configuration Server (default is 3464).

14 Click **Next**.

The Package Settings window opens.

15 Review the settings displayed in the Package Settings window. If you would like to change any of the settings, click **Back** until you get to the appropriate window.

16 When you are satisfied with the Package Settings, click **Install**.

The CM Administrator is installed.

Non-graphical Installation

This section describes a non-graphical (using a command line) installation of the CM Administrator for UNIX.

To install the CM Administrator for UNIX using a command line

These instructions guide you through the non-graphical installation of the CM Administrator. For the graphical installation, see Graphical Installation on page 52.

Task 1 Depending on your version of UNIX, change your current working directory to the correct subdirectory on the installation media.

Example: For HP-UX:

```
/Configuration Server/management_infrastructure
/administrator_workstation/hpux/
```

1 Type `./install -mode text`, and then press **Enter**.

The CM Administrator installation begins.

2 Type `C`, and then press **Enter**.

If you are installing the CM Administrator to a computer with the same UNIX user ID that had previously installed a CM agent, the installation program will prompt you for the CM Configuration Server's IP address next. The next three sections: CM Administrator Location, `Lib` directory, and `Log` directory, are only needed if you are installing the CM Administrator to a computer that does not have a CM agent already installed.

3. Specify the installation location for the CM Administrator, and then press **Enter**.
4. Specify the location for the CM proprietary objects (IDMLIB), and then press **Enter**.
5. Specify the location for the log files created by CM (IDMLOG), and then press **Enter**.
6. Specify the IP address of the CM Configuration Server, and then press Specify the port number of the CM Configuration Server, and then Review the installation settings you have chosen.
7. If you would like to install the CM Administrator with these settings, press **Enter** to accept the default (Y) and begin the installation or type **N**, to re-enter your installation information.
8. To complete the configured installation process, press **Enter**.

   The CM Administrator is installed.

Summary

- We strongly recommend that you install and run the CM Administrator as root.
- Install the CM Administrator for UNIX using either the graphical or non-graphical mode.

4 Installing the CM Administrator for Windows

At the end of this chapter, you will:

- Understand the system requirements for installing the HP OpenView Configuration Management Administrator (CM Administrator).
- Be familiar with the installation files.
- Know how to install the CM Administrator using the Installation Wizard and command lines.
- Understand the feature settings of the CM Administrator.
- Know how to remove and repair the CM Administrator using the Installation Wizard and command lines.

The CM Administrator installation program uses Microsoft Windows Installer. The program consists of one MSI package with six feature sets—CM Admin Packager, CM Admin Configuration Server DB Editor, CM Admin Agent Explorer, CM Admin Publisher, CM Admin Screen Painter, and CM AMP Editor.

System Requirements

- Clean computer. (A **clean computer** is a computer with only the target subscriber's operating system installed.)
- Windows 2000, Server 2003, XP or Vista. (x86 or x64 where applicable).
- TCP/IP connection to the CM Configuration Server.
- Minimum resolution of 800 x 600.
- MS Windows Installer Version 2.0 or higher.
  The MSI 2.0 installation program is available in the `managementinfrastructure\administratorworkstation\win32\msi` folder on the CM Infrastructure media. If Windows Installer does not exist, or if an earlier version is detected on the computer, the MSI 2.0 installation program runs automatically.
- For Windows NT, 2000, Server 2003, or XP or Vista, you must have administrator rights to the computer to install the CM Administrator.

About the Installation Files

setup.exe

`setup.exe` is stored on the CM Infrastructure media in the `managementinfrastructure\administratorworkstation\win32\` folder. It accepts any command line parameters and passes them to Windows Installer.

You can also create a Windows Installer Administrative Installation Point (AIP) for network installations.

A Windows Installer Administrative Installation Point (AIP) is also known as an Administrative Control Point (ACP).

The AIP starts Windows Installer and passes any command line parameters to it. To create the Windows Installer AIP in a specified target directory, type:

```
SETUP.EXE /a TARGETDIR=drive:\targetdirectory /qb
```

The target directory contains `CM-ADMIN50.MSI`, the installation folders, and setup.exe.

CM-ADMIN50.MSI

`CM-ADMIN50.MSI` is the MSI database file, which contains the default configuration information for the installation. This file is stored on the CM Infrastructure Media in the `managementinfrastructure \administratorworkstation\win32\` folder.

Installing the CM Administrator

This section describes how to install the CM Administrator using the Installation Wizard and using a command line.

Using the Installation Wizard to Install the CM Administrator

This section describes how to install the CM Administrator for Windows using the Installation Wizard.

To install the CM Administrator using the Installation Wizard

1. From the folder containing the CM Administrator installation files, run `setup.exe`.

   The CM Administrator Installation Wizard opens.

2. Click **Next**.

   The License Agreement window opens.

3. After reading and accepting the license agreement, click **Next**.

   If the CM agent is not installed on the computer, the Destination Folder window opens.

If the CM agent is already installed on the computer, this window will not open and the CM Administrator is installed in the same location as the CM agent.

If you want to select a different destination for the CM Administrator, click **Browse**, and then navigate to the appropriate destination folder.

Click **OK** to continue.

4 Click **Next**.

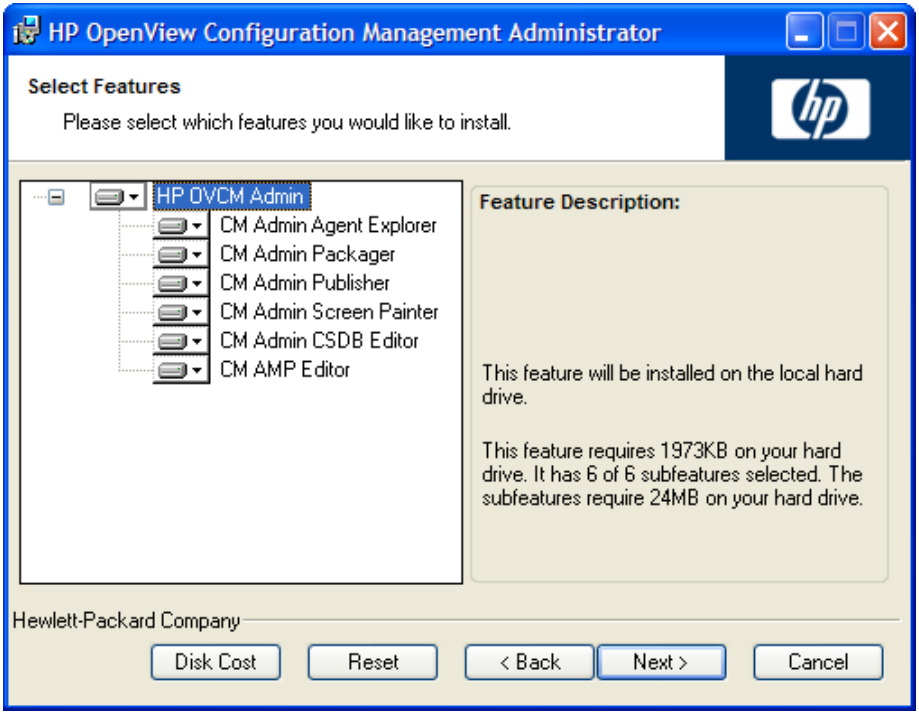
The CM Configuration Server window opens.

5 In the IP Address text box, type the IP address for the CM Configuration Server.

6 In the Port text box, type the port number (default is 3464).

7 Click **Next**.

The Select Features window opens.



8 Click  to select the features that you would like to install.

If you want to set the same options for all of the features, you can click  next to CM Administrator and select the appropriate option to apply the setting to all features.

Click **Disk Cost** to see an overview of the disk space needed for the installation.

Each time you click  a shortcut menu for that feature opens.

9 From the shortcut menu, select an installation option. These options are described in Table 10 below.

Table 10 Feature settings for the CM Administrator

Option	Description
Will be installed on local hard drive	Installs the top-level feature on the local hard drive, but not any sub-features listed below.
Entire feature will be installed on local hard drive	Installs the entire feature, including any sub-features listed below. Note: In this installation program, selecting this option or the "Will be installed on local hard drive" option for any of the features results in the same installation because these features do not contain sub-features.
Entire feature will be unavailable	The feature will not be installed. If previously installed, this feature will be removed.

10 Click **Next**.

The Ready to Install the Application window opens.

11 Click **Install** to begin the installation.

When the installation is done, the CM Administrator has been successfully installed window opens.

12 Click **Finish** to exit the installation.

Using a Command Line to Install the CM Administrator

You can also use a command line to run the CM Administrator installation program, For example, if you want to install only the CM Admin Packager on a computer, the command line that you run from the directory containing the CM Administrator installation files might be:

```
SETUP.EXE ADDLOCAL=NVDINSTALLPACKAGER
```

Specifying the Features to Install

To specify the features that you want to install, use the appropriate feature state argument, such as ADDLOCAL, and specify the features that you want to install.

Table 11 CM Administrator feature state arguments

Specify the following arguments	To set the feature state
ADDLOCAL	Type a comma-delimited list of features that you want set to "Will be installed on local hard drive."
ADDSOURCE	Type a comma-delimited list of features that you want set to "Will be installed to run from network."
ADVERTISE	Type a comma-delimited list of features that you want set to "Feature will be installed when required."
REMOVE	Type a comma-delimited list of features that you want set to "Entire feature will be unavailable." This only removes the features—not the entire product. Therefore, if you use the REMOVE property and type each of the feature names, the core product will still be stored on your computer. If you want to remove the entire product, type REMOVE=ALL.

When specifying features on the command line, reference the CM Administrator features as follows:

- NVDINSTALLPACKAGER CM Admin Packager
- NVDINSTALLPUBLISHER CM Admin Publisher
- NVDINSTALLSYSTEMEXPLORER CM Admin CSDB Editor
- NVDINSTALLCLIENTEXPLORER CM Admin Agent Explorer
- NVDINSTALLSCREENPAINTER CM Admin Screen Painter
- NVDINSTALLAMPEDITOR CM AMP Editor

For example, if you want to install the CM Admin Configuration Server DB Editor and the CM Admin Agent Explorer to the computer, type the following command line:

```
SETUP.EXE ADDLOCAL=
NVDINSTALLSYSTEMEXPLORER,NVDINSTALLCLIENTEXPLORER
```

If you run the installation from a command line, be sure to pass the IP address for the CM Configuration Server to the installation. For example:

```
SETUP.EXE NVDOBJZMASTER_ZIPADDR=10.10.10.1
```

Additional Command Line Arguments

Table 12 below describes some additional arguments that you can pass to the installation program on the command line.

Table 12 Command Line Arguments

Argument	Description
`/qn`	Performs a silent installation.
`/qb`	Displays the progress bar only during the installation.
`/l*v` *drive*`:\install.log`	Creates a detailed Windows Installer log. Note: Using this option may impact the performance of the installation.
`/a TARGETDIR=` *drive*`:\targetdirectory`	Creates a Windows Installer AIP in the specified target directory. Note: A Windows Installer Administrative Installation Point (AIP) is also known as an Administrative Control Point (ACP). The target directory contains `CM-ADMIN50.MSI`, the installation folders, and setup.exe. Once you have created the AIP, you can run setup.exe and pass the appropriate command line parameters. This starts the Windows Installer and passes the specified parameters to it.

Removing the CM Administrator

The Windows Installer installation program offers the ability to remove the CM Administrator. This section describes how to remove the CM Administrator using the Installation Wizard and using a command line.

Using the Installation Wizard to Remove the CM Administrator

This section describes how to remove (uninstall) the CM Administrator using the Installation Wizard.

To remove specific features of the CM Administrator, use the Modify option on the Application Maintenance window. See Modifying the CM Administrator Installation on page 66.

To remove the CM Administrator using the Installation Wizard

1. From the folder containing the CM Administrator installation files, double-click **setup.exe**.

   The Application Maintenance window opens.

2. Select the **Remove** option.

3. Click **Next**.

   The CM Administrator Uninstall window opens.

4. Click **Next**.

   The files for the CM Administrator are removed from the computer.

   The CM Administrator has been successfully uninstalled window opens.

5. Click **Finish**.

Using a Command Line to Remove the CM Administrator

This section describes how to remove (uninstall) the CM Administrator using a command line.

To remove the CM Administrator using a command line

- From the folder containing the CM Administrator installation files, type the following command line:

```
SETUP.EXE REMOVE=ALL
```

or

If you would like to remove a single CM Administrator feature, type a comma-delimited list of the features that you want to remove on the command line.

Example

If you want to silently remove the CM Admin CSDB Editor and CM Admin Agent Explorer, type:

```
SETUP.EXE REMOVE=NVDINSTALLSYSTEMEXPLORER,
NVDINSTALLCLIENTEXPLORER /qn
```

Reference the features for the CM Administrators as follows:

- CM Admin Packager NVDINSTALLPACKAGER
- CM Admin Publisher NVDINSTALLPUBLISHER
- CM Admin CSDB Editor NVDINSTALLSYSTEMEXPLORER
- CM Admin Agent Explorer NVDINSTALLCLIENTEXPLORER
- CM Admin Screen Painter NVDINSTALLSCREENPAINTER
- CM AMP Editor NVDINSTALLAMPEDITOR

This only removes the features—not the entire product. Therefore, if you use the REMOVE property and type each of the feature names, the core product will still be stored on your computer.

Repairing the CM Administrator

The Windows Installer installation program offers the ability to repair the CM Administrator. For example, if you have a missing CM Administrator module, you can use this tool to repair the installation. This tool will not overwrite modules that exist on the computer if they are newer than the ones provided with the installation.

This section describes how to repair the CM Administrator using the Installation Wizard and using a command line.

Using the Installation Wizard to Repair the CM Administrator

This section describes how to repair the CM Administrator using the Installation Wizard.

To repair the CM Administrator using the Installation Wizard

1. From the folder containing the CM Administrator installation files, double-click **setup.exe**.

   The Application Maintenance window opens.

2. Select the **Repair** option.

3. Click **Next**.

   The Ready to Repair the Application window opens.

4. Click **Next**.

   When the repair is done, the CM Administrator has been successfully installed window opens.

5. Click **Finish**.

Using a Command Line to Repair the CM Administrator

This section describes how to repair the CM Administrator using a command line.

To repair the CM Administrator using a command line

- From the folder containing the CM Administrator installation files, type the following command line:

  ```
  msiexec /f Cm-Admin50.Msi
  ```

  Additional parameters can be used with this command line. For more information, see your Windows Installer documentation.

Modifying the CM Administrator Installation

The Windows Installer installation program offers the ability to modify the CM Administrator installation by adding or removing individual features.

This section describes how to modify the installation of the CM Administrator using the Installation Wizard and using a command line.

Using the Installation Wizard to Modify the CM Administrator

This section describes how to modify the installation of the CM Administrator using the Installation Wizard.

To modify the CM Administrator installation using the Installation Wizard

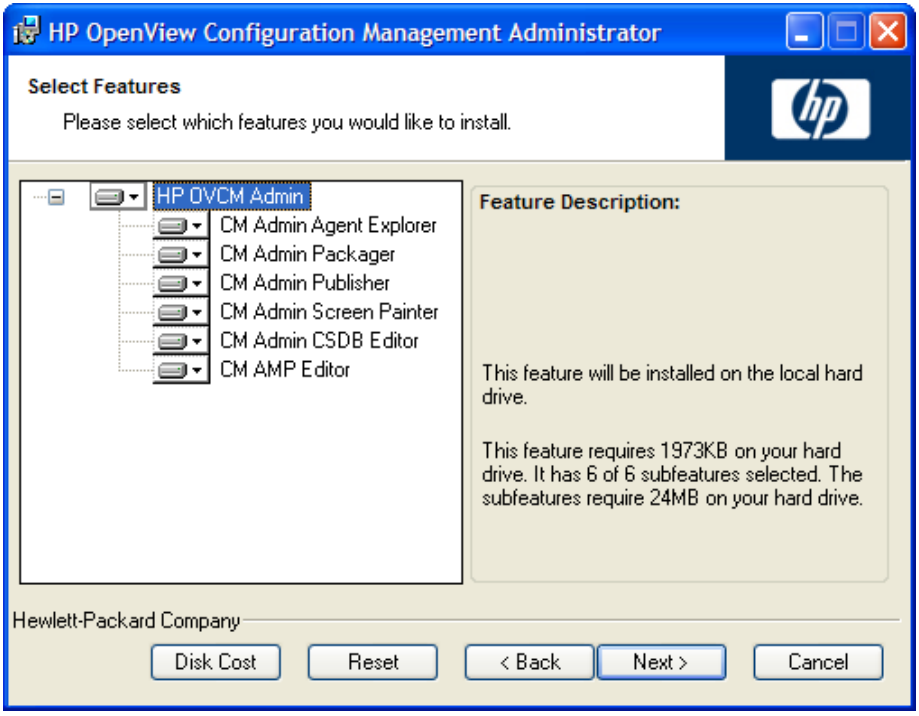
1 From the folder containing the CM Administrator installation files, double-click **setup.exe**.

The Application Maintenance window opens.

2 Select the **Modify** option.

3 Click **Next**.

The Select Features window opens. See Installing the CM Administrator on page 59 for information about how to use this window.



4 Click **Next**.

The Ready to Modify the Application window opens.

5 Click **Next**.

The CM Administrator has been successfully installed window opens.

6 Click **Finish** to close the installation program.

Using a Command Line to Modify the CM Administrator Installation

To modify the CM Administrator installation using a command line

- From the folder containing the CM Administrator installation files, type the following command line:

```
SETUP.EXE FeatureStateArgument=feature1,feature2
```

See Table 11 on page 62 for more information.

Example

To install the CM Admin Packager to the local hard drive, and to make the CM Admin CSDB Editor and CM Admin Explorer unavailable, type the following command line:

```
SETUP.EXE ADDLOCAL=NVDINSTALLPACKAGER
REMOVE=NVDINSTALLSYSTEMEXPLORER,NVDINSTALLCLIENTEXPLORER
```

See Additional Command Line Arguments on page 63 for additional arguments.

Summary

- The CM Administrator consists of one package with six feature sets: CM Admin Publisher, CM Admin Packager, CM Admin CSDB Editor, CM Admin Agent Explorer, CM Admin Screen Painter, and CM AMP Editor.
- Install the CM Administrator on a clean computer.
- You can install the CM Administrator using a command line or using the Installation Wizard.

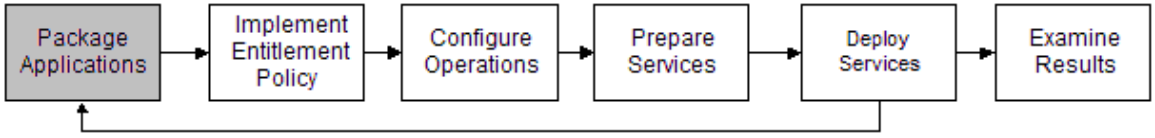
5 Packaging Applications and Content

At the end of this chapter, you will:

- Understand the packaging process.
- Understand the requirements for publishing software or content.
- Be able to publish an application using Component Selection Mode
- Be aware of the CM Batch Publisher.
- Use the New Application Wizard in the CM Admin CSDB Editor to create a service.
- Be able to prepare and distribute maintenance packages to the CM Application Manager.

This guide discusses the *suggested* implementation for the CM Application Manager. Although you will modify this strategy to meet your organization's needs, we recommend that you review this guide for a comprehensive understanding of the CM Application Manager. This chapter focuses on packaging.

Figure 3 Tasks completed in this guide



About Packaging and Publishing

Packaging is the process of identifying resources, editing those resource's installation attributes, defining how they are to be installed, and saving the resources and installation instructions in a machine-readable file format or package. A package typically contains one or more files and configuration settings

Publishing is the process of importing a package and its imbedded information to the CM Configuration Server DB. A package must be published before its content can be distributed and deployed into your environment.

For the UNIX version of the CM Admin Publisher, there is one publishing mode available, **Component Selection Mode**. In Component Selection Mode, you select the individual components that make up the application, such as files, directories, and links.

After you create a package, you **promote** it to the CM Configuration Server DB. The package is copied to the CM Configuration Server DB and several instances are created, as described below.

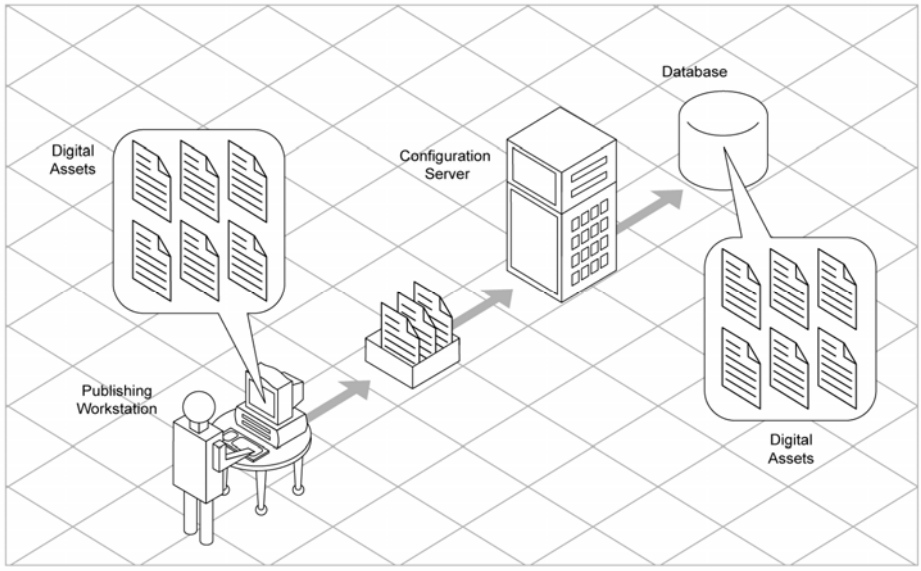
- An **Application Packages (PACKAGE)** instance that represents the promoted package.
- One **UNIX File Resources (UNIXFILE)** instance for each file in the package.
- One **Path (PATH)** instance for each unique path to one or more components on the computer where the software is installed.

Above are some of the default classes available in the SOFTWARE Domain. You can also add your own classes to the CM Configuration Server DB.

Then, you will use the CM Admin CSDB Editor to create a service, assign policies, and prepare the package for deployment. See Chapter 6, Implementing Entitlement Policy and Chapter 9, Deploying Services for more information.

The CM Admin CSDB Editor is currently available for Windows platforms. For more information, refer to the *CM Admin CSDB Editor Guide*.

Figure 4 Packaging digital assets



Packaging Considerations Checklist

Before packaging your data, there are several items that you need to consider.

General

- What do you know about the data to be packaged?
- What is the name of the package going to be? Follow your naming conventions.
- Do you have a unique session ID? Follow your conventions.

System Configuration

- What operating systems are your target computers (workstations or servers) using?
- How much RAM is necessary to handle the data?
- What is the minimum processor necessary to handle the data?

Activation Options

- When do you want to activate the application - immediately on distribution or later?
- Which version of the application do you want to distribute, and when do you want to activate it?

Data Options

- What type of compression do you want to use?
- Will your data be stored in the CM Configuration Server DB or on a proxy server?
- Are you distributing maintenance to the CM Application Self-service Manager agent?
- How do you want to promote the resources? Will you use the force lock method?
- Are you sending out an update and only want to deploy the changes?

Verify Options

- Do you want to use the standard, default verification options?
- Is this a first time installation? Is there anything that you need to verify?
- When deploying files, what types of statistics do you want to check – date, time, size?
- Do you want to update all files, or only newer files?
- If a file already exists, do you want to deploy it again to overwrite any changes that may have been made?
- Do you want to use the internal version to determine whether a file should be deployed?

Delivery Options

- Do your files or methods need to be deployed in a particular order?
- Is the data mandatory or optional? Note: You can only deliver mandatory files if the CM Application Manager is installed.
- Do you want the data deployed under the user or machine context?

Agent Behaviors

- After the file is deployed, do you want to run any methods? If so, what are they?
- Does anything need to happen to enable the file once it's deployed? If so, what method will you run to enable it?
- If the subscriber is no longer subscribed to the software, do you want to delete the file?
- Do you want to compare the old and new version of the file that you are deploying? If so, what method do you want to use?

Using Component Selection Mode

In Component Selection Mode, you select the individual components that make up the application, such as files, directories, and links to create a package.

Publishing in Component Selection Mode involves four phases:

1. **Select** the individual files to be published
2. **Edit** the file properties and methods
3. **Configure** the package and service options
4. **Publish** the files to the CM Configuration Server DB

Prerequisites

Before publishing your application in Component Selection Mode:

- Install the target application on your packaging machine. This ensures that the files you need to select reside on the computer.

Publishing

This section guides you through publishing a sample application using Component Selection Mode and provides detailed information about each screen that you encounter.

Use this example to become familiar with Component Selection Mode. However, please remember that there are many variables when publishing applications.

Task 1 Logging On to CM Admin Publisher

1. Log on as root.
2. Launch the CM Admin Publisher by running `./publishr` from the location where you installed the CM Administrator.
3. Type your User ID and Password in the appropriate fields.

The User ID, as shipped, is RAD_MAST. No password is necessary. This may have been changed in your installation. Check with your CM security administrator to obtain your own User ID and Password, if necessary.

The User ID and password are case-sensitive.

4 In the Type of data to publish drop-down box, leave the default choice of **Component Select**. (This is the only Publishing mode currently available)

5 Click **OK**.

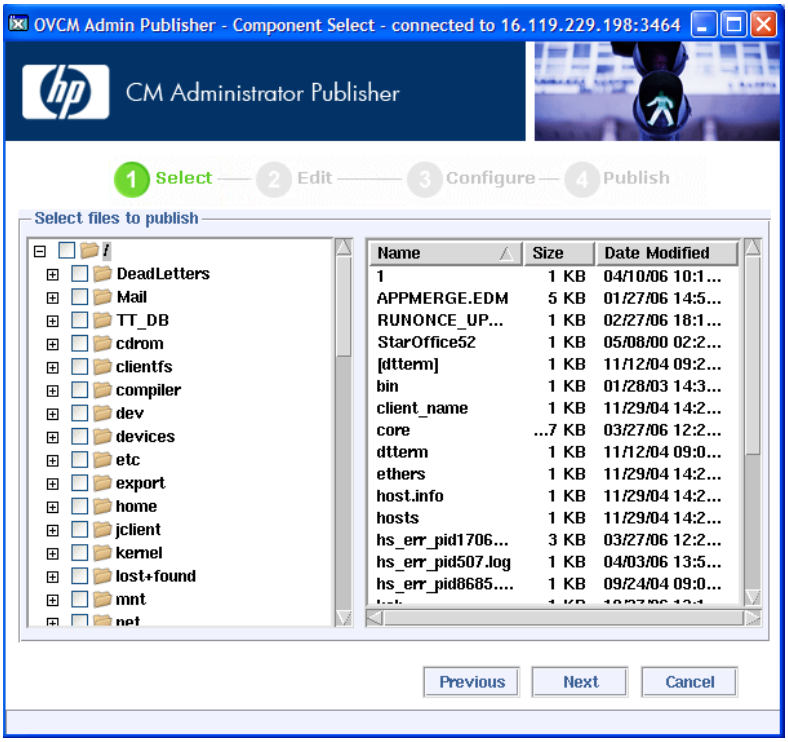
The Select files to Publish window opens.

Task 2 Selecting Files to Publish

Use the Select files to publish window to select all files that need to be included in the package.

To select the files to publish

- Navigate through your file system (shown in the figure below) and select the files or directories to be included in the package. Click a check box again to clear a selection.



The file selection window displays the files available in order by:

- An alphabetized listing of directories.
- Then, an alphabetized listing of files.
- An alphabetized listing of UNIX links.
- Re-size the file selection window by positioning your mouse over the vertical bar separating the two windows, clicking and dragging to the left or right.

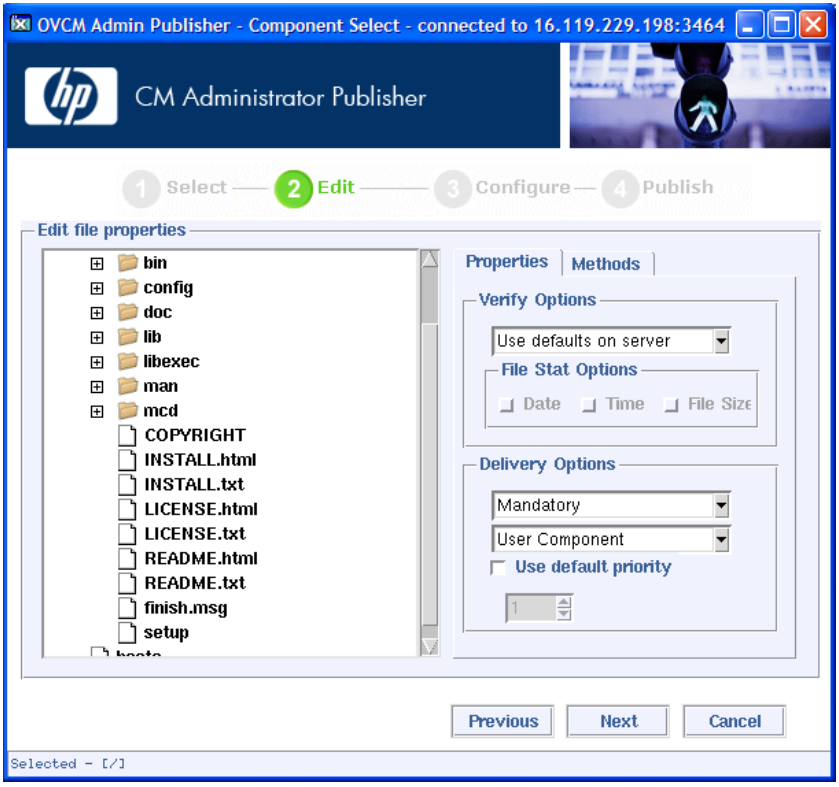
  Click **Next** to go to the Edit file properties window.

Task 3 Editing Properties and Methods

Use the tree-view on the left to see all files and folders that will be included in the package. Use the tabs in the right-hand pane to adjust file properties and methods.

Properties

File properties consist of verification and delivery options.



Verify Options

- **Use defaults on server** (default)
  Select this option so that verification options for these files or directories are inherited from the base instance of the UNIXFILE Class.

  Use the CM Admin CSDB Editor to look at the ZRSCVRFY attribute of the base instance of the UNIXFILE Class to determine what verification options apply, by default. For example, in the next image, ZRSCVRFY=Y.

  The following figure and bullet points refer to the CM Admin CSDB Editor. The CM Admin CSDB Editor is currently available for 32-bit Windows platforms. For more information, refer to the *CM Admin CSDB Editor Guide*.

- **File statistics**
  Select this option so that the CM agent checks the selected statistics (Date, Time, or File Size) for the files or directories on the computer. Select the Date, Time, or File Size using the check boxes below the Verify options drop-down list.

The files or directories are deployed from the CM Configuration Server DB or CM Staging Server if the statistics of the files or directories on the computer are different from the statistics for these files or directories. You can also use CM Admin CSDB Editor to set this option: ZRSCVRFY=D, ZRSCVRFY=S, ZRSCVRFY=T, or ZRSCVRFY=Y.

- **Content check using CRC**
  Select this option to perform content CRC checking for the resource. This populates the ZRSCCRC attribute of the resource's UNIXFILE Class. ZRSCVRFY is set to Y.

  Content CRC checking is a time consuming process and should be used sparingly.

- **Update if newer**
  Select this option so that these files or directories are deployed if the files or directories in the CM Configuration Server DB (or CM Staging Server) have a later date/time stamp than those on the subscriber's computer. You can also use the CM Admin CSDB Editor to set this option: ZRSCVRFY=U.

- **Existence only**
  Select this option so that these files or directories are deployed if they are not on the subscriber's computer. No action is taken if the files or directories already exist on the subscriber's computer, even if the statistics differ from those in the CM Configuration Server DB. You can also use the CM Admin CSDB Editor to set this option: ZRSCVRFY=E.

- **None**
  Select this option so that the files are deployed the first time the application is deployed. No subsequent action is taken. You can also use the CM Admin CSDB Editor to set this option: ZRSCVRFY=N.

Delivery Options

The following options apply only if there is not enough space on the subscriber's computer to install the entire application.

Select if the application will be Mandatory or Optional.

- **Mandatory**
  Select this option to indicate that these files or directories are critical to the application. If there is not enough space on the subscriber's computer for the entire application, CM will deploy only mandatory files. If there is not enough space for the mandatory files, then the application is not deployed at all. You can also use the CM Admin CSDB Editor to set this option: ZRSCMO=M.

- **Optional** (default)
  Select this option to indicate that files or directories are not critical to the application. If there is not enough space on the subscriber's computer for the entire application, CM will not deploy optional files. You can also use the CM Admin CSDB Editor to set this option: ZRSCMO=O.

The following options apply only to operating systems supporting multiple users with a required sign on.

- **User Component**
  Select User if you want to indicate that the file will be deployed only to the subscriber logged on when the application is initially deployed. You can also use the CM Admin CSDB Editor to set this option: ZCONTEXT=U.
- **Machine Component**
  Select Machine to indicate that the file will be deployed to all users of the computer. You can also use the CM Admin CSDB Editor to set this option: ZCONTEXT=M.

Select **Use default priority** (this is selected by default) to use the default priority of 50. Priority determines the order of deployment, from highest priority to lowest priority. You can also use the CM Admin CSDB Editor to set this option: ZRSCPRI=50.

If you de-select the check box, you can enter a number from 1 to 99 to override the default priority of 50. 1 is the highest priority and 99 is the lowest. You can also use the CM Admin CSDB Editor to set this option: ZRSCPRI=1.

Methods

The command lines that you type in the following dialog boxes are stored in variables in the UNIXFILE Class instances in the SOFTWARE Domain.

- **Resource Initialization Method** (Variable in database: ZINIT)
  Type the method to run when the files or directories are stored on the subscriber's computer.
- **Method to Install Resource** (Variable in database: ZCREATE)
  Type the method to run after the file is stored on the computer. This is used if some processing is required to enable the file to be used on the computer.
- **Method to Uninstall Resource** (Variable in database: ZDELETE)
  Normally, files are removed if the subscription to the software is cancelled. If a file, such as a shared object file, should not be deleted from

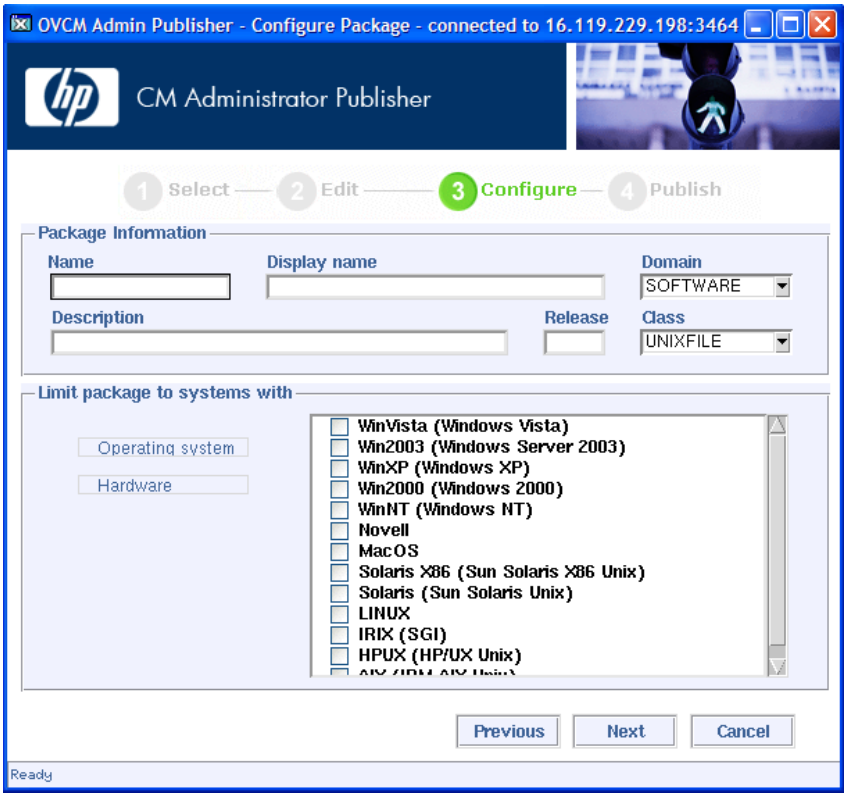
the subscriber's computer, even if the subscription to the software is cancelled, type **_NONE_** (with the underscores) as the value for Method to De-install Resource.

- **Instance Update Method** (Variable in database: ZUPDATE)
  Type the method to run when the instance is modified on the computer, after the file has been deployed.
- **File Update/Add Method** (Variable in database: ZFILEUPD)
  Type the method to run when the file is new or has been updated. The method executes just before the file is deployed to the computer.
- **File Arbitration Method** (Variable in database: ARBITRAT)
  Type the method to run if files or directories are about to be replaced. This method examines the version information of the files or directories that exist and the files or directories that are going to replace it, and then determines which to keep.

Click **Next** to view the Package Properties window.

Task 4 Entering Package Properties

Use the Package Properties window to name the package and include additional descriptive information as well as set package deployment limitations based on hardware and operating system settings.



Package Information

- **Name**
  Type a name for the package. This is the name for the PACKAGE Class instance in the CM Configuration Server DB and should conform to your naming conventions. Note that the name cannot contain any spaces.

  You may want to establish a naming convention to ensure that identifiers are unique. CM Admin Publisher uses this identifier to construct data objects and filenames. See Appendix A, Naming Conventions for more information.

- **Display Name**

  Enter a display name for the package. This is the friendly name used in the CM Admin CSDB Editor.

- **Domain**
  Select the domain in which to store the instance. This is normally the SOFTWARE Domain unless you customized the CM Configuration

Server DB with proprietary domains. As shipped, the default domains are ADMIN, AUDIT, PATCH, POLICY, SOFTWARE, and SYSTEM.

See Chapter 1, Introduction for more information about the domains.

- **Description**
  Type a description for the package.
- **Release**
  Type the release number of the software.

Package Deployment Limitations

Use the Limit package to systems with section to limit the distribution of the package to computers that meet specific requirements. If none of the options are selected, the package will be available to all eligible subscribers.

- **Operating system**
  Select the operating systems for which this package applies..
- **Hardware**

  To limit distribution based on minimum RAM or processor speed, select the check box before the appropriate configuration option and enter the minimum requirement in the text box provided.

Click **Next** to go to the Service Information window.

Task 5 Entering Service Information

Select whether you want to create a new service (Create new), use an existing service (Use existing), or skip creating a service (No service) at this time.

If you want to create a package only, select **No service**. This is useful if, for example, you have a single service, but want to create multiple packages and later connect them to the existing service using the CM Admin CSDB Editor.

Enter the appropriate information in the service description text boxes. If you are using an existing service, make sure you enter the service name correctly.

In the Assignment type section, select whether the service is mandatory or optional. By default, **Mandatory** is selected, which will distribute this service to all available subscribers. Optional services are only available if you are using the CM Application Self-Service Manager agent.

In the Management type section, indicate how the application will be managed after it is deployed.

Management Type and Assignment Type correspond to the ZSVCMO service attribute value. This field can contain two values, depending on what you select for each type. The table below displays the possible attribute values.

Table 13 ZSVCMO Variables

Assignment Type	Management Type	Resulting ZSVCMO Value	Explanation
Mandatory	Automatic	M	Service is deployed to all subscribers and managed by CM.
Mandatory	Manual	MO	Service is deployed to all subscribers. It is not managed by CM after deployment.
Optional	Automatic	OM	Service is deployed to only those subscribers that accept it. Service is managed by CM.
Optional	Manual	O	Service is deployed to only those subscribers that accept it. It is not managed by CM after deployment.

In the Report on the following events section, select each check box next to the events you would like to record. If you would like to use the default application reporting events available in the Base Instance, select **Use Base**.

Click **Next** to open the Summary window.

Task 6 Viewing Summary information and Promoting the Package

Use the Summary window to view the package and service information before publishing.

If you need to change or modify your selections, click **Previous** until you reach the appropriate window. When you are satisfied with the package, click **Next** until you arrive back at this window.

You can also use the buttons in the toolbar to return to a previous screen.

To publish the package

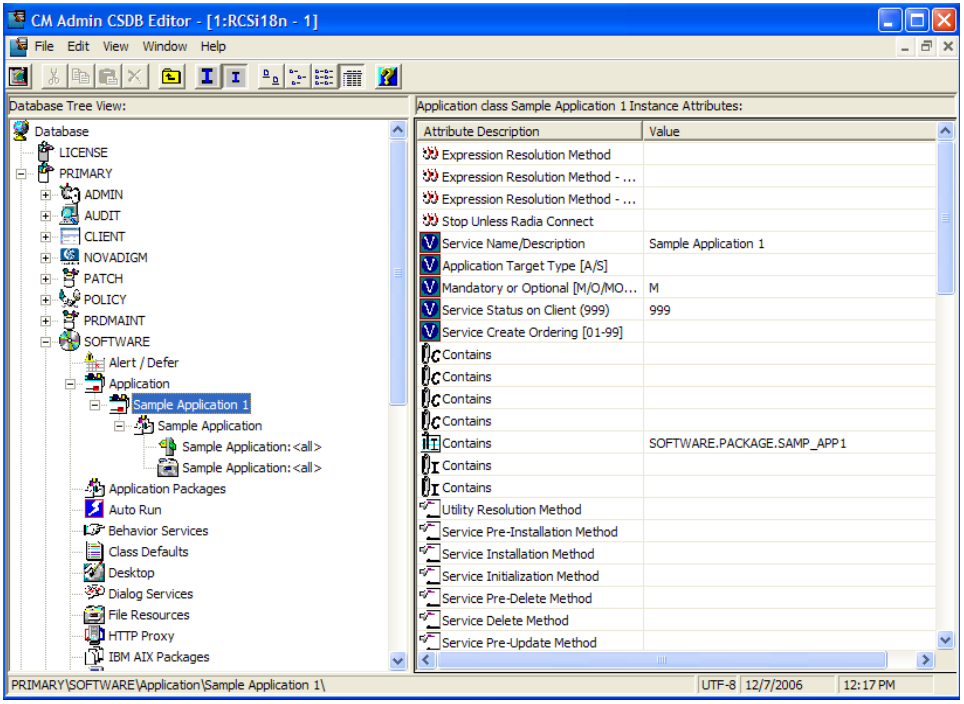
1 Click **Publish**. The package is promoted to the database.

2 Click **Finish** when you receive the message that the process completed successfully.

3 Click **Yes** to confirm that you want to exit.

View the published service in the CM Admin CSDB Editor

Figure 5 Sample Application viewed in database



CM Batch Publisher

The CM Batch Publisher is a command-line alternative to using Component Selection Mode, which offers an automated, repeatable command-line process to create CM packages and store them in the CM Configuration Server DB for distribution.

The CM Batch Publisher can:

- Search for files on multiple drives/file systems.
- Search for, and publish files, from any mapped file/drive system.

- Be configured to limit the subdirectories that are searched.
- Include or exclude at the file level.
- Select files by type.

The CM Batch Publisher can also accommodate frequent patching of internal applications. Its capacity to revise content material is reliable, and can be designed to perform continuously, at designated times, and in predetermined intervals. The CM Batch Publisher can be easily executed from within any script or code capable of calling a command prompt.

CM Native Packaging

CM Native Packaging, is a feature of the CM Batch Publisher specifically designed for UNIX environments. CM Native Packaging is a command-line driven content-publishing tool supporting native HP-UX and Solaris software. CM Native Packaging is installed during the regular installation of the CM Batch Publisher on a UNIX system.

CM Native Packaging explores UNIX native software depots, searches for available native packages and publishes wrapped native packages to the CM Configuration Server. CM Native Packaging will publish all necessary information that will allow you immediate installation of native software to end agents. When the CM Application Manager agent is installed, a Tcl script is included in the `IDMSYS` directory that is required when packages published using CM Native Packaging are deployed. For more information, refer to the *CM Batch Publisher Guide*.

The CM Batch Publisher is an optional feature available from HP. Contact your sales representative for more details.

Creating a Service

If you did not create a service when you published your application, you can use CM Admin CSDB Editor to create the service from the package you promoted.

Use the New Application Wizard in the CM Admin CSDB Editor to create a service. Use the CM Admin CSDB Editor to see services listed in the Application (ZSERVICE) Class.

Using the New Application Wizard to Create a Service

The following instructions use the CM Admin CSDB Editor. The CM Admin CSDB Editor is currently available for 32-bit Windows platforms. For more information, refer to the *CM Admin CSDB Editor Guide*.

Task 1 Accessing the CM Admin CSDB Editor

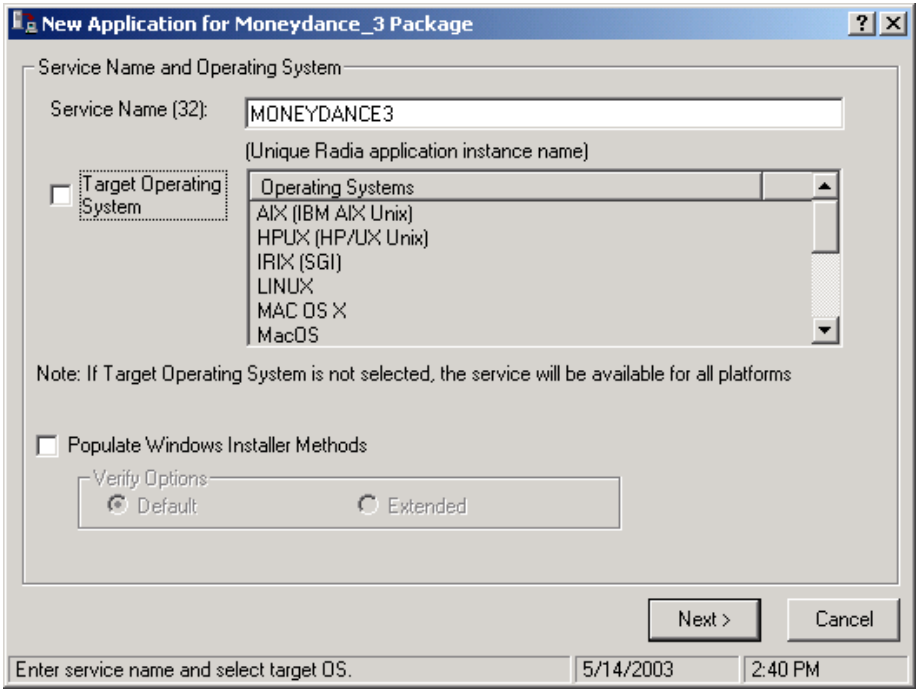
1. Go to **Start** → **Programs** → **HP OVCM Admin** → **CM Admin CSDB Editor**.
2. In the CM Admin CSDB Editor Security Information dialog box, type your User ID and Password in the appropriate fields.
3. Click **OK**.

Task 2 Navigating to the PACKAGE Class of the SOFTWARE Domain

1. Double-click **PRIMARY**. The domains of the PRIMARY File appear beneath its icon in the tree view and in the list view.
4. Double-click **SOFTWARE**. The classes of the SOFTWARE Domain appear beneath its icon in the tree view and in the list view.
5. Double-click **Application Packages (PACKAGE)** to open the PACKAGE Class. The instances of the PACKAGE Class appear beneath its icon in the tree view and in the list view.

Task 3 Using the New Application Wizard to Create a Service

1. In the PACKAGE class of the SOFTWARE domain, right-click an application instance and select **New Application Wizard** .from the shortcut menu.

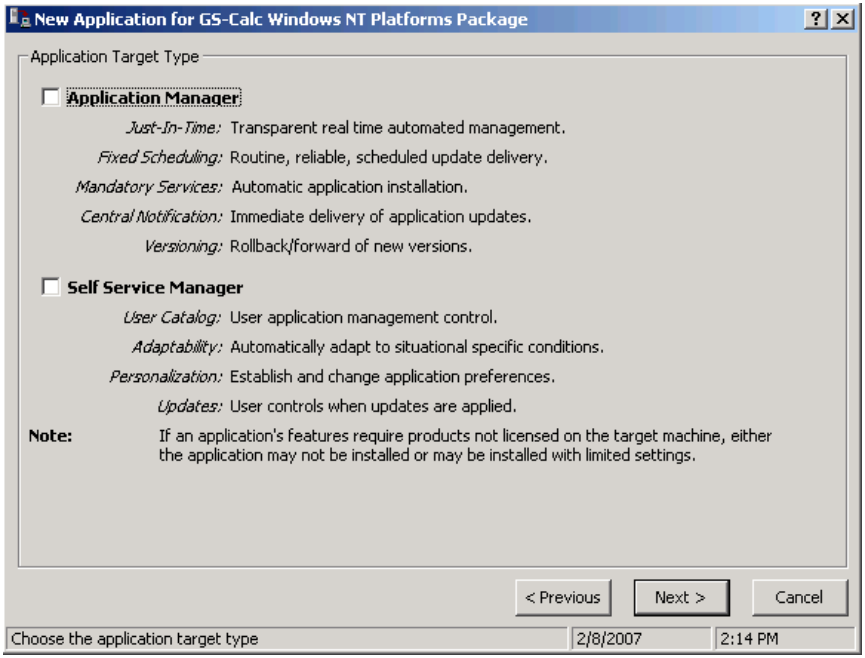


6 In the Service Name (32) text box, type a name for the Application (ZSERVICE) instance.

7 Select the **Target Operating System** check box only if your intended target operating system appears in the list, and the specific operating system for which the package applies is selected.

8 If you are creating a service for a Windows Installer-enabled application, you must select the **Populate Windows Installer Methods** check box. Do not select this check box for this exercise. This option is not applicable to UNIX-specific packages.

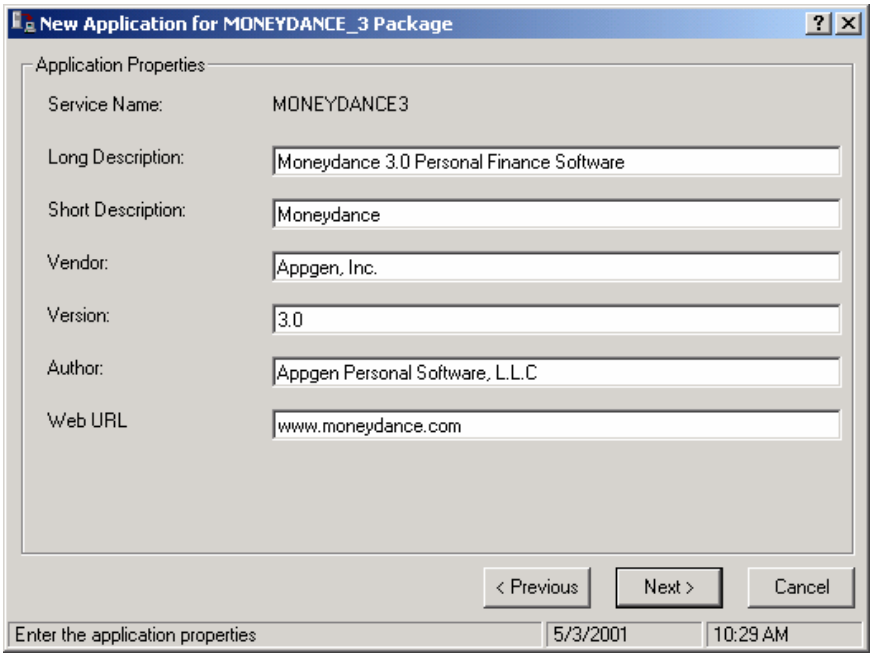
9 Click **Next** to select the application target type.



10 Select the **Application Manager** check box. This designates the service as a mandatory application for your subscribers.

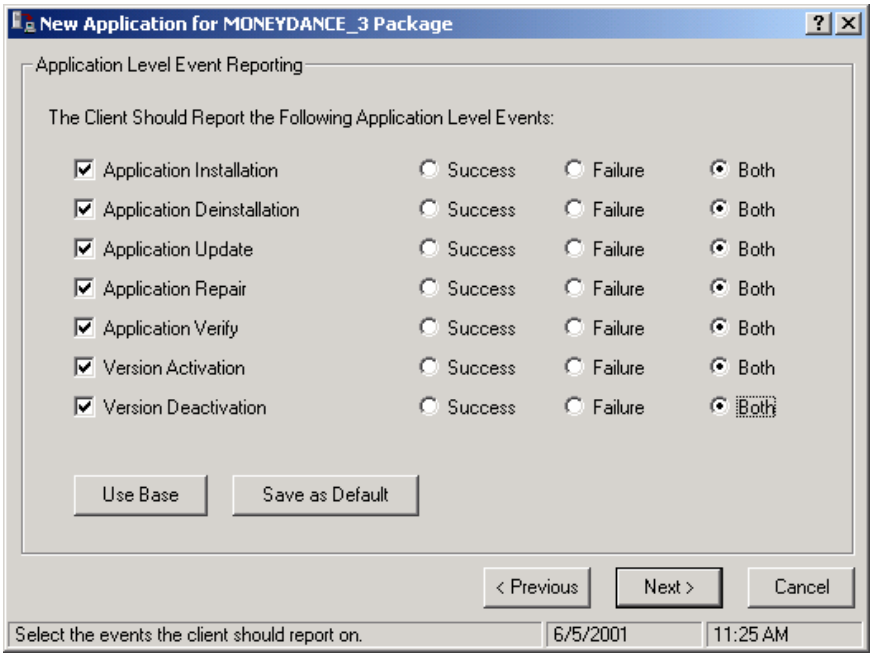
When using the CM Application Manager, applications *must* be mandatory in order to deploy them to your subscribers.

11 Click **Next** to enter the application properties.



12 Type the appropriate information into each Application Properties field.

13 Click **Next** to select the events that the CM Application Manager will report on.



14 Click the check box for each event that you want to report on. Then, select the appropriate option button to indicate whether to report on the event's success, failure, or both.

or

Click **Use Base** if you want to inherit the values for the ERTYPE and EVENTS variables from the base instance of the Application (ZSERVICE) instance. These variables control event reporting.

For this example, we selected every Application Event to be reported in the event of a success or failure.

For more information about these variables and the APPEVENT object, see Chapter 10, CM Agent Objects and Directories.

If you want to save the current settings as the default settings for the Application Event Panel, click **Save as Default**.

15 Click **Next** to review your selections.

16 Click **Finish** to create the application instance.

17 Click **OK** when you are prompted with a message indicating that the application has been added. The instance appears in the ZSERVICE Class.

If you want to modify any of the information that you entered in the New Application Wizard, locate the corresponding variable and change its value.

Now, you are ready to set up policies identifying *which* subscriber receives *what* software. See Chapter 6, Implementing Entitlement Policy for more information.

Service Groups

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

This includes products where:

- A product may utilize more than one service-package.
- A large product may need to be split into smaller sub-services to install only specific parts of the product suite.

For detailed information on creating CM Service Groups, refer to the *CM Admin CSDB Editor Guide*.

UNIX File Resources (UNIXFILE)

During the publication process, the UNIXFILE attributes are defined. These attributes define the owner and group associations and permissions of each published resource. Each package published has a corresponding UNIXFILE instance within the PRIMARY File Use the CM Admin CSDB Editor to view and modify these attributes.

The following instructions use the CM Admin CSDB Editor. The CM Admin CSDB Editor is currently available for 32-bit Windows platforms. For more information, refer to the *CM Admin CSDB Editor Guide*.

To view the UNIXFILE Class instances using the CM Admin CSDB Editor

1. Go to **Start** → **Programs** → **HP OVCM Admin** → **CM Admin CSDB Editor**.

   The CM Admin CSDB Editor Security Information dialog box opens.

2. If necessary, type a User ID and Password, and then click **OK**.

   The User ID, as shipped from HP, is RAD_MAST. No password is necessary. This may have been changed during installation. You can also change this by selecting the **Change Password** check box and typing the new password in the New Password and Verify New Password text boxes.

3. Double-click **PRIMARY**.
4. Double-click **SOFTWARE**.
5. Double-click **Unix File Resources (UNIXFILE)**.
6. Double-click the appropriate application. The attributes for the UNIXFILE instances for that application appear in the list view.

   To change any instance attribute, double-click the attribute name in the list view. Make your desired changes in the box that opens, and click **OK** when finished.

Published Owner, Group, and Permission Considerations

The UNIXFILE Class contains the attributes ZPERUID and ZPERGID. They define the user ID and group association of the promoted resource. These attributes are populated during the publishing session and reflect the user ID and group association of the resources being promoted. In addition, permission characteristics are captured during publishing and stored in the UNIXFILE.ZRSCRASH attribute. These attributes can be changed using the CM Admin CSDB Editor.

Table 14 Attributes exclusive to the UNIXFILE Class

Attribute	Description
ZPERUID	UNIX user ID associated with the promoted resource. The resource will be owned by this user ID when deployed, providing the CM Application Manager is run by root and the user ID exists on the agent workstation.

Attribute	Description
ZPERGID	UNIX group ID associated with the promoted resource. The resource will be associated with this group when deployed, providing the CM Application Manager is run by root and the group exists on the agent workstation.
ZRSCRASH	This should be a four-digit octal notation of the managed resources permissions (example: 7555). This is populated during the publishing session based on the characteristics of the published resources.
DIRPERMS	Permissions assigned to unmanaged folders that are created when the directory structure does not exist. CM assignes permissions based on umask settings if this attribute is not defined. For example, if the managed file `file.txt` is placed in `/opt/newlocation`, and the directory `/newlocation` does not exist, it will be created. This is an unmanaged directory. If more than one file is installed to a new location, the first one installed determines the assigned permissions.

If the CM Agent is run as a non-root user ID:

- All deployed resources will be associated with the user ID and group of the user ID who is running the CM Application Manager.
- During publishing, the owner and group of the resource is stored in the UNIXFILE instance data. The owner and group attributes within the instance are only applied if the CM agent is run as root for only root has the ability to perform changes in owner and group characteristics.
- CM agent capabilities are limited to the permission constraints of the current user ID and group membership for the UNIX user ID running the connect.
- CM will be unable to deploy to directories where the directory permissions prohibit the non-root user and or group membership to write.
- CM may be unable to set permissions on resources placed under CM Management that are already on the agent workstation though owned by a different UID and/or GID.
- CM will be unable to launch agent methods requiring root authority.

If the CM Application Manager is run as root and:

- If the owner name of the resource, as defined in ZPERUID, and the user ID exist on the agent workstation, the resource will be owned by the UNIX user ID specified.
- If the group name of the resource, as defined in ZPERGID, and the group exist on the agent workstation, the resource will be associated with the UNIX group specified.

To prevent security breaches please note the following:

If the owner of a resource, as defined in ZPERUID, does not exist on the agent workstation, the owner designation of the managed resource will be set to "nobody" (uid 60001).

If the group of a resource, as defined in ZPERGID, does not exist on the agent workstation, the group designation of the managed resource will be set to "nobody" (gid 60001).

Optimizing Services

- Service Optimization uses byte level differencing and its ability to generate patches to recreate original data. A **patch** allows administrators to upgrade data to reflect bug fixes, feature additions, and added information. These patches contain the minimum number of bytes required to fix a flawed program and/or complete software upgrades. These patches are smaller than the data, thus conserving network bandwidth at the expense of CPU overhead.
- The CM Admin Publisher automatically creates components that are eligible for byte-level differencing patching, assuming the component class contains the proper signature attributes as specified in the CM Configuration Server specifications.

Initially, to allow for the functionality of byte level differencing, the following limitations are set:

- Patches will be managed at the SOFTWARE.ZSERVICE level between PACKAGES instances that are hierarchically connected together.
- Patches can only be created between components with the same fully qualified names.
- Patches can only be created for components that contain a signature. 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.

For detailed information, refer to the *CM Admin CSDB Editor Guide*.

Summary

- Publishing is the process of identifying the components of the software or content and organizing them into packages.
- CM publishing mode: Component Selection Mode.
- To publish packages, install the CM Admin Publisher onto a clean computer. To configure applications you must use the CM Admin CSDB Editor.
- Install the CM Admin Publisher onto a machine you will be using for publishing applications.
- You can use Component Selection Mode for packaging simple applications by selecting the individual components that make up the software.
- Use the CM Batch Publisher as an alternative to Component Selection Mode.
- After publishing applications, use the New Application Wizard in the CM Admin CSDB Editor to create a service.
- You can use CM to prepare and distribute maintenance to the CM Application Manager.

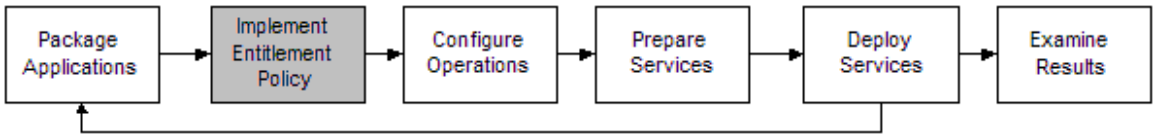
6 Implementing Entitlement Policy

At the end of this chapter, you will:

- Understand how CM can integrate with your existing policy information.
- Understand the POLICY Domain.
- Be able to create new users and assign them to groups for use in simple environments.
- Be able to connect services to groups.

This guide describes the *standard* implementation for the HP OpenView Configuration Management Application Manager (CM Application Manager). Although you will tailor this strategy to meet your organization's needs, we recommend that you review this guide for a comprehensive understanding of the CM Application Manager. This chapter describes how to integrate your existing policy information with CM. It also describes the POLICY Domain, and provides background information about how to create users and groups, assign users to groups, and connect services to groups.

Figure 6 Tasks described in this guide



About Policy Management and CM

As your organization grows and changes, it is your job to manage *who* has access to *what* software. You have invested time and money to determine the best way to handle policy information for your organization. Now, you want to use CM to manage your data. CM allows you to use your *existing* policy information, while using managing your data.

CM can use real-time policy information from:

- Active Directory
- NDS
- iPlanet
- ISOCOR
- SQL Server, Oracle, or Sybase
- SQL 92-compliant (ODBC) data sources
- Any LDAP-compliant directory

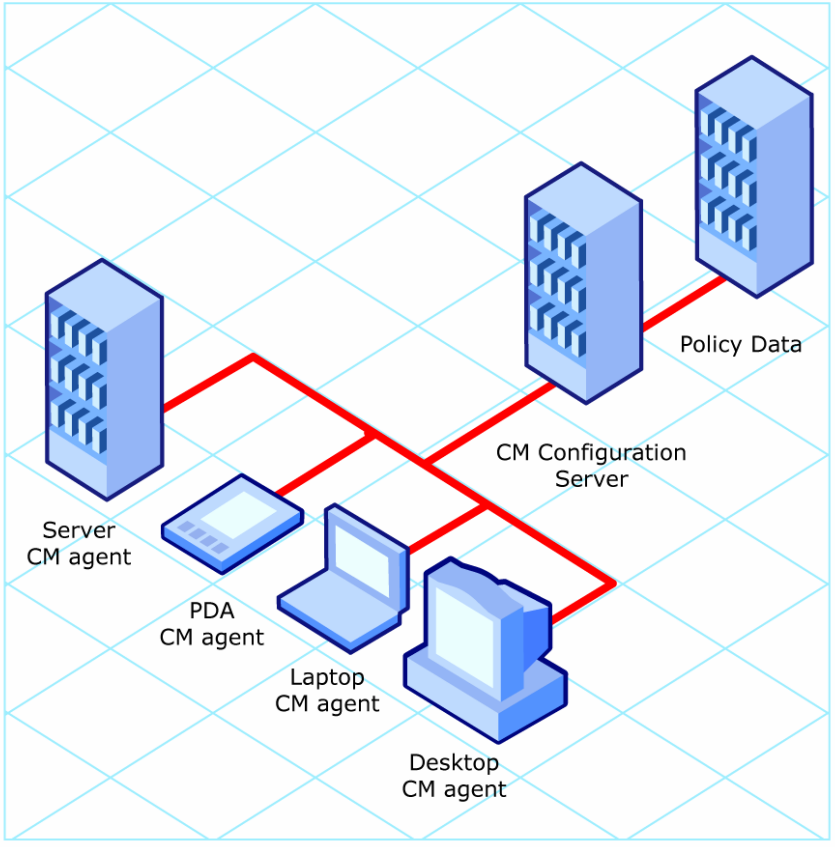
You can continue to use the tools that you are already familiar with to administer policies. And, as you modify group assignments, subscriptions to data are kept up-to-date.

Accessing Existing External Policy Information

When a CM agent connects to the CM Configuration Server, CM retrieves policy information in real-time from the appropriate data stores. In the simplest environment, such as a lab used for testing, you might want CM to search the CM Configuration Server DB for this information. In a large-scale environment where an external policy store already exists, CM can leverage this existing information. This information is sent back to the CM Configuration Server, which determines which data are to be managed for the user, group, or computer.

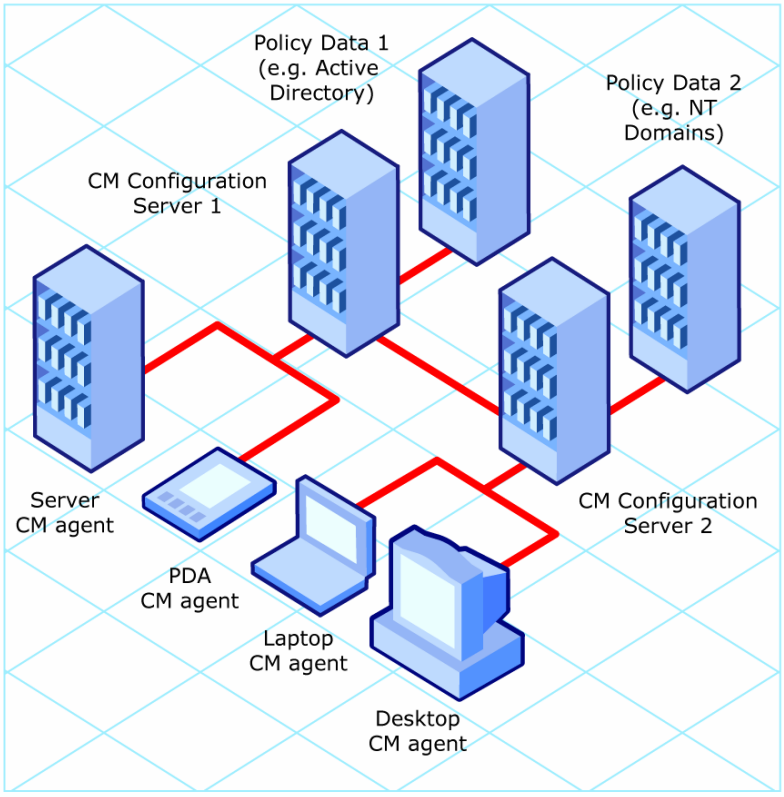
For technical details about integrating your existing policy using the CM Policy Server, see the HP OpenView web site and the *HP OpenView Configuration Management Policy Server Installation and Configuration Guide (CM Policy Server Guide)*.

Figure 7 Policy information from an external source



CM also supports multiple CM Configuration Servers with multiple types of external policy stores. This is especially useful in migration scenarios where you may be consolidating multiple external policy stores over a period. During this time, you can continue to use as many existing policy stores as necessary.

Figure 8 Policy information from multiple external sources



Integrating with Existing External Policy

In order to use real-time policy information from an external source to manage your data, CM must communicate with your policy system. Each policy system has its own interface; some are proprietary, some are standardized.

For technical details about integrating your existing policy with CM, see the HP OpenView web site.

Directories-Based Entitlement

(such as Active Directory and NDS)

If you want to leverage your investment in LDAP-based directory services or SQL-based databases, HP offers the HP OpenView Configuration Management Policy Server (CM Policy Server). The CM Policy Server is a plug-in to the CM Integration Server used for administration purposes such as mapping services to users or computers in the directory tree. The CM Configuration Server can be configured to query the CM Policy Server to determine what services should be distributed and managed for the agent.

The CM Policy Server is an optional feature available from HP. Contact your HP sales representative for details.

See the *CM Policy Server Guide* for more information.

The way that CM integrates with existing policy greatly reduces the total cost of ownership of your environment by allowing you to continue to manage policies from your existing repository while CM manages your data.

About the POLICY Domain

If you are using real-time policy information from an external source to manage your data, you may need to configure a connection from your external policy store to the POLICY Domain in the CM Configuration Server DB. The configuration may vary based on the policy store.

This section is intended to provide you with an overview of the POLICY Domain. Most medium to large organizations will use their existing policy information and will have limited use for this domain. However, in the simplest environment, you can use the POLICY Domain to organize subscribers into logical groups in preparation for distributing software.

In this section, you will learn:

- About the classes in the POLICY Domain.
- How to create users and groups.
- How to assign users to groups.

Once you are familiar with the POLICY Domain and understand the basics of managing policy information within CM, you can extend that knowledge to learn how to integrate your existing policy information with CM. This

information may also be useful if you want to create a simple lab environment to test the management of your data.

The following section uses the CM Admin CSDB Editor, which is available for 32-bit Windows platforms. For more information, refer to the *CM Admin CSDB Editor Guide*.

To access the POLICY domain

1. Go to **Start** → **Programs** → **HP OVCM Admin** → **CM Admin CSDB Editor**.

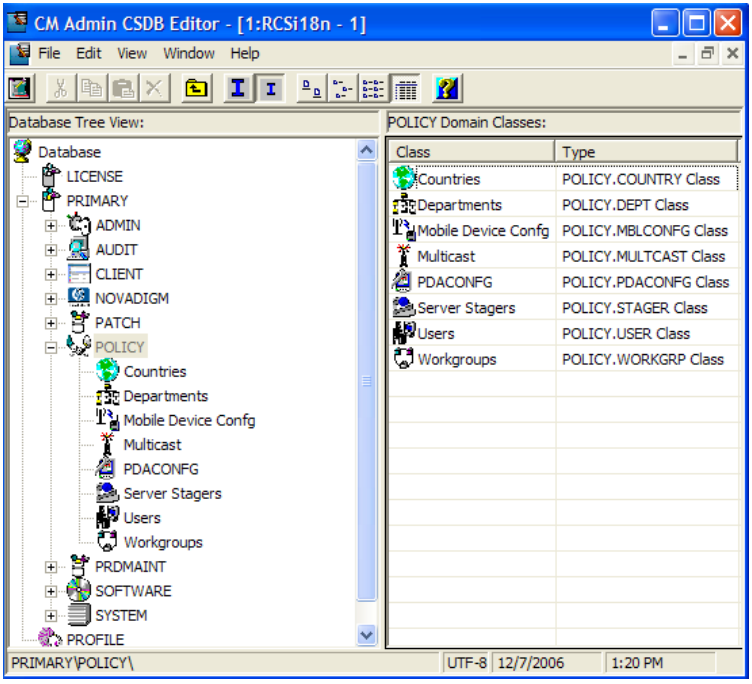
   The CM Admin CSDB Editor Security Information dialog box opens.

2. If necessary, type a User ID and Password, and then click **OK**.

   The User ID, as shipped from HP, is RAD_MAST. No password is necessary. This may have changed during installation. You can also change this by selecting the **Change Password** check box and typing the new password in the New Password and Verify New Password text boxes.

3. Double-click **PRIMARY**.

4. Double-click **POLICY**.



Classes in the POLICY Domain

The POLICY Domain has five default classes, Countries (COUNTRY), Departments (DEPT), Server Stagers (STAGER), Users (USER), and Workgroups (WORKGRP), as described below.

Table 15 Classes in the POLICY Domain

Class	Description	Instance Examples
Countries (COUNTRY)	Use for clock synchronizations with the CM Configuration Server. Do not assign services to this class.	France, Japan, Italy
Departments (DEPT)	Use to group subscribers into departments.	Finance, Customer Service, Manufacturing
Mobile Device Config (MBLCONFG)	Defines the parameters for mobile device configuration when using the Mobility Server.	RmmUser
Multicast (MULTCAST)	Use the MULTCAST class to configure agent computers to use multicasting.	MCast1, Mcast2
PDACONFG (PDACONFG)	This class defines the parameters for PDA configuration.	PDAUser
Server Stagers (STAGER)	Use to define Staging Servers within your distribution network. Also, use to define storage locations on a Staging Server computer.	CDROM, Stager, Server001
Users (USER)	Use to define individual subscribers.	William, John Doe, SSampson

Class	Description	Instance Examples
Workgroups (WORKGRP)	Use to group subscribers into functional groups. For example, a project team may be made up of subscribers from several different departments.	Project Planning, Managers, ABC Project Team

You can also add other classes to the POLICY Domain, as per your organization's needs. For example, if your organization is an insurance company, you may add an AGENTS or OFFICES Class. Or, if your organization is a bank, you might add classes such as BRANCHES or TELLERS to organize your subscribers.

Refer to the *CM Admin CSDB Editor Guide* for information about creating new classes.

Creating Users or Groups

There may be times when you need to create individual users or groups in CM. For example, you might want to create a lab environment used to test the distribution and management of your data. To create a simple environment, you may want to create several users, assign them to groups, and then assign services to the groups.

In this section, you will learn how to create a user in the Users (USER) Class in the POLICY Domain of the CM Configuration Server DB. You can follow the same steps to create a new Workgroups (WORKGRP) instance or Departments (DEPT) instance by substituting the appropriate class name.

In the following example, you will use the CM Admin CSDB Editor to create a new user n the USER Class.

The following instructions use the CM Admin CSDB Editor. The CM Admin CSDB Editor is currently available for 32-bit Windows platforms. For more information, refer to the *CM Admin CSDB Editor Guide*.

To create a new user

1 Go to **Start** → **Programs** → **HP OVCM Admin** → **CM Admin CSDB Editor**.

The CM Admin CSDB Editor Security Information dialog box opens.

2 If necessary, type a User ID and Password, and then click **OK**. The CM Admin CSDB Editor window opens.

3 Double-click **PRIMARY**.

4 Double-click **POLICY**.

5 Right-click **Users (USER)**.

6 Select **New Instance**.

7 In the Create Instance dialog box, type a display name (up to 25 characters) and instance name (up to 25 characters).

8 Click **OK**.

The user instance s created.

Assigning Users to Groups

If you have created several users, you might want to assign them to one or more groups. In the following example, we will use the CM Admin CSDB Editor to assign a user to the Sales department.

The Sales instance, shown in the Departments (DEPT) class in these examples, may not appear in your CM Configuration Server DB. To add this instance (or instances that are appropriate to your organization), follow the procedure To create a new user on page 107. However, instead of right-clicking USER, you would right-click the appropriate class, such as Departments (DEPT).

The following instructions use the CM Admin CSDB Editor. The CM Admin CSDB Editor is currently available for 32-bit Windows platforms. For more information, refer to the *CM Admin CSDB Editor Guide*.

To assign a user to a department

1 Go to **Start** → **Programs** → **HP OVCM Admin** → **CM Admin CSDB Editor**.

The CM Admin CSDB Editor Security Information dialog box opens.

2 If necessary, type a User ID and Password, and then click **OK**. The CM Admin CSDB Editor window opens.

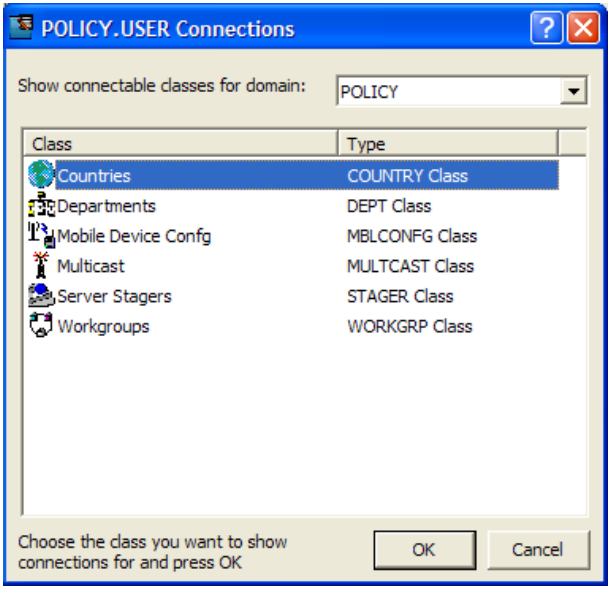
3 Double-click **PRIMARY**.

4 Double-click **POLICY**.

5 Double-click **Users (USER)** to open the list of all user instances.

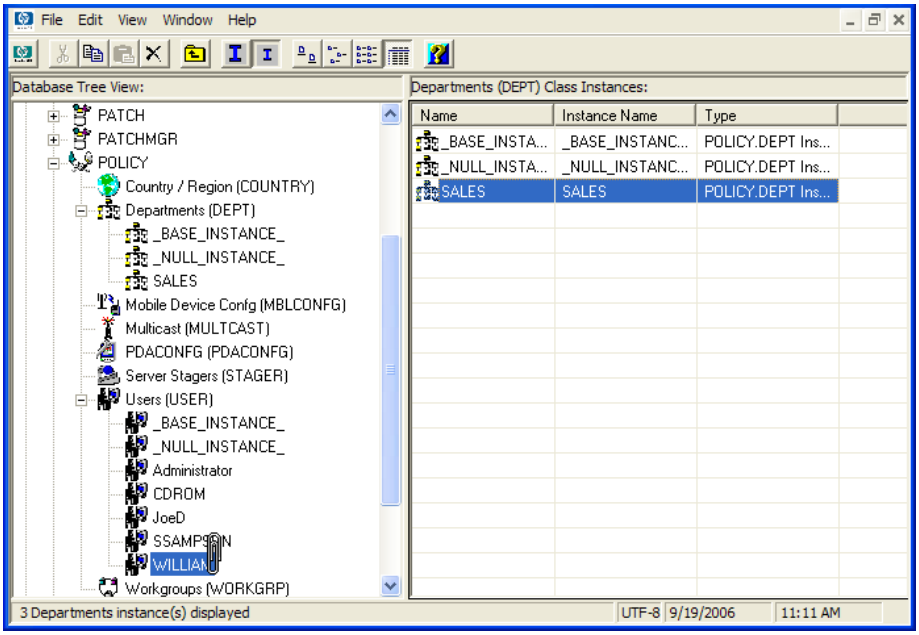
6 Right-click the user instance and select **Show Connections**.

The POLICY.USER Connections dialog box opens. This dialog box displays a list of classes that you can connect the selected instance to.



7 Select **Departments (DEPT)** and then click **OK**. The DEPT Class instances appear in the list view of the CM Admin CSDB Editor. This allows you to easily make a connection between an instance in the DEPT Class and an instance in the USER Class.

8 Select the **Sales** instance from the list view and drag it to the appropriate Users instance. When your cursor turns into a paper clip, release the mouse button.



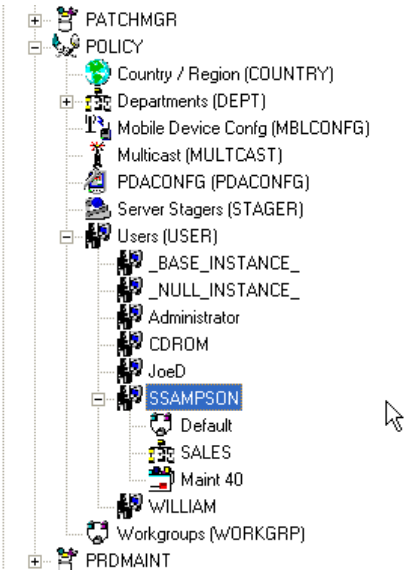
The Select Connection Attribute dialog box opens.

9 Click **Copy** to create the connection.

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

11 Click **OK** when you receive a confirmation.

In the CM ADMIN CSDB Editor tree view, notice that Sales is now listed under the user instance, which indicates that user is part of the Sales department.



Connecting Services to Groups

Whether you are using an external policy source, or you are managing policy within CM, you will need to define the services that your subscribers will receive.

If you are using the CM Policy Server, refer to the *CM Policy Server Guide* for more information.

In this section, you will learn how to connect users and groups to the services that CM will manage. In the following example, we will use the CM Admin CSDB Editor to authorize all subscribers in the Sales department for a sample application.

The following instructions use the CM Admin CSDB Editor. The CM Admin CSDB Editor is currently available for 32-bit Windows platforms. For more information, refer to the *CM Admin CSDB Editor Guide*.

To connect an application to the Sales Department

1 Go to **Start** → **Programs** → **HP OVCM Admin** → **CM Admin CSDB Editor**.

The CM Admin CSDB Editor Security Information dialog box opens.

2 If necessary, type a User ID and Password, and then click **OK**. The CM Admin CSDB Editor window opens.

3 Double-click **PRIMARY**.

4 Double-click **POLICY**.

5 Double-click **Departments (DEPT)** to open the Departments Class.

6 Right-click the **Sales** instance (in the tree view), and from the shortcut menu 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 n application instance from the list view and then drag it to the appropriate Department instance. When your cursor turns into a paper clip as shown below, release the mouse button.

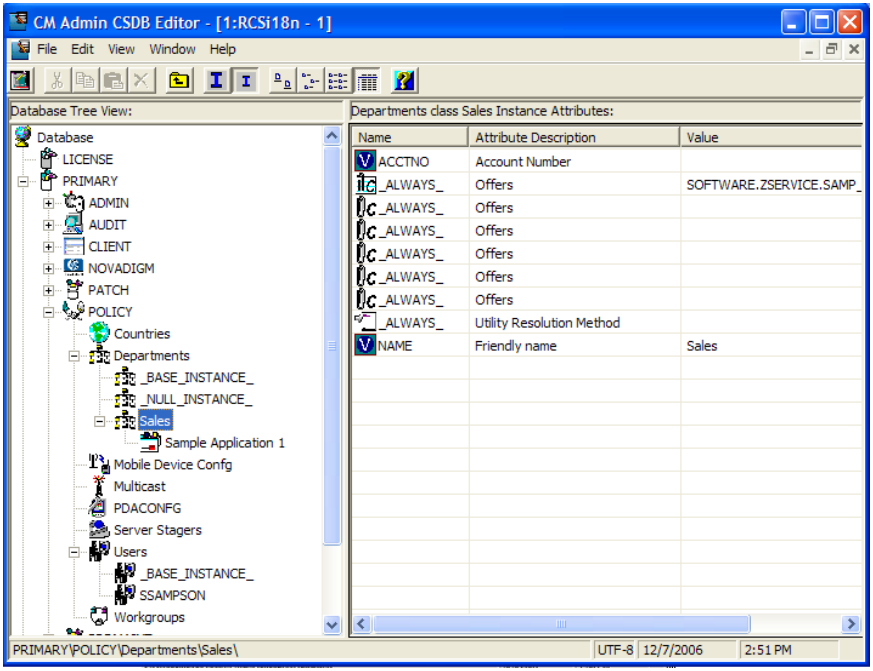
10 The Select Connection Attribute dialog box opens.

11 Click **Copy** to create the connection from Departments.Sales to the application.

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

13 Click **OK** when you receive the confirmation.

In the CM Admin CSDB Editor tree view, notice that the application is listed under the Sales department instance, which indicates that the entire Sales department is now authorized to receive the application.



Notice now that the user instance you created listed in the Users (USER) Class, is part of the Sales department. You can also see that the sample application has been authorized for the entire Sales department. Therefore, as long as the user SSampson is part of the Sales department, CM will manage the sample application on his computer.

You can see how using groups simplifies assigning applications to users. You can modify the applications that the individuals in the Sales department are authorized for simply by manipulating the connections between the applications and the Sales department group. And, you can add users to the Sales department, quickly authorizing them for a series of applications. Or, you can remove users from the Sales department, taking away their authorization to applications.

Summary

- CM can integrate with your existing policy information.
- The POLICY Domain organizes subscribers into logical groups.
- You can create new users and assign them to groups.
- Assign the services to be managed by CM to the appropriate groups.

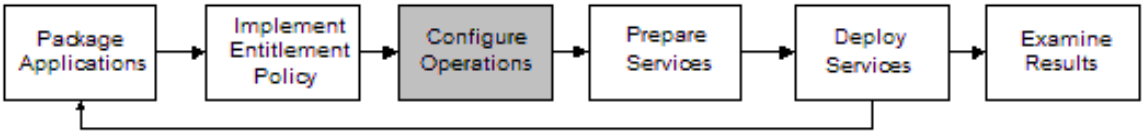
7 Configuring CM Client Operations Profiles

At the end of this chapter, you will:

- Understand the benefits of Client Operations Profiles.
- Know how to implement Client Operations Profiles.
- See a simple implementation example.

This guide describes the *standard* implementation for the HP OpenView Configuration Management Application Manager (CM Application Manager). Although you will tailor this strategy to meet your organization's needs, we recommend that you review this guide for a comprehensive understanding of the CM Application Manager. This chapter discusses how to configure the operations of your CM agent. It describes the CLIENT Domain, and contains information about providing failover capabilities, designating servers for an agent based on criteria you set, controlling trouble shooting settings, hardware scan settings, and user interface settings.

Figure 9 Tasks described in this guide



Client Operations Profiles

If you have multiple CM Configuration Servers, CM Proxy Servers, or want to store files that manage applications on a local CD-ROM, you might want to reconfigure the CM agent before connecting to the CM Configuration Server. Use Client Operations Profiles to do this. Client Operations Profiles, along with the CM object-oriented schema, allow you to control certain agent behaviors based on any attribute from an agent object. Benefits of using Client Operations Profiles include, but are not limited to:

- Failover capability for CM servers.
- Dynamic assignment and selection of an agent computer's available servers based on network location, speed, or other criteria.
- Defining which CM Configuration Server to use based on its functional role, allowing for load balancing among CM Configuration Servers.
- Enhanced diagnostics capabilities.

To use Client Operations Profiles, you must be using the CM Application Manager version 3.1 or higher and the CM Configuration Server DB version 3.1 or higher.

The CLIENT Domain

The CLIENT Domain in the CM Configuration Server DB controls Client Operations Profiles. It includes six classes with sample instances that you can use to configure your CM agent computers operations. We provide an example of using the agent computer's network location, stored in the ZCONFIG object to prioritize the downloadable locations for application data for each agent computer. The six classes are:

- **Core Settings (SETTINGS)**
  Use an instance in the SETTINGS Class to define how to use your Server Access Profile, to define scripts you want to use, and to set other global parameters.
- **Diagnostics (DIAGS)**
  Use instances in this class to override tracing levels set on the CM agent.
- **Hardware Scan Config (RADHWCFG)**
  Use an instance in this class to control the type of hardware scan that the CM agent should perform.
- **Network Location (LOCATION)**
  Use the LOCATION Class to group users based on a location, such as their subnet.
- **RSM UI Preferences (RADUICFG)**
  Use instances in this class to manage the display of the CM Application Self-service Manager User Interface.
- **Server Access Profile (SAP)**
  Use instances in the Server Access Profile (SAP) Class to define CM Configuration Servers and possible data access points for CM-managed services.

Recommendations

We recommend the following for using Client Operations Profiles.

- Use our Professional Services to help you implement this feature.
- Configure Client Operations Profiles *only* if you fully understand this process.
- Avoid single point of failure in all aspects, for servers of both types and for each role. Create redundancy where possible.
- Use base and null instances for unknown and new network addresses.

Implementing Client Operations Profiles

Use CLIENT Domain class instances to customize the profiles to meet your enterprise's needs. There are five major steps for implementation.

1 Identify Servers.

2 Create Server Access Profile instances.

3 Set criteria for Server Access Profile instances.

4 Set priorities for Server Access Profile instances.

5 Enable Client Operations Profiles.

The following sections describe each of these steps. Before beginning this procedure, a discussion of Server Types and Roles is required.

Understanding Server Types and Roles

A Server Access Profile (SAP) is a generic way to define all possible data access points for a service. A SAP can be a CM Configuration Server, CM Proxy Server, or CD-ROM drive. Client Operations Profiles allow you to identify and prioritize data access points without the need to use additional customized scripts.

Before beginning this process, you must have an understanding of server types and roles. These are reflected, respectively, in the TYPE and ROLE attributes of the SAP Class. A server can either be identified as an RCS or DATA type. Only a CM Configuration Server can be identified as type RCS. A CM Configuration Server, CM Proxy Server, or CD-ROM drive can be identified as DATA in the TYPE attribute. Use DATA type only for servers from which the agent will download applications.

In addition, each CM Configuration Server can have a role, or function, specified in the ROLE attribute of the SAP Class. Possible roles are:

- **Client Operations Profiles (O)**
  Use this CM Configuration Server to get the agent computer's Client Operations Profile.
- **Service resolution (S)**
  Use this CM Configuration Server to resolve the agent computer's services.
- **Agent self maintenance (M)**
  Use this CM Configuration Server to perform agent self-maintenance.

- **Reporting (R)**
  Use this CM Configuration Server for storing reporting objects from the agent computer. These objects are stored in the PROFILE File in the CM Configuration Server DB.
- **Data download (D)**
  Use this CM Configuration Server to download application data to the agent computer.
- **All (A)**
  Use this CM Configuration Server for any of the roles listed above.

A CM Proxy Server or CD-ROM can only serve the role of Data download (D). A CM Configuration Server can serve any of the above roles.

When an agent is ready to download files, the agent will first use servers with TYPE=DATA in order of priority set in the LOCATION Class. If a server with TYPE=DATA, has a ROLE of anything other than D, it will be skipped and not used for data download.

If the agent is still unable to download all the needed files after processing all servers with TYPE=DATA, then the agent will begin processing servers of TYPE=RCS in order of priority. In order to use a CM Configuration Server for data download, two conditions must be met:

- The SETTINGS.RCSDATA attribute for the agent computer must be set to Y.
- The SAP.ROLE for the CM Configuration Server must be either D or A.

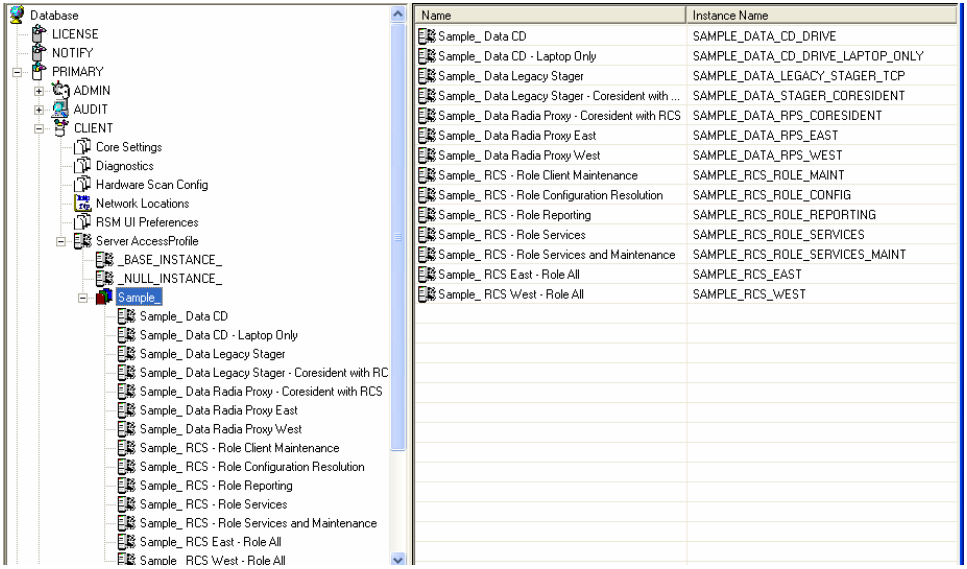
Task 1 Identify CM Servers

Identify your CM Servers, and determine the type and role for each. Types are defined in the TYPE attribute of the SAP Class. Roles are defined in the ROLE attribute of the SAP Class. Your CM Server can be a CM Configuration Server, CM Proxy Server, or CD-ROM. CM Servers with TYPE=DATA, must have a role of Data download (D). CM Configuration Servers, set to TYPE=RCS, can serve many roles. You will need to decide which roles your CM Configuration Servers can perform.

Only a CM Configuration Server can be designated as anything other than data download. For all other SAP instances, if you specify anything other than data download, the SAP will be ignored.

Task 2 Create Server Access Profile Instances (SAP)

Use CM Admin CSDB Editor to create one SAP instance for each Server Access Profile. Table 16 below describes the attributes in the SAP Class. We provide you with samples for each type of Server Access Profile in the CM Configuration Server DB. The figure below displays the SAP samples we provide.



Use the CM Admin CSDB Editor to make a copy of the instance that most closely approximates the server type and role for which you need an SAP instance. After copying the instance, use Table 16 to configure the instance for your enterprise.

Instances in the SAP Class are used before the IP specified in the radskman command line.

Table 16 Attributes of the SAP Class

Attributes	Description
ZSTOP00n	**Expression Resolution Method** Use a ZSTOP expression to stop the process from completing if certain requirements are met. For example, you may want to prevent a laptop computer from using this SAP.
NAME	**Friendly Name** Friendly name of the SAP instance.

Attributes	Description
TYPE	**Type [RCS/DATA]** Specify the Type of CM server. Set to RCS if using CM Configuration Server. Set to Data for CM Proxy Server or CD-ROM. If the agent computer is unable to reach any of its Server Access Profile, then the agent will default to the last known CM Configuration Server.
URI	**Universal Resource Identifier** Create the Universal Resource Identifier to specify the CM Configuration Server, or CM Proxy Server. See Table 17 on page 124 for examples.
ROLE	**RCS Role A,O,S,M,R,D** Specifies the role of the SAP. Specify as many values as are needed separated by a comma. A blank or null value defaults to ALL. Possible values are A = all, O = Client Operations Profiles, S = Service Resolution, M = Agent Self Maint, R = Reporting, D= Data Download. Default: The default value is A. Note: Only a CM Configuration Server can be designated as anything other than type D. For servers where TYPE=DATA, if you specify anything other than D, that SAP instance will be skipped.
ENABLED	**Enable SAP [Y/N]** Specify if this SAP is enabled (Y) or disabled (N). If the variable is blank or non-existent, then this SAP is enabled. Default: The default value is Y.
TIMEOUT	**Communications Timeout (0-3200)s** Specify the timeout in seconds. This will override agent timeout (ZMASTER.ZTIMEO) if it contains a valid numeric value. If blank, then use existing variable value on agent.
PUSHBACK	**Push Back (0-999 retries)** Set to 0 to skip a CM Configuration Server if the CM Configuration Server pushes back on the agent connect. Set to 1 - 999 for number of retries if the CM Configuration Server pushes back. Default: The default setting is 0.

Attributes	Description
THROTYPE	**Throttle [NONE/ADAPTIVE/RESERVED/]** This attribute applies to Windows only. Type of bandwidth throttling to use. • Set to **ADAPTIVE** to yield to other services that are using the network. • Set to **RESERVED** to allow for a specific reservation of the bandwidth. It is the maximum percentage of network bandwidth to use. • Set to **NONE** for no bandwidth throttling, and use the maximum available bandwidth. This will override agent bandwidth throttling if it contains a valid value. If blank, then use existing variable value on the agent computer.
BANDWDTH	**Bandwidth Percentage (1-99)** Specify the percentage of bandwidth to use between 1 and 99. If blank value or non-existent variable, then use all of the bandwidth. This attribute applies to Windows only. This will override agent bandwidth setting if it contains a valid value. If blank, then use existing variable value on agent computer.
STREAM	**Enable Streaming [Y/N]** Specify Y to use streaming. This will override the agent setting in ZMASTER.ZNORSPNS. Default: The default value is N. Caution: Streaming is not suitable for all network environments. Consult your network administrator before setting this to Y.
PROXY	**Internet Proxy URI** *Do not modify.* The Internet proxy URI through which the agent will connect to the SAP. Maintained by agent.
PRIORITY (&(LOCATION.SAPPRI))	**Selection Priority** *Do not modify.* The SAP obtains its priority by looking at the priority specified in the LOCATION Class.

Attributes	Description
PRODUCT	**Product Filter** Specify which types of CM agents can use this SAP instance. Specify multiple agents separated by a comma. Below are suggested identifiers for each CM agent: CM Application Manager: CM-AM CM Inventory Manager: CM-IM CM Application Self-service Manager: CM-ASM CM OS Manager: CM-OSM CM Patch Manager: CM-PATCH On your radskman command line, specify which products to filter by using the product parameter. For example, if this SAP should only be used by CM Application Manager, then you may want to set this attribute to CM-AM. Then, set product to CM-AM on your radskman command line. Default: Blank means that all products can use this SAP instance.
FILTER	**Filter Expression [Obj.Var = Value]** Use this attribute to filter the SAP based on any available object attribute. For example, if you only wanted to use this SAP for a specific service, specify APPINFO.ZOBJNAME=GS-CALC. Note: The ZSERVICE object is not available during installation. Use the APPINFO object instead. Appinfo, located in the service's `LIB` directory, is a copy of the service's instance from the ASERVICE object.

If you want to override the use of the SAP object, add the RCSURI parameter to the radskman command line. RCSURI should be in the same format as the Universal Resource Identifier. For the syntax of this parameter, see Table 17 on page 124.

Creating the Universal Resource Identifier

For each instance of the SAP Class, you will need to identify a URI (Universal Resource Identifier) for the Server Access Profile (SAP). RFC 1630 proposes the following format:

```
scheme:scheme specific format
```

where the scheme is usually the network protocol such as HTTP or TCP. If the scheme-specific format has slashes, it indicates a hierarchical path. Universal Resource Locator (URL) is a form of URI where the scheme specific format is defined as:

```
//user:password@host:port/url path
```

Details on how to access the specified resource are defined in the URL path. The most prevalent form is: `//host:port/url path`. Table 17 contains examples for specifying the URI.

Table 17 URI Example

SAP Type	URI and TYPE attributes are set to:
CM Configuration Server over TCP/IP using default port of 3464	URI = tcp://ovcmcs:3464 TYPE = RCS
CM Configuration Server over TCP/IP using port 7800	URI = tcp://ovcmcs:7800 TYPE = RCS
Configuration Server using SSL on port 443	URI = tcps://ovcmcsssl:443 TYPE = RCS
CM Proxy Server using HTTP	URI = http://ovcmps:3466 TYPE = DATA
CD-ROM	URI = file://&(ZCONFIG.ZHDWCDDR)|/DATA/ TYPE = DATA

Task 3 Set Criteria for each SAP Instance

After creating your SAP instances, you must decide how you are going to segment your enterprise. You may want to assign a SAP to an agent computer based on its subnet. If so, use CM Admin CSDB Editor to create one LOCATION instance for each subnet. In the sample database, there are two locations, Sample_Location East and Sample_Location West.

The ZCONFIG object for an agent computer includes an attribute called NETLOC. The ZCONFIG.NETLOC variable identifies the agent computer's subnet using underscores instead of periods. You may want to name the LOCATION instance based on possible subnets of your agent computers so that you can easily connect a user to their appropriate LOCATION Instance based on their subnet. For example, if you have a subnet of 10.10.10.1, create

a LOCATION instance called 10_10_10_0. Enable Client Operations Profiles on page 128 shows you how to connect the subscribers to the appropriate LOCATION Class using the agent computer's ZCONFIG.NETLOC variable.

Task 4 Set Priority for each SAP for each Location

Use instances in the LOCATION Class to define the priorities of your Server Access Profile based on location criteria. The figure below shows one of the samples that we provide. The priority for a Server Access Profile is defined directly above the connection to that SAP instance in the SAPPRI attribute. Lower numbers have a higher priority. For example, SAP.SAMPLE_RCS_EAST has a priority of 10, and SAP.SAMPLE_DATA_RPS_EAST has a priority of 40.

The SAPPRI can be any whole number from 01 to 99. The SAPs do not need to be listed in the LOCATION instance in their priority order.

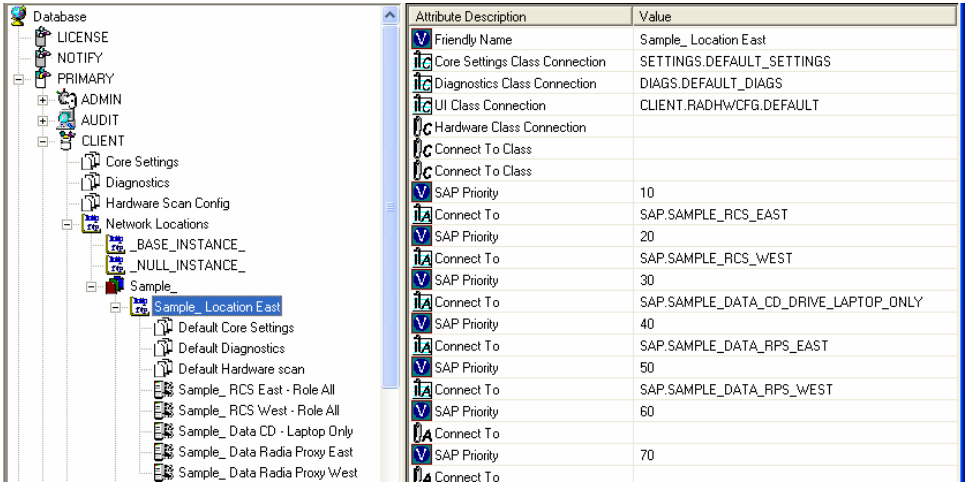


Table 18 on page 126 describes the attributes of an instance in the LOCATION Class. For example, if you want your SAMPLE_RCS_ WEST to be used before your SAMPLE_RCS_ EAST, increase its priority to 7. To do this, change the SAPPRI for SAMPLE_RCS_WEST to 7 as shown in the figure below.

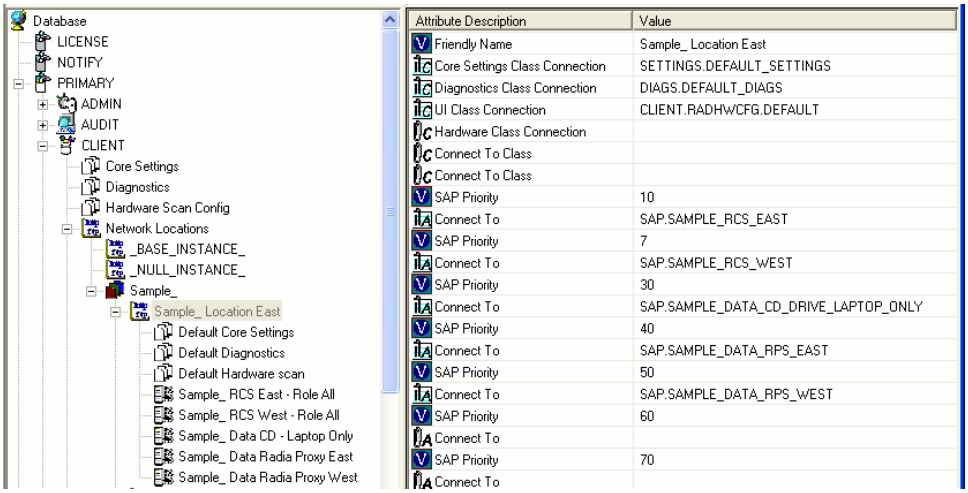


Table 18 Attributes of the LOCATION Class

Attribute	Description
COPLNAME	**Friendly Name** Type the friendly name of the instance.
ALWAYS	**Core Settings Class Connection** Specify an instance in the SETTINGS class. Default: The default connection is SETTINGS.DEFAULT_SETTINGS.
ALWAYS	**Diagnostics Class Connection** Specify an instance in the DIAGS Class. Default: The default connection is DIAGS.DEFAULT_DIAGS.
ALWAYS	**UI Class Connection** Specify an instance in the RADUICFG Class.
ALWAYS	**Hardware Class Connection** Specify an instance in the RADHWCFG Class.
ALWAYS	**Class Connection** Specify an instance in any class to connect to this LOCATION instance.

Attribute	Description
SAPPRI	**SAP Priority** Specify the priority of the SAP instance referenced in the _ALWAYS_ class connection below this attribute. Default: The SAP referenced in the connection below this instance has a priority of 10 by default.
ALWAYS	**Connect to** Specify an SAP instance for the priority entered into SAPPRI above this attribute. This SAP will default to a priority of 10.
SAPPRI	**SAP Priority** Specify the priority of the SAP instance referenced in the _ALWAYS_ class connection below this attribute. Default: The SAP referenced in the connection below this instance has a priority of 20 by default.
ALWAYS	**Connect to** Specify an SAP instance for the priority entered into SAPPRI above this attribute. This SAP will default to a priority of 20.
SAPPRI	**SAP Priority** Specify the priority of the SAP instance referenced in the _ALWAYS_ class connection below this attribute. Default: The SAP referenced in the connection below this instance has a priority of 30 by default.
ALWAYS	**Connect to** Specify an SAP instance for the priority entered into SAPPRI above this attribute. This SAP will default to a priority of 30.
SAPPRI	**SAP Priority** Specify the priority of the SAP instance referenced in the _ALWAYS_ class connection below this attribute. Default: The SAP referenced in the connection below this instance has a priority of 40 by default.

Attribute	Description
ALWAYS	**Connect to** Specify an SAP instance for the priority entered into SAPPRI above this attribute. This SAP will default to a priority of 40.
SAPPRI	**SAP Priority** Specify the priority of the SAP instance referenced in the _ALWAYS_ class connection below this attribute. Default: The SAP referenced in the connection below this instance has a priority of 50 by default.
ALWAYS	**Connect to** Specify an SAP instance for the priority entered into SAPPRI above this attribute. This SAP will default to a priority of 50.
SAPPRI	**SAP Priority** Specify the priority of the SAP instance referenced in the _ALWAYS_ class connection below this attribute. Default: The SAP referenced in the connection below this instance has a priority of 60 by default.
ALWAYS	**Connect to** Specify an SAP instance for the priority entered into SAPPRI above this attribute. This SAP will default to a priority of 60.
SAPPRI	**SAP Priority** Specify the priority of the SAP instance referenced in the _ALWAYS_ class connection below this attribute. Default: The SAP referenced in the connection below this instance has a priority of 70 by default.
ALWAYS	**Connect to** Specify an SAP instance for the priority entered into SAPPRI above this attribute. This SAP will default to a priority of 70.

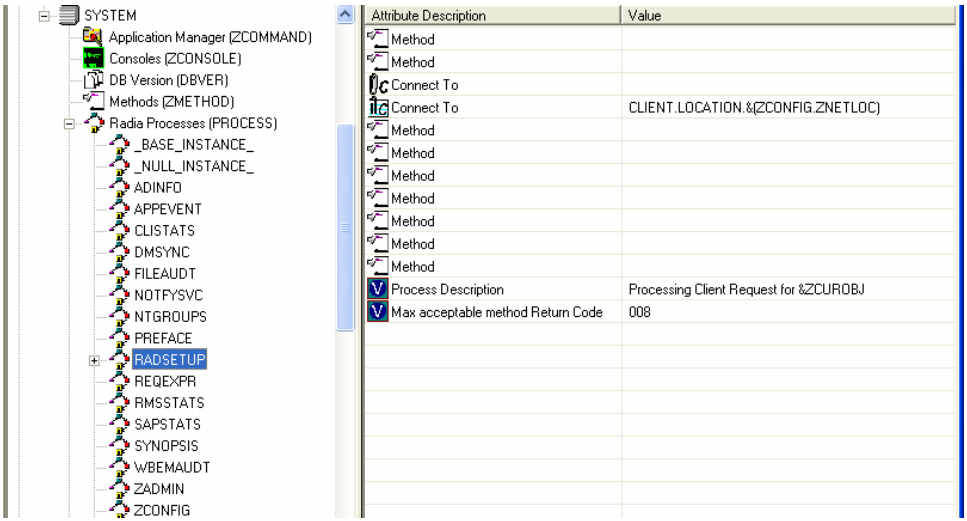
Task 5 Enable Client Operations Profiles

There are two phases to this step. First, you will need to create a process on the CM Configuration Server so that the objects associated with Client

Operations Profiles are resolved. Second, you will need to enable the agent computer to use Client Operations Profiles.

Enable on the CM Configuration Server

To enable Client Operations Profiles, you must create a new instance called RADSETUP in the Processes (PROCESS) class in the SYSTEM Domain. After creating the instance, make a class connection to CLIENT.LOCATION.&(ZCONFIG.ZNETLOC). This will map the user to its appropriate LOCATION instance based on the ZNETLOC attribute in the agent computer's ZCONFIG object. The ZNETLOC attribute identifies the agent computer's subnet using underscores instead of periods to separate the four octets. Instances of the LOCATION Class must be defined to match each of the subnets in your environment to be able to dynamically assign Client Operations Profiles configuration settings based on locations within your network. The next figure shows an example.



Enable on the Agent

By default, Client Operations Profiles are disabled on agent computers for backwards compatibility with older version of CM. There are three ways to enable Client Operations Profiles on the agent computer. Choose your method based on whether the CM agent has already been installed, and the method that suits your needs best.

If you have not already installed the CM agent,

- You can customize the `install.ini` to add the COP variable to the RADSETUP object. To do this, add one line to the [Objects] section of the `install.ini`. The figure below shows an example of the section with a new line.

  The following code shows the RADSETUP line added to the Objects section of the `install.ini`.

```
[Objects]
; Set CM object attribute values
; A value of _NONE_ will set the attribute to blank
;
RADSETUP_COP=Y
;ZMASTER_ZDSTSOCK=
;ZMASTER_ZIPADDR=
;ZMASTER_ZNTFPORT=3465
```

If you want to enable Client Operations Profiles on already existing CM agents:

- You can use a rexx method, `initmeth.rex` to add and set the COP variable in the RADSETUP object. `Initmeth.rex` runs each time a First Refresh Catalog is called and can build the RADSETUP object with COP=Y to enable Client Operations Profiles. The figure below shows the lines to add to your `initmeth.rex`. Be sure to deploy the updated `initmeth.rex`.

  Add the following lines to the `initmeth.rex`:

```
/* Sample INITMETH.REX to Enable the COP */
call edmget('RADSETUP',0)
RADSETUP.COP = 'Y'
call edmset 'RADSETUP'
```

- You can use the COP parameter of radskman to enable or disable Client Operations Profiles. To enable, add COP=Y to your list of parameters for radskman. *This will only enable or disable Client Operations Profiles for this agent connect.* Use `initmeth.rex` as shown above to enable COP for all agent connects or create a variable in CLIENT.SETTINGS called COP, and set the value to Y. Ultimately, both of these methods will create a COP attribute in the RADSETUP object with a value of Y. Alternatively, if you need to disable Client Operations Profiles after enabling it, run radskman with COP=N to disable for *that* agent connect only.

Additional Classes in the CLIENT Domain

There are two additional classes in the CLIENT Domain used for customizations and diagnosis. Define your own scripts to be used during the agent connect with the Core Settings (SETTINGS) Class. Set tracing levels and use other diagnostic tools by configuring the attributes in the Diagnostics (DIAGS) Class.

Core Settings (SETTINGS)

Use an instance in the SETTINGS Class to define how to use your Server Access Profile, define scripts you want to use in pre-configuration processing, and set other global parameters.

If similar attributes exist in both the SETTINGS and SAP Classes, the attribute in the SAP Class will be used.

Table 19 Attributes of the SETTINGS Class

Attributes	Description
COPSNAME	**Friendly Name** Type the friendly name of the instance.
SAPPING	**Ping all SAP [Y/N]** Set to Y if the agent should ping all of the SAPs. If EQUISORT is set to S, then you must set SAPPING to Y. A result reflecting the speed of the connection will be returned and stored in the SPEED attribute in the SAPSTATS object. Default: The default setting is N.
PUSHBACK	**Push Back (0-999 retries)** Set to 0 to skip a CM Configuration Server if the CM Configuration Server pushes back on the agent connect. Set to 1 to 999 for number of retries if the CM Configuration Server pushes back. Default: The default setting is 0.

Attributes	Description
EQUISORT	**Secondary SAP Priority [R/S]** If several SAP instances have the same priority, set this to R to randomly select which one to use. Set to S to use the SAP with faster network speed. SAPPING must be set to Y to use EQUISORT= S. Use R for workload balancing. Default: The default setting is R.
USELSAP	**Use Last SAP [N/Y]** Set this Y to specify that the last SAP used in this agent connect should be the SAP used for all remaining services to be resolved. Use of SAPs with type of DATA is at the service level. If set to N, then the agent will go through the SAPs in priority for each service. During an agent connect, if a service has to go to a lower priority SAP to complete the data download, decide if you want the remaining services to continue from this SAP (USELSAP=Y) or go back to the highest priority DATA SAP to search for files for the next service (USELSAP=N) and to continue through the SAP priorities. Default: The default setting is Y.
RCSDATA	**Download DATA from RCS [Y/N]** After using all of the TYPE = DATA SAPS, if all the needed data has not been downloaded then specify Y to go to SAPs with TYPE = RCS. If you do not want the agent computers to use CM Configuration Servers, set RCSDATA to N. Default: The default setting is Y.
ADINFO	**Query Active Directory Info [Y/N]** Specify Y if you want to collect the agent computer's active directory information. The information is stored in the ADINFO object in the `RADSETUP` directory. The default location for the `RADSETUP` directory is the Agent lib directory. This information will be sent to the CM Configuration Server for all resolution processes. Default: The default setting is Y.

Attributes	Description
ZGRPINFO	**Query NT User Group Info [Y/N]** Specify Y if you want to collect the agent computer's Windows NT user group information. This information will be reflected in the NTGROUPS object in the `RADSETUP` directory. The default location for the `RADSETUP` directory is the Agent lib directory. This information will be sent to the CM Configuration Server for all resolution processes. Default: The default setting is Y.
LSCRIPT	**Disable Connect on UI Reboot [Y/N]** If you have set a service to perform an immediate reboot and you run radskman from a login script, set this to Y to run radskman from the login script. If you have set a service to perform an immediate reboot, and you want radskman to be restarted in the User context when a user logs, set this to N. In other words, if your users are configured to connect to CM Configuration Server when they log in, set this to N. If you want to do an immediate reboot with context=u, and you want the user to re-establish connection with the CM Configuration Server, then set LSCRIPT=Y. For more information on reboot options, see the section Restarting the Agent computer on page 148. Default: The default setting is Y.
ALWAYSD	**Always Download CFG Objects [Y/N]** Set to Y to always download pre-configuration objects. This guarantees that your SAP or persistent objects are downloaded even if nothing has changed. If your SAP agent object is corrupted for any reason, then it will be re-downloaded even if the desired state did not change. In addition, if one of the variables is a substitution then you will download the object with the new values since a variable change by substitution does not change the desired state. Default: The default setting is Y.

Attributes	Description
ALWAYSS	**Always Upload CFG Objects [Y/N]** Set to Y to always upload all objects in the RADSETUP directory to the CM Configuration Server. The default location for the RADSETUP directory is the Agent lib directory. Set to N to prevent the objects from being sent. Default: The default setting is Y.
EXBSETUP	**Pre Config Resolution Script** Specify a script to run before pre-configuration processing. This script must be in the agent computer's IDMSYS directory. The default location is /opt/HP/CM/Agent. Default: The default setting is PRESETUP.REX.
EXASETUP	**Post Config Resolution Script** Specify a script to run after pre-configuration processing. This script must be in the agent computer's IDMSYS directory. The default location is /opt/HP/CM/Agent.
CMETHOD	**Post Catalog Script** Specify a script that can run after catalog resolution, but before service processing.
EXBOUTBX	**Pre Outbox Script** Specify a script that can run after service processing, but before the objects in the outbox are flushed to the CM Configuration Server.
EXBEXIT	**Post Connection Script** Specify a script to execute before radskman ends. If you are doing a customized reboot process, this is where you would specify it. This script must be in the agent computer's IDMSYS directory. The default location is /opt/HP/CM/Agent. Note: Client Operations Profiles must be enabled on the agent for the EXBEXIT to be used. If Client Operations Profiles are not enabled, the EXBEXIT will be ignored.

Attributes	Description
TIMEOUT	**Communications Timeout (0-3200)s** Specify the timeout in seconds for the Server Access Profile (SAP). This will override the agent timeout (ZMASTER.ZTIMEO) if it contains a valid numeric value. If the value is blank, then the agent will use the existing timeout value on agent.
THROTYPE	**Throttle [RESERVED/ADAPTIVE/NONE/]** This attribute applies to Windows only. Type of bandwidth throttling to use. • Set to **ADAPTIVE** to yield to other services that are using the network. • Set to **RESERVED** to allow for a specific reservation of the bandwidth. It is the maximum percentage of network bandwidth to use. • Set to **NONE** for no bandwidth throttling, and use the maximum available bandwidth. NONE is the default.
BANDWDTH	**Bandwidth Percentage (1-99)** This attribute applies to Windows only. Specify the percentage of bandwidth to use between 1 and 99. If the value is blank or the variable does not exist, then all of the bandwidth will be used.
RADTRAY	Radtray Command Line Arguments Set command line arguments you want to use for the CM System Tray. Specify Y for the first argument to enable the CM System Tray, and N to disable it. If set to Y, you can then specify other parameters separated from the Y with a comma. Possible parameters are: /C — Show the CM System Tray in console mode when it starts. /NOCANCEL — Hide the Cancel button. /NOPAUSE — Hide the Pause button. /D — Add debug message to the log for troubleshooting. Example: Y, /C /NOPAUSE enables the CM System Tray in console mode and does not display the PAUSE button.

Attributes	Description
USEDEFS	**Use Default SAP [Y/N]** If a SAP cannot be found for the needed ROLE, specify Y to default to the CM Configuration Server set on the command line.
DEFROLE	**Default SAP ROLE (A,O,S,M,R)** Specify roles for the CM Configuration Server specified on the command line. If not specified, the ROLE is set to A (All), and the CM Configuration Server will be able to perform any ROLE. Note: USEDEFS must be set to Y to use DEFROLE.
RAD2XUI	**Enable RADUI 2.x** Specify Y to view the vintage UI dialogs. Use this if you are not using System Tray or if you want a message to pop up on the screen in addition to the CM System Tray.
RSTROPT	**Bandwidth Checkpoint Restart** Use this attribute to determine when a file is eligible for checkpoint restart based on calculated network bandwidth. This will apply to *all* files to be downloaded in this agent connect. Specify eligibility in the format (Below Threshold limit, Network Threshold Value, Above Threshold). Suppose you set RSTROPT to 100 KB, 86 KB, 10 MB. First the agent calculates the network bandwidth. One of two scenarios will apply: If the network bandwidth is under 86KB, the file size is compared to 100KB. If the file size is over 100KB, checkpoint restart is enabled for that file. If the network bandwidth is over 86 KB, the file size is compared to 10 MB. If the file size is over 10 MB, checkpoint restart is enabled for that file.
DISKFREE	**Minimum Free Disk Space Threshold** Specify a minimum of free disk space for CM to maintain. If a service is over the limit, it will not be installed.

Attributes	Description
REMUNINS	**Allow Remote Notify Uninstall [Y/N]** Specify Y to stop notifies from remote machines from uninstalling a service. This does not stop applications from being un-installed as part of a policy change if a normal agent connect is started from a remote notify. The remove notify string must contain the text req="Un-install."
DETPROXY	**Internet Proxy detection [Y/N]** Set to N to skip running Internet proxy detection at the beginning of the agent connect.
ACTMAINT	**Maintenance Activation [I/D/P]** The CM maintenance module, upgrdmaint, processes all maintenance activities. Upgrdmaint can be launched by radskman immediately after the maintenance is staged or on an independent schedule. Note: The mnt parameter of radskman must be set to Y for maintenance to be processed. • Set ACTMAINT to I (Immediate) to download maintenance files and immediately activate them. CM Application Self-service Manager subscribers will receive a dialog box showing just an OK button that the CM Application Self-service Manager needs to be updated. CM Application Self-service Manager will close, install maintenance, and then restart. • Set to D to defer maintenance activation. Maintenance files are downloaded, but not activated. To activate maintenance, you can call radsksman req="Self Maintenance" or call upgrdmaint directly using a timer or other method. • Set to P to prompt CM Application Self-service Manager users. A dialog box will display stating that maintenance is available, but the subscriber will be given the option to cancel. The files are downloaded, but not activated. The subscriber will be prompted again at the next check for maintenance by the CM Application Self-service Manager Interface. P is the same as I for Application Manager users.

Attributes	Description
	Default: I
SENDRPT	**Send Reporting Object [I/D]** Set to D to defer sending all reporting objects to CM Configuration Server at the end of agent connect. Usually, the reporting objects for each service, such as APPEVENT, CLISTATS, and ZSVCSTAT, are sent to the CM Configuration Server immediately (I) after they are created. This requires multiple disconnects and reconnects to the CM Configuration Server. Default: I

Diagnostics (DIAGS)

Use this class to override default trace settings on the CM agent computer. In addition, you can set parameters for running the radstate program. Radstate is a diagnostic module designed to give an overview of the current state of the CM agent. The information in the radstate output is based on data retrieved from numerous CM agent objects. For additional information on radstate, see the HP OpenView web site.

Instances of this class allow you to easily set tracing levels and set parameters for radstate for a particular user, machine, or group of users for troubleshooting purposes. These attributes were intentionally put into their own transient class for this purpose.

To do this, set the _ALWAYS_ Diagnostics Class Connection in the _BASE_INSTANCE_ of the LOCATION Class to DIAGS.&(ZCONFIG.ZHDWCOMP). Then, create an instance in the DIAGS Class with the computer name of the CM agent computer that you want to set the tracing for. If the machine name does not exist in the DIAGS Class, then the settings in the DEFAULT_DIAGS Instance will be used.

Table 20 Attributes of the DIAGS Class

Attribute	Description
COPDNAME	**Friendly Name** Type the friendly name of the instance.

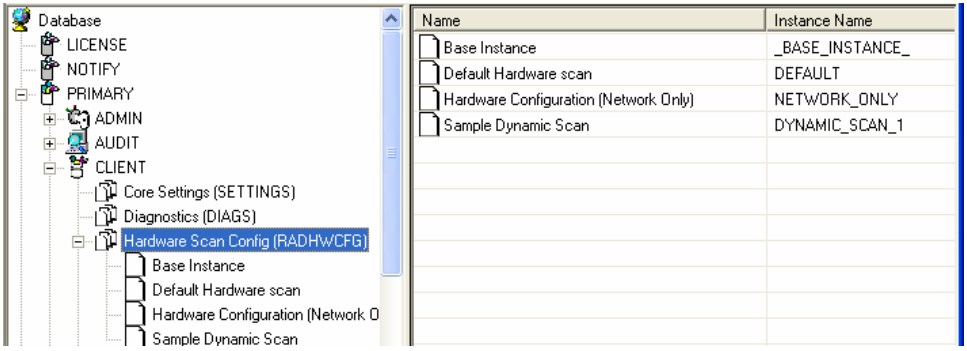
Attribute	Description
RADSTATE	**Command String for radstate** This will run radstate with the parameters specified in this attribute. The base instance of the DIAGS Class is set to VO, which will run radstate in verbose mode, building the ZRSTATE and ZRSTATES objects. See the documentation on radstate for more information. If no parameters are specified, radstate will not run. Radstate must exist in the `IDMSYS` directory. You only need to specify the parameters for radstate, not the radstate executable. Refer to the document on radstate for additional information.
ZTRACE	**Communication Tracing [Y/S/N]** • Specify N to turns off communication buffer tracing. Tracing is off by default. • Specify S to provide summary communication buffer information to the agent log. This includes number of records read and written and the type of records processed. • Specify Y to provide full communication buffer information to the agent log. All data transmitted and received will be echoed to the agent log file Caution: Setting ZTRACE = Y may generate very large logs and severely impact performance of the agent. Do not set this unless instructed to do so by Technical Support.
ZTRACEL	**Trace level (000/040/999)** Specify tracing level. If blank, use existing value. Caution: Setting ZTRACEL to a higher number may generate very large logs and severely impact performance of the agent. Do not set this unless instructed to do so by Technical Support.

Hardware Scan Options (RADHWCFG)

Use instances in the RADHWCFG Class in the CLIENT Domain to specify the type of hardware scans you want performed on the agent device. Hardware scan information is reported in the ZCONFIG object. To implement the hardware scan options, connect an instance of the RADHWCFG Class to an instance in the LOCATION Class.

Client Operations Profiles must be enabled to use the RADHWCFG Class. For testing, you may want to create a RADHWCFG object on the agent device with all the attributes in the RADHWCFG Class. Then, change the attributes to Y or N to see the result in the ZCONFIG object.

Figure 10 Sample instances in the RADHWCFG Class



We provide you with four sample instances in RADHWCFG.

- **Base Instance**
  Create copies of the base instance to create your own hardware scans.
- **Default Hardware Scan**
  This instance scans for the most commonly requested information.
- **Hardware Configuration (Network Only)**
  This instance scans for network information only.
- **Sample Dynamic Scan**
  This instance provides samples using the Dynamic Scan variables.

The table below details each of the possible hardware scans. Examples of the ZCONFIG attributes that may be returned are provided.

Note that the attributes returned depend on what the hardware configuration is. For example, if the agent device has only one printer connected, then only one ZHDWPA0*n* attribute will be reported in ZCONFIG.

Table 21 Attributes in the RADHWCFG Class

Attribute	Description
NAME	**Friendly Name** The friendly name of the instance.
CPU	**CPU [Y/N]** Specify Y to scan for CPU information. ZCONFIG attributes: ZHDWBIOS, ZHDWCOMP, ZHDWCPU, ZHDWCPUN, ZHDWCPUS, ZHDWFPU, ZHDWXPAG, ZHWCPU01, ZHDFPU01
OS	**OS [Y/N]** Specify Y to scan for Operating System information. ZCONFIG attributes: REBOOTD, REBOOTT, WTSSRVR, ZHDWLANG, ZHDWOS, ZHDWOSDB, ZHDWOSOG, ZHDWOSOW, ZHDWSVCP
MEMORY	**Memory [Y/N]** Specify Y to scan for memory information. ZCONFIG attributes: ZHDWMEM, ZHDWMEMF
HDLOCAL	**Local Drives [Y/N]** Specify Y to scan for internal hard drives. ZCONFIG attributes: ZHDWCDDR, ZHDWD00, ZHDW00C, ZHDWD00F, ZHDWD00S, ZHDW00T, ZHDWD01, ZHDW01C, ZHDWDF_A, ZHDWDLST, ZHDWDNUM
HDREMOTE	**Remote Drives [Y/N]** Specify Y to scan for external hard drives. ZCONFIG attributes: ZHDW00, ZHDWD00C, ZHDWD00F, ZHDW00S, ZHDW00T, ZHDWDLST, ZHDWDNUM
NETWORK	**Network [Y/N]** Specify Y to scan for network information. ZCONFIG attributes: GATEWY01, IPADDR01, LADAPT01, NETLOC01, SUBNET01, ZGATEWAY, ZHDWIPAD, ZHDWLANA, ZHDWNET1, ZHDWNNET, ZNETLOC, ZSUBNET

Attribute	Description
PERIPHER	**Peripherals [Y/N]** Specify Y to scan for peripherals such as keyboard and mouse. ZCONFIG attributes: ZHDWKYBD, ZHDWMOUS, ZHDWPPAR, ZHDWPSER, ZHDWVIDO, ZHDWVRES
PRINTER	**Printers [Y/N]** Specify Y to scan for printers. ZCONFIG attributes: ZHDWPA00, ZHDWPA01, ZHDWPPRN
HAL_VER	**HAL Statistics [Y/N]** Specify Y to scan for the HAL (Hardware Abstraction Layer) version. ZCONFIG attributes: HALCOMP, HALDATE, HALFNAME, HALFVER, HALINAME, HALLANG, HALPNAME, HALPVER, HALSIZE.
APP_VER	**Application Version [Y/N]** Specify Y to scan for versions of MSI (ZHDWVMSI) and IE (ZHDWVIE).
WMISCAN	**Use WMI to collect data [Y/N]** Specify Y to perform the scan using WMI (Windows Management Instrumentation).
DSCAN00n	**Dynamic Scan 00n** Specify Y to use the dynamic scan variable.

Dynamic Scanning

In addition to the built in scans, create your own scans using the Dynamic Scan (DSCAN00n) instances. File is the only type of dynamic scan instance supported for the UNIX version of CM. The format for a dynamic scan is: VariableName = Type(Parm1, Parm2, …) where VariableName is the attribute in ZCONFIG where you want the information to be reported, Type is File, and Parm*n* is the query for the information.

Example: File

Dynamic File scanning can return size (SIZE), date stamp (DATE), file version (FVER), product version (PVER), and time (TIME) stamp of a specified file. You may request any combination of these properties. To scan for the file `/opt/temp/test`, create a DSCAN002 similar to:

```
TEST####=FILE(/opt/temp/test;SIZE,DATE,FVER,PVER,TIME)
```

The #### will be replaced by the corresponding file property name. One attribute will be created in the ZCONFIG object for each file property for which you scanned. In this example, five variables will be created based on the information collected on the `/opt/temp/test` file, ZCONFIG.TESTSIZE, ZCONFIG.TESTDATE, ZCONFIG.TESTFVER, ZCONFIG.TESTPVER, and ZCONFIG.TESTTIME.

Setting User Interface Properties (RADUICFG)

Use the RADUICFG Class to specify settings for the CM Application Self-Service Manager User Interface. You must be licensed for the CM Application Self-Service Manager agent. To implement the hardware scan options, connect an instance of the RADUICFG Class to an instance in the LOCATION Class. For a description of the RADUICFG Class and instances, refer to the *CM Application Self-Service Manager Guide*.

You must be licensed for and install the CM Application Self-Service Manager to use this class.

Client Operations Profiles Example

This section provides a simple example of how to configure a Client Operations Profiles. The driving force is to have your agent computers connect with the most appropriate CM Configuration Server. Usually, you will want to assign your agent computers to a CM Configuration Server based on network address.

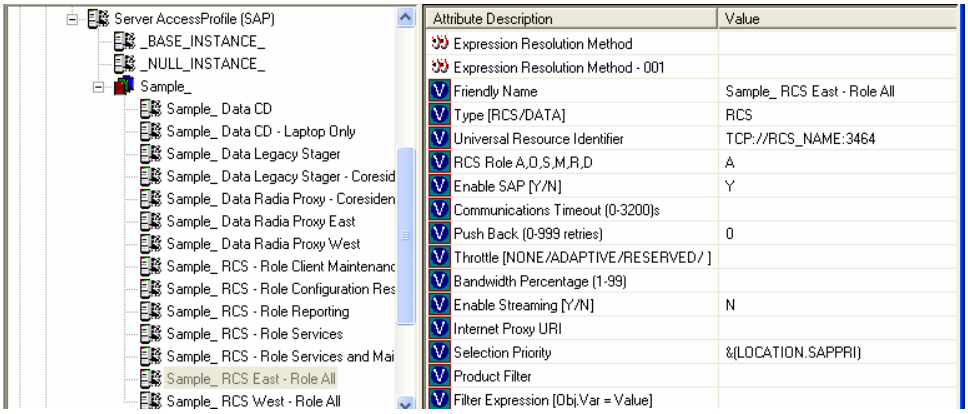
Scenario

Suppose you divide your enterprise into two regions, EAST and WEST. All agent computers in the EAST region are in the 192.111.111.0 network, and

all agent computers in the WEST region are in the 193.111.111.0 network. In addition, suppose you have two CM Configuration Servers, one called RCS_EAST as the primary CM server for the EAST region, and one called RCS_WEST as the primary CM server for the WEST region.

To configure the sample scenario

1 Build two Server Access Profile (SAP) instances, one for RCS_EAST and one for RCS_WEST. The figure below shows a SAMPLE_RCS_EAST.



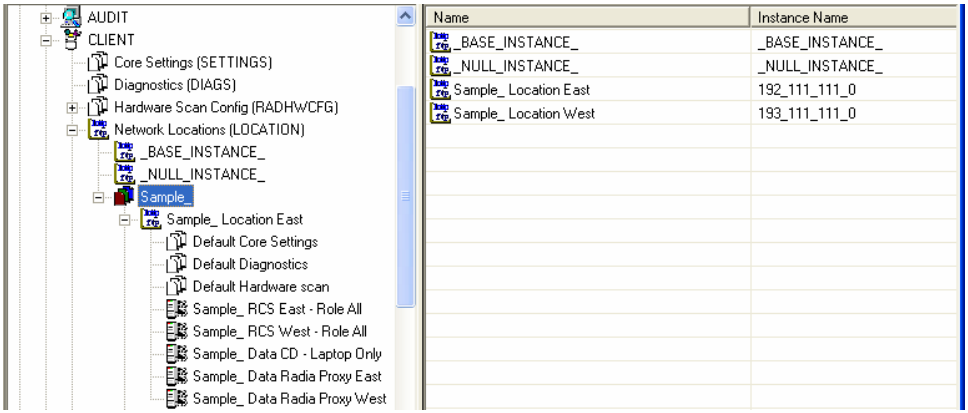
In the Server Access Profile (SAP) Class use the TYPE attribute to specify a server as type DATA or RCS. In this example, we will be configuring only CM Configuration Servers. Therefore, all servers will have SAP.TYPE set to RCS.

For each Server Access Profile instance, you must also identify a role. Again, for simplicity, we will set SAP.ROLE to A for all. This means that the CM Configuration Server can provide agent operations profiles, service resolution, maintenance, data, and reporting.

At a minimum, you will need to specify the Universal Resource Identifier (URI) attribute. Customize other variables as needed.

2 Build two location instances, one for the EAST region, and one for the WEST region.

Create a location instance called 192_111_111_0 with a friendly name of Sample_Location East, and a location instance called 193_111_111_0 with a friendly name of Sample_Location West. See the figure below for an example.



3 Connect the LOCATION instance to the appropriate Server Access Profile (SAP) instance.

— In the LOCATION.Sample_Location East instance define a connection to the SAP.Sample_RCS EAST.

— In the LOCATION.Sample_Location West instance define a connection to the SAP.Sample_RCS WEST.

4 Now, consider what you want to happen in the following cases:

— Suppose you are an agent in the EAST Region and the RCS_EAST is unavailable. Your options are:

– Abort

or

– Go to RCS_WEST as a second choice.

— Suppose you are an agent in the EAST region and the RCS_EAST is busy. In other words, the task limit defined in your CM Configuration Server settings file has been reached. Your options are:

– Continue to retry the RCS_EAST until a connection is available

or

– Go to RCS_WEST as a second choice.

Once you understand the concepts involved, and feel comfortable with this process, you can begin to add other components to the Client Operations Profile. After you finish with TYPE=RCS, configure your servers with TYPE=DATA. In addition, you can identify particular servers of TYPE=RCS to use different ROLES.

Summary

- Use Client Operations Profiles to provide redundancy in your environment.
- Select which servers will perform which roles.
- You can assign agent computers to specific servers based on network location or any other criteria.
- You must enable Client Operations Profiles in the CM Configuration Server DB and on the agent computer.

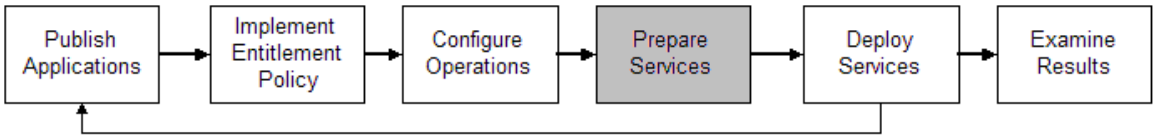
8 Preparing Services

At the end of this chapter, you will:

- Be aware of service options.
- Know how to create a service from a promoted package.
- Know how to restart the agent computer.
- Know how to install services under the system account.
- Be familiar with preparing versioned applications.

This guide covers the *suggested* implementation for the CM Application Manager. Although you will tailor this strategy to meet your organization's needs, we recommend that you review this guide for a comprehensive understanding of the CM Application Manager. This chapter covers preparing services.

Figure 11 Tasks completed in this guide



Restarting the Agent computer

You may need to restart an agent computer based on an application event. To do this, specify a reboot type and reboot modifiers in the ZSERVICE.REBOOT attribute. The modifiers allow you to:

- set the type of warning message
- handle a reboot with either a machine or user connect
- and cause an immediate restart after the application event.

If the hreboot parameter is missing from the radskman command line, the parameter defaults to Y to handle service reboot requests. If you set hreboot to p, the agent computer will *power down*, regardless of whether or not there is a service requiring a reboot.

First, specify the application event that needs the reboot. Table 22 on page 149 lists the codes for all possible application events. Set the application event code to a reboot type and any reboot modifier that you need to use. The sections below describe each type of reboot and all reboot modifiers.

If you need an application to immediately perform a hard reboot with no warning messages on application installation and repair, set the ZSERVICE.REBOOT variable to AI=HQI, AR=HQI.

The parameters for the reboot attribute are not case-sensitive.

Table 22 Reboot Events and Codes

Application Events	Code	Description
Install	AI	Use AI to specify a reboot behavior for application installations. The default is no reboot.
Deinstall	AD	Use AD to specify a reboot behavior for application removals. The default is no reboot.
Locked File	AL	Use AL to specify a reboot behavior when a locked file is encountered. The default behavior when a locked file is encountered is to perform a Hard reboot with just an OK button (HY).
Update	AU	Use AU to specify a reboot behavior for application updates. The default is no reboot.
Repair	AR	Use AR to specify a reboot behavior for application repairs. The default is no reboot.
Version Activation	VA	Use VA to specify a reboot behavior for application version activations. The default is no reboot.

Reboot Types

After deciding which application events need a computer reboot, you will need to choose the type of reboot. CM sends a message to the operating system that the computer needs to reboot. There are three types of reboot.

- **Hard Reboot (H)**
  All applications are shut down regardless of whether there are open, unsaved files or not. The subscriber will not be prompted to save open, modified files.
- **Soft Reboot (S)**
  Users are prompted to save their data if applications have open, unsaved files. If applications have unsaved data, the reboot will wait for the user to respond to the application's request for the user to save his data.
- **No Reboot (N) (default reboot type)**
  The computer will not restart after completing the specified application event. This is the default reboot type for all application events except a Locked File Event (AL). If you specify AL=N, then the agent computer

will not perform a hard reboot with an OK and Cancel button when a locked file is encountered. If no restart type is specified for an application event, no restart will occur.

Reboot Modifier: Type of Warning Message

You can specify the type of warning message you want to send to the subscriber before the restart occurs. If you specify a type of reboot, but do not specify a type of warning message, the default warning message for that type will be displayed. There are three types of warning messages. Warning messages are displayed automatically for the CM Application Self-service Manager and for CM Application Manager used with the CM System Tray. If you do not want to show a warning message, specify ask=N in a radskman command line.

- **Quiet (Q)**
  No reboot panel will be displayed.
- **OK Button (A)**
  A warning message will display with an OK button only. Click **OK** to initiate the reboot. The user will not be able to cancel the restart.
- **OK and Cancel Button (Y)**
  Click **OK** to initiate reboot. If the subscriber clicks **Cancel**, the reboot will be aborted.

You can specify a timeout value for the Warning Message box by adding the RTIMEOUT value to the radskman command line. Set RTIMEOUT to the number of seconds you want the CM agent to wait before continuing with the reboot process.

Reboot Modifier: Immediate Restart

You can modify each type of reboot by adding I for Immediate. Use Immediate when you want the computer to restart immediately after resolving the current service. CM will resolve the rest of the subscriber's services after the computer restarts. If you specify I, but do not specify H or S as the type of reboot, a hard reboot will be performed.

Specifying Multiple Reboot Events

If you have two services that require a reboot event on the same agent connect, the most restrictive reboot type and reboot panel will be used. The least restrictive reboot type is No Reboot (N), followed by Soft Reboot (S), and the most restrictive is Hard Reboot (H). The least restrictive reboot warning message supplies both an OK and Cancel button (Y), followed by an OK button only (A), and the most restrictive is completely quiet (Q).

Suppose a subscriber is assigned an application that needs a soft reboot with just an OK button on installation, AI=SA. The subscriber is also assigned a second application that needs a hard reboot that displays both an OK and Cancel button, AI=HY. After all of the subscriber's application events are completed, a Hard Reboot (H) with only an OK button displayed (A) will be performed.

Preparing Versioned Applications

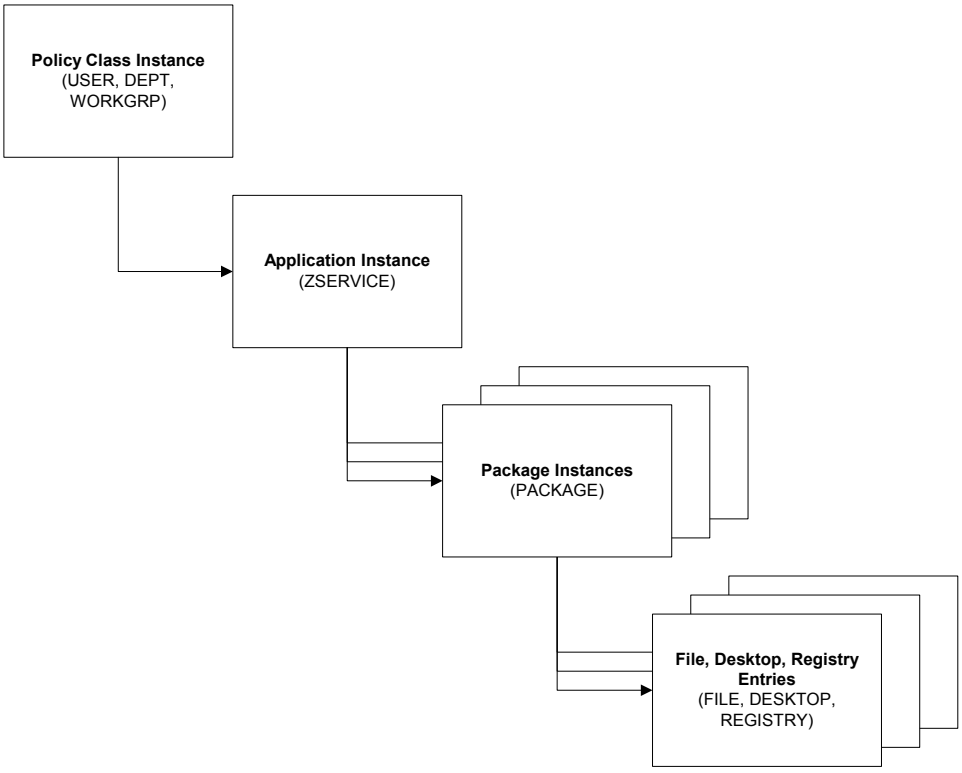
Normally, when you deploy an application to an agent computer, it is activated immediately. This is the case when you use CM Scheduler or CM Notify. However, you can use Version Groups to roll out a new version of an application to subscribers, and activate it upon delivery or at a pre-determined time. If the installation of the new version fails, CM will automatically roll back to the previous version. If problems occur in the new version after installation, you can deactivate the new version and roll back to the previous version for some, or all, subscribers.

After versioning is configured, the compressed files are stored on the agent computer, and the versioning action takes place. The roll forward/roll backward activity can be entirely local, not requiring any data to be transferred at the version change time. It can also be configured to be partially local, with a minimum of data transmitted.

Versioned vs. Non-Versioned Applications

Versioned and non-versioned applications adhere to different connection models within the CM Configuration Server DB. For non-versioned applications, one application instance connects to one or more package instances.

Figure 12 Model for non-versioned deployments

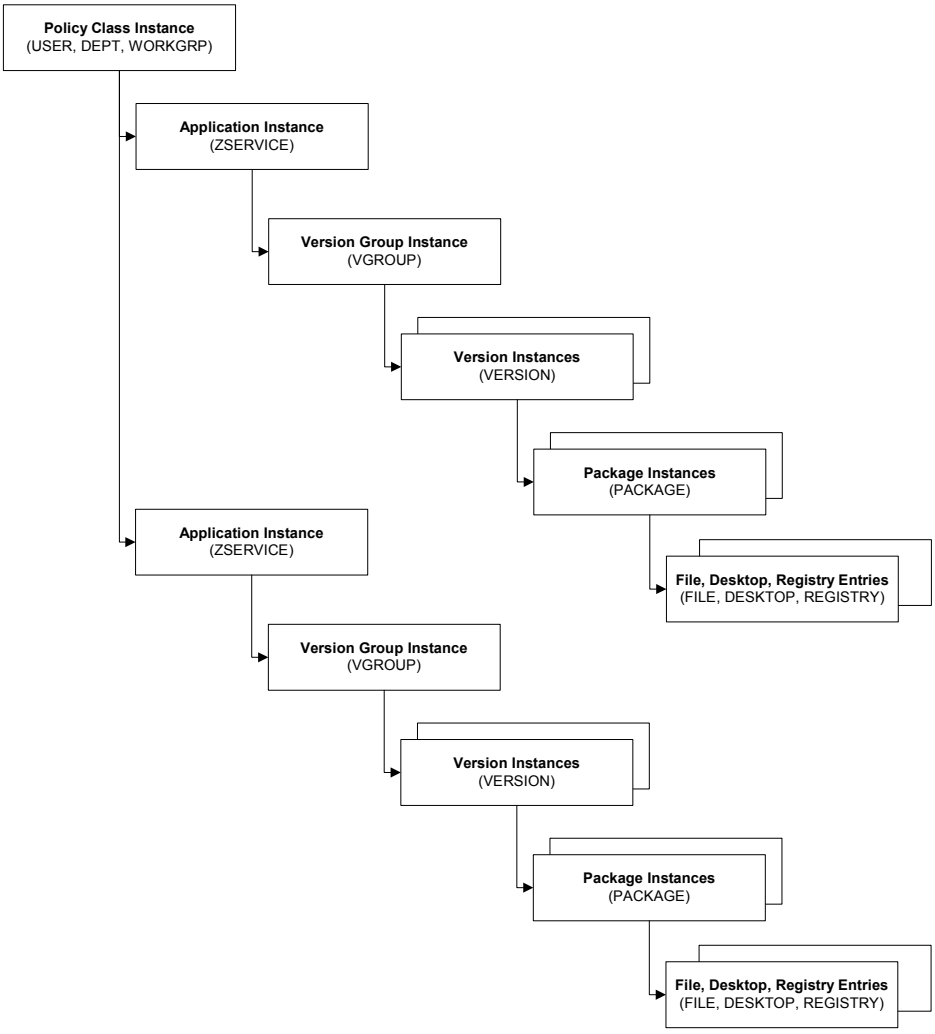


Versioned applications adhere to a different connection model than non-versioned applications. For versioned applications, an Application instance (ZSERVICE) connects to a single Version Group (VGROUP) instance.

If you want to use multiple Version Groups, you must create one Service for each Version Group.

The Version Group instance connects to one or more Version instances that connect to one or more package instances. A Version instance (which represents one version of a software application) contains one CM package. Each CM package is represented in the CM Configuration Server DB by an instance of the PACKAGE Class.

Figure 13 Model for versioned deployment



To prepare versioned applications

1 Use the CM Admin Publisher to package the application.

2 Right-click the **ZSERVICE** Class and select **New Instance**.

3 In the Create Instance dialog box, type a display name and an instance name.

4 Click **OK**.

See The Version Group Editor below to finish creating the Version Group.

The Version Group Editor

Use the Version Group Editor, in the CM Admin CSDB Editor, to create, edit, or delete instances for each version of an application, as well as manage the deployment of a **version group**. A version group contains all of the versions of an application.

To manage a versioned application, create an instance in the Version Group class, which represents the set of versions for the application. Then, use the Version Group Editor to create instances for each version of the application.

Creating a Version Group

In the following example, we will use the CM Admin CSDB Editor to create a new instance in the Version Group (VGROUP) class.

To create a Version Group

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

The CM Admin CSDB Editor Security Information dialog box opens.

The User ID, as shipped from HP, is RAD_MAST. No password is necessary. This might have changed during installation. You can also change this by selecting the **Change Password** check box and typing the new password in the New Password and Verify New Password text boxes.

2 If necessary, type a User ID and Password, and then click **OK**.

The CM Admin CSDB Editor window opens.

3 Double-click **PRIMARY**.

4 Double-click **SOFTWARE**.

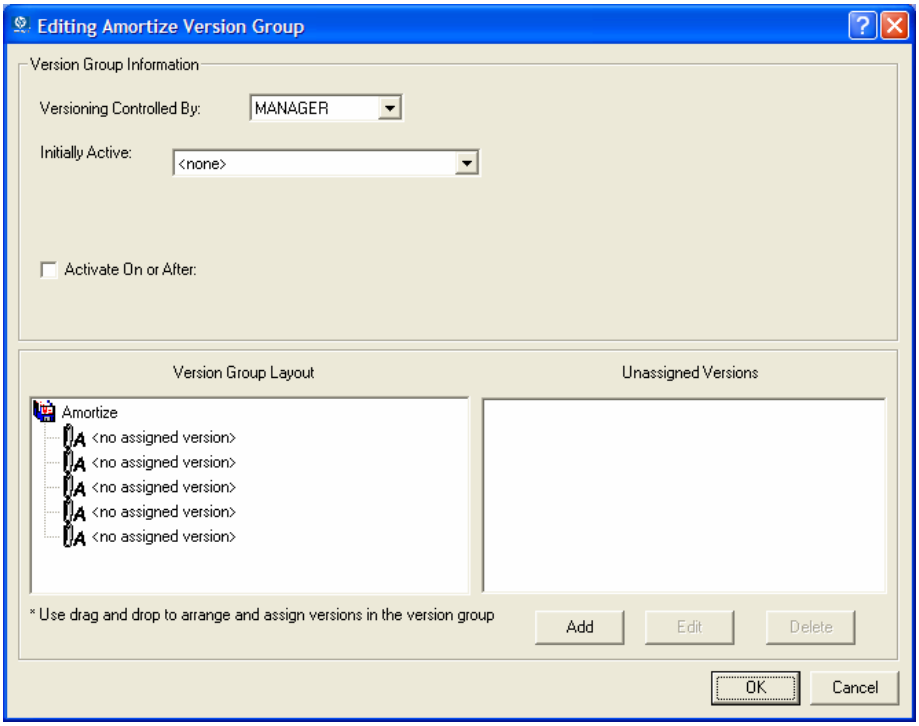
5 Right-click **Version Group (VGROUP)**.

6 Select **New Instance**. The Create Instance dialog box opens.

7 Type a name for the Version Group in the text field in the Create Instance dialog box, such as Amortize.

8 Click **OK**.

The Editing Version Group dialog window opens.



Creating a Version Instance

Now that you have created a Version Group (VGROUP) instance, you will learn how to create an instance for each version of your application.

To create a version instance

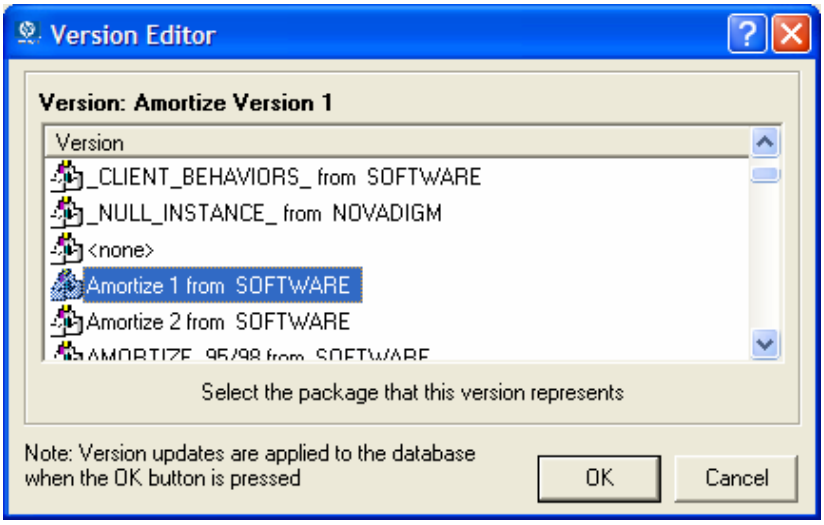
1 In the Version Group Editor, click **Add**.

The Create Version dialog box opens.

2 Type a suffix that identifies the version. For example, type **`Version 1`**. The Version instance will be named Amortize Version 1.

3 Click **OK**.

The Version Editor dialog box opens.

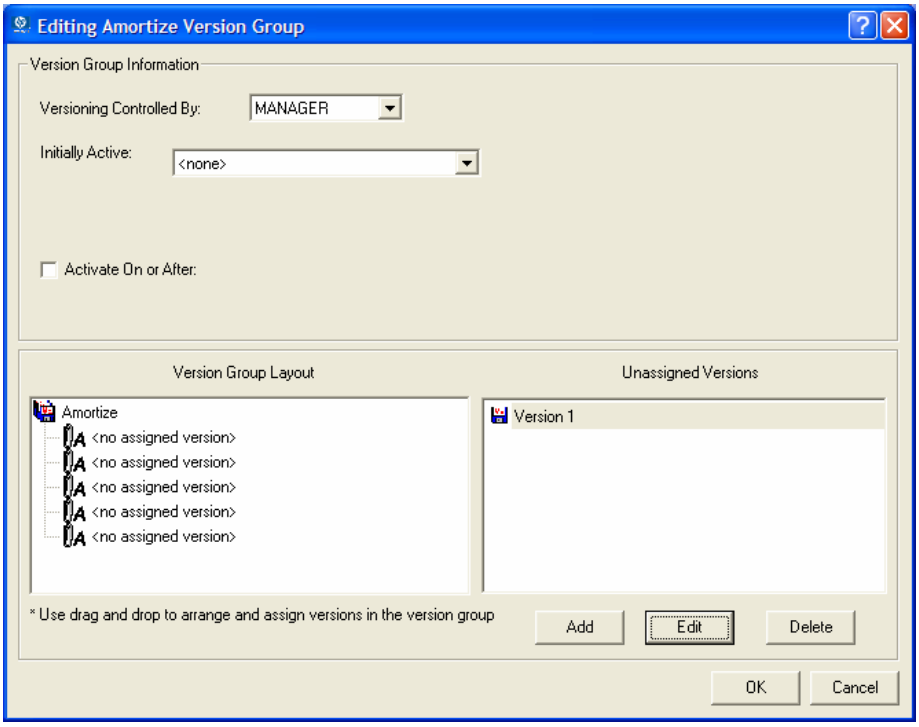


The Version Editor dialog box contains a list of Application Package (PACKAGE) instances stored in the CM Configuration Server DB. Use this dialog box to connect the new Version (VERSION) instance to an Application Package (PACKAGE) instance. There is a one-to-one correspondence between these two instances.

4 Click the appropriate Application Package (PACKAGE) instance, such as Amortize1.

5 Click **OK**.

The Version instance appears in the Unassigned Version list.



Add a Version instance for each version of the application that will be available to subscribers through this Version Group.

To delete a version instance

1. In the Version Group Layout list, select the version that you want to delete.
2. Click **Delete**.

   The instance for the version appears in the Unassigned Versions list. The icon is in a dimmed state, ready for deletion.

   To restore the instance, click the instance in the Unassigned Versions list, and then click **Un-delete**.

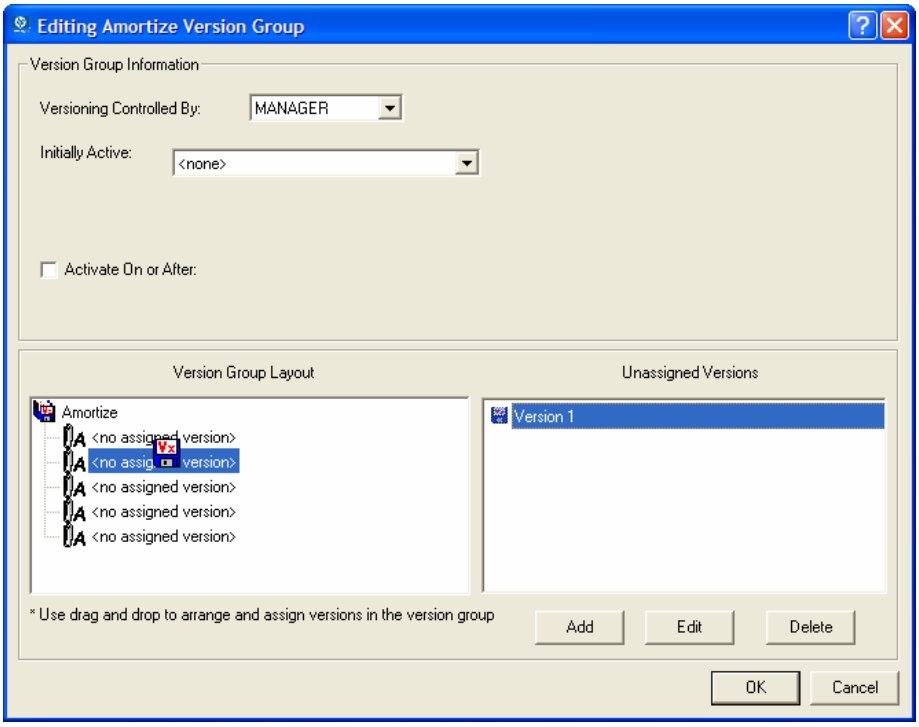
   The version instance will not be deleted until you close the Version Group Editor.
3. Click **OK** to close the Version Group Editor.

Assigning Version Instances to the Version Group

After creating your Version instances, you must assign them to the Version Group.

To assign Version instances to the Version Group

- In the Unassigned Versions list, click a Version instance and drag it over a connection labeled <no assigned version> in the Version Group Layout list.



Assign each of the Version instances that you created to the Version Group.

To remove a Version instance assignment

- In the Version Group Layout, click a Version instance and drag it to the Unassigned Versions area. Then, release the mouse button.

Preparing a Version Group for Deployment

Now that you have created the Version instances and assigned them to your Version Group, you are ready to specify how you would like to deploy the

versions. Use the Version Group Information area in the Version Group Editor to define the deployment of the versions.

To prepare a Version Group for deployment

1 In the Versioning Controlled By drop-down list, select **Manager** or **Client**.

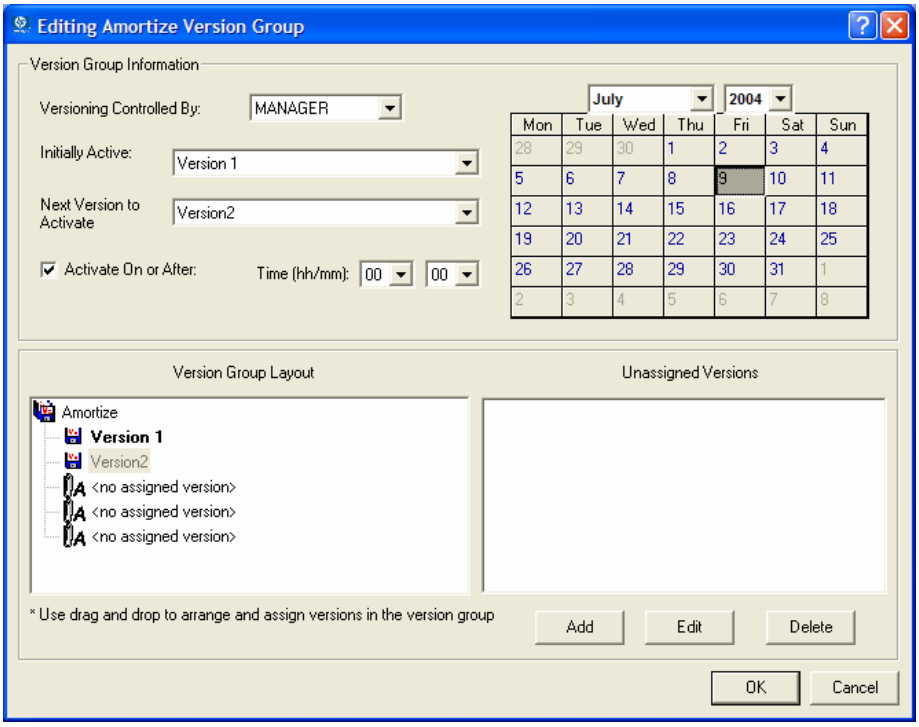
- Select **Manager** if you want to control the version to be deployed.
- Select **Client** if you want the subscriber to control the version to be deployed. *This is used only with the CM Application Self-service Manager agent.*

> You can schedule deployments of versions *only* if the CM Configuration Server controls the versions. Therefore, to schedule deployments, you must select **MANAGER** from the Versioning Controlled By drop-down list.

2 In the Initially Active drop-down list, select the version that you want to activate on the agent computer the next time the subscriber connects to the CM Configuration Server. You can select from the versions that appear in the Version Group Layout list. The selected version is bolded in the Version Group Layout list, as shown in the next figure.

3 Select the **Activate on or After** check box to access additional controls used to delay activation of a version until a specific date and time.

> If you select **MANAGER** in the Versioning Controlled By drop-down list, you *must* select the **Activate On or After** check box so that the CM Configuration Server knows when to activate the next version.



4 In the Next Version to Activate drop-down list, select the version of the application that you want to activate after the initial version.

5 In the Time (hh/mm) drop-down list, select when you want the version to be activated.

6 Use the Calendar controls to set the date of deployment for the next version.

If you use the Time and Calendar controls to schedule the deployment of a version, consider the following:

- You can schedule deployments of versions *only* if the CM Configuration Server controls the versions. Therefore, to schedule deployments, you must select **MANAGER** from the Versioning Controlled By drop-down list.
- If you selected **MANAGER** in the Versioning Controlled By drop-down list, you must select the **Activate On or After** check box to let the CM Configuration Server know when to activate the next version.
- If you delete a VGROUP instance, the associated timer instance will be deleted.

7 Click **OK** to save the information in the Version Group Editor.

8 Click **Yes** to confirm your changes.

The Version Group instance appears in the Version Groups (VGROUP) Class. If you scheduled the next version to activate, CM creates a timer instance in the Scheduling (TIMER) Class and automatically connects the timer to the Version Group.

To connect the Version Group to the Service

1 Navigate to PRIMARY.SOFTWARE.ZSERVICE.

2 Right-click the appropriate service, in this example Amortize.

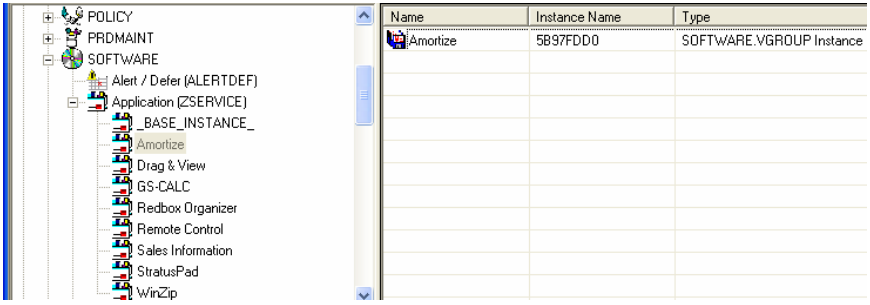
3 Select **Show Connections**.

The SOFTWARE.ZSERVICE Connections dialog box opens.

4 Click **Version Groups (VGROUP)**.

5 Click **OK**.

The Version Group instances appear in the list view of the CM Admin CSDB Editor.



6 Click **Amortize** in the list view and drag it to the appropriate Application (ZSERVICE) instance (in this example, Amortize). When your cursor changes to a paper clip, release the mouse button.

The Select Connection Attribute dialog box opens.

7 Click **Copy**.

8 Click **Yes** to confirm that you want to connect the Amortize Version Group to the Amortize service.

9 Click **OK** when you receive a confirmation message.

Be sure to connect the ZSERVICE to the POLICY instance for the subscribers to whom you want to deploy this.

The next time the agent computer connects to the CM Configuration Server, the initial version of the application is activated, and the compressed files for the next version will be stored on the agent computer.

Editing a Version Group

After you create a version group and its instances, you may want to return to the Version Group Editor to make changes.

To edit a Version Group

1. Navigate to the Version Group instance, located in PRIMARY.SOFTWARE.VGROUP.
2. Right-click the appropriate Version Group instance.
3. Click **Version Group Editor**.

   The Version Group Editor opens. Modify the Version Group as necessary.
4. Click **OK** to save your changes.

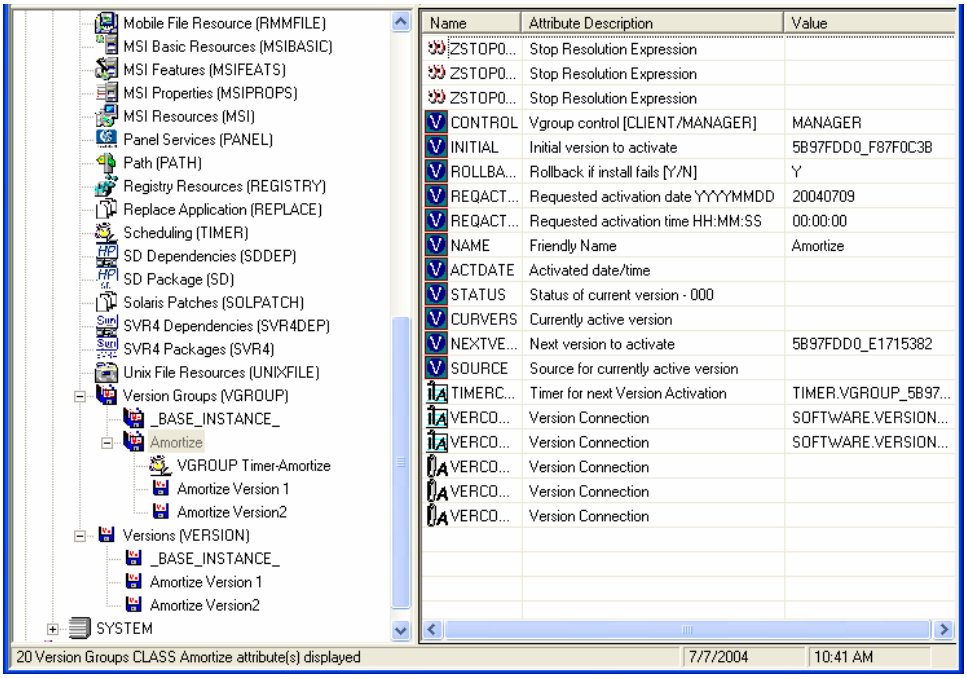
   or

   Click **Cancel** to close the Version Group Editor without saving your changes.

The Version Group (VGROUP) Class

Each instance of the Version Group (VGROUP) Class defines a set of versions for an application. This class contains connections to the Versions (VERSION) Class, created using the Version Group Editor in the CM Admin CSDB Editor.

Figure 14 Version Groups (VGROUP) Class Instance



The following table describes each of the attributes in the Version Group (VGROUP) Class.

Table 23 Version Group (VGROUP) Class attributes

Attribute	Description
ZSTOP00*n*	Expressions evaluating to true in ZSTOP attributes cause resolution of the instance to be skipped. If left blank, the instance is not skipped, and resolution continues. This is useful for assigning a version to a specific set of users. Use the CM Admin CSDB Editor to set this attribute.
CONTROL	Indicates whether the CM administrator (MANAGER) or the subscriber (CLIENT) controls which version to activate on the agent computer. Use the Versioning Controlled By drop-down list in the Version Group Editor to set this option. Note: The CM Application Manager supports CM administrator-controlled version activation, but does not support subscriber-controlled activation.

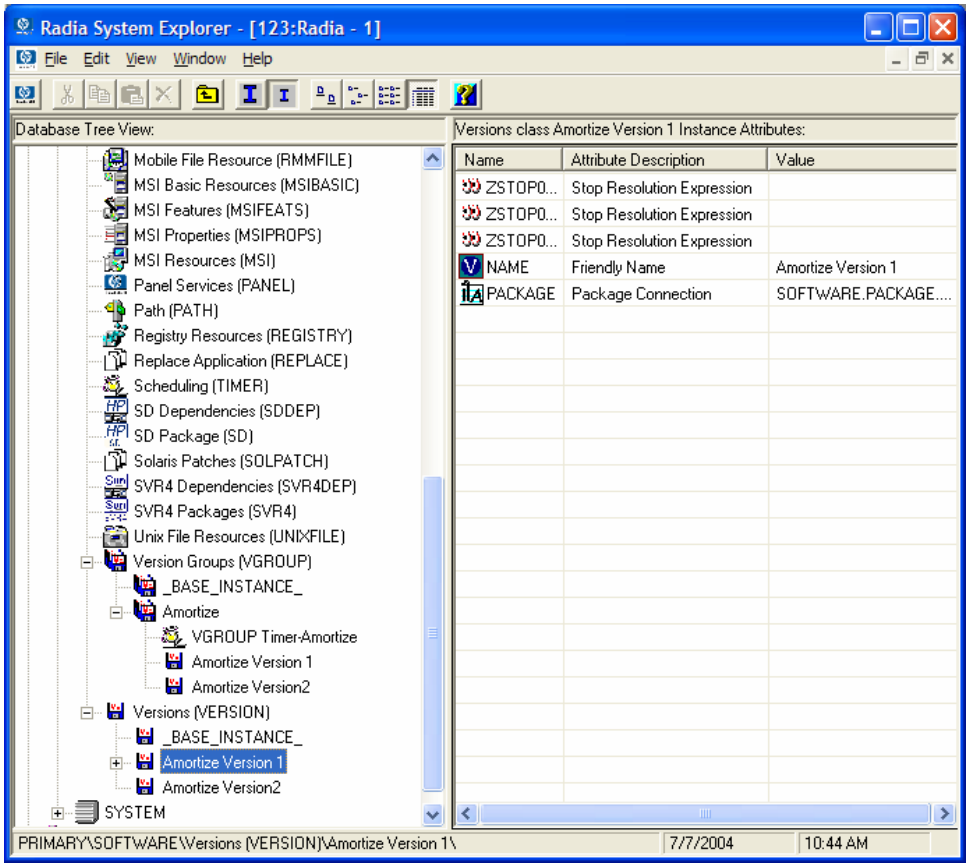
Attribute	Description
INITIAL	Indicates which version to activate on the agent computer. Use the Initially Active drop-down list in the Version Group Editor to set this option.
ROLLBACK	Indicates whether to automatically roll back to the previously activated version when deployment of a new version fails. A new version may fail to deploy because of lack of sufficient disk space on the agent computer, improper packaging, or failure of a method to complete successfully. By default, ROLLBACK = y.
REQACTDT	The earliest date on which a version in this version group will be activated on any agent computer. If this attribute is blank, the version identified by the INITIAL attribute will be activated at the end of the agent connect that causes the version to be transferred to the agent computer. Use the calendar controls in the Version Group Editor to set REQACTDT.
REQACTTM	The earliest time, on the date specified by the REQACTDT attribute, after which a version in the version group will be activated on any agent computer. The version identified by the INITIAL attribute will be activated during the next agent connect. Use the Time (hh/mm) drop-down lists in the Version Group Editor to set REQACTTM.
NAME	The friendly name for the VGROUP instance. This is set when you create the instance using the CM Admin CSDB Editor.
ACTDATE	Data maintained by the CM agent in the VGROUP object on the agent computer. Do not enter a value. This is set by the CM agent.
STATUS	Data maintained by the CM agent in the VGROUP object on the agent computer. Do not enter a value. This is set by the CM agent.
CURVERS	Data maintained by the CM agent in the VGROUP object on the agent computer. Do not enter a value. This is set by the CM agent.

Attribute	Description
NEXTVERS	Data maintained by the CM agent in the VGROUP object on the agent computer. Do not enter a value. This is set by the CM agent.
SOURCE	Data maintained by the CM agent in the VGROUP object on the agent computer. Do not enter a value. This is set by the CM agent.
TIMERCON	If you specify a "next version to activate," the CM Admin CSDB Editor automatically creates a timer and stores the connection to that timer in this attribute.
VERCON0*n*	Connects to each version in the version group. Each VERCON0*n* attribute contains a connection to one instance of the VERSION Class. This is set when you assign a version to the version group in the Version Group Editor.

The Versions (Version) Class

Each instance of the version class defines one *version* of an application to be deployed and managed by CM. Use the Version Group Editor to create Versions class instances and assign them to a Version Group.

Figure 15 Versions (VERSION) Class instance



The following table describes each of the attributes in the Versions (VERSION) Class.

Table 24 Versions (VERSION) Class attributes

Attribute	Description
ZSTOP00*n*	Expressions evaluating to true in ZSTOP attributes cause resolution of the instance to be skipped. If left blank, the instance is not skipped, and resolution continues. This is useful for assigning a version to a specific set of users. Use the CM Admin CSDB Editor to set this attribute.
NAME	The friendly name for the VERSION instance. This is set when you create the instance using the Version Group Editor.

Attribute	Description
PACKAGE	Connects to a PACKAGE Class instance, which represents the packaged software for this version.

Application (ZSERVICE) Attributes

This section describes the attributes that you will see if you open an Application (ZSERVICE) instance in the CM Admin CSDB Editor. Many of the values for these attributes are set when using the CM Administrator, such as the CM Admin Publisher or the New Application Wizard in the CM Admin CSDB Editor. You can also use the CM Admin CSDB Editor to modify the values of these attributes in the SOFTWARE.ZSERVICE Class.

You may notice that some attributes do not have values, or their values are not displayed in the CM Admin CSDB Editor. The CM agent uses these attributes. For example, an attribute such as INSTDATE is used to record the date the service was installed on the agent computer. The value for this attribute is stored in the PROFILE File for the agent computer in the CM Configuration Server DB.

Table 25 Modifiable SOFTWARE.ZSERVICE attributes

Attribute	Description
ZSTOPnnn	**Expression Resolution Method** Stops resolution if the expression evaluates to TRUE. **Example**: WORDPOS(EDMGETV(ZMASTER,ZOS),'WINXP WIN2K NT')=0 This example expression will stop resolution on the instance if the agent computer's operating system if the operating system is *not* Windows XP, Windows 2000, or Windows NT. In other words, the application will not be installed unless the agent computer is running Windows XP, Windows 2000 or Windows NT.
ZSVCNAME	**Service Name/Description** Name of the service used for display in the CM Application Self-service Manager user interface. Value is set initially in the Short Description field in the New Application Wizard.

Attribute	Description
ZSVCTTYP	**Application Target Type [A/S]** Indicates which CM agent this application was packaged for, CM Application Manager or CM Application Self-service Manager. Value is set initially in the New Application Wizard. Possible values are A for CM Application Manager and S for CM Application Self-service Manager.
ZSVCMO	**Mandatory or Optional Service [M/O]** Designates a service as mandatory or optional. This value is set initially based on the setting for the application target type (ZSVCTTYP) in the New Application Wizard. Usually, when using the CM Application Manager, services are marked as mandatory, M. When using the CM Application Self-service Manager, services are usually marked as optional, O. If you are using CM Application Manager *and* CM Application Self-service Manager, you could also specify mandatory and then optional, MO, or optional then mandatory, OM. The first character indicates how the application should be handled before installation. The second character indicates how the application should be handled after installation. For example, suppose you want a CM Application Self-service Manager subscriber to have the option of installing the application, but, once installed, want the maintenance or removal of the application to be mandatory, set ZSVCMO to be OM. Note: If you may need to edit the ZSERVICE Class template, to allow you to set ZSVCMO to OM. Refer to the *CM Admin CSDB Guide* for more information on editing a class template. To process mandatory applications using CM Application Self-service Manager, add "`enterprisemanagement=auto`" to the `args.xml` file.
ZSVCPRI	**Service Create Ordering [01-99]** Set the priority level for the service. Services are created based their priority. The lower the number, the higher the service's priority. A service with ZSVCPRI set to 01 would have the highest priority while a service set to 99 would have the lowest priority.

Attribute	Description
ALWAYS	**Contains** Any method that you specify for this attribute is unconditionally executed when this instance is resolved. Example: A valid method name such as ZSYSTEM.ZMETHOD.PUTHIST_ZERROR.
ZCREATE	**Service Installation Method** Method that runs when the service is installed. For example, a command to start a service that was stopped to install files.
ZINIT	**Service Initialization Method** Method that runs when the service is initialized. For example, a command to stop a service before installing files that the service might lock.
ZDELETE	**Service Delete Method** Method to run when the service is deleted.
ZUPDATE	**Service Update Method** Method to run when the service is updated.
ZVERIFY	**Service Verify Method** Method to run when the service is verified.
ZREPAIR	**Service Repair Method** Method to run when the service is repaired.
PUBDATE	**Published Date of Service** *Reserved for future use.*
UPDDDATE	**Upgrade Date (Programmatic)** *Reserved for future use.*
AUTHOR	**Author Name** Name of the author of the service that appears in the extended information area in the CM Application Self-service Manager user interface. Value is set initially in the Author field in the New Application Wizard.

Attribute	Description
DESCRIPT	**Application Description** Description of the service that appears in the properties for the service in the Service List. Value is set initially in the Long Description field in the New Application Wizard.
VENDOR	**Vendor Name** Name of the vendor of the service that appears in the CM Application Self-service Manager user interface. Value is set initially in the Vendor field in the New Application Wizard.
URL	**WEB URL Name** Address of a web page where the subscriber can find additional information about the service. This appears in the properties for the service in the CM Application Self-service Manager user interface. Value is set initially in the Web URL field in the New Application Wizard.
CATGROUP	**Catalog Group Name** Use CATGROUP to group a set of applications into a group. You can display applications based on their group in the CM Application Self-service Manager user interface.
PRICE	**Price** Type in the price of an application to be displayed to subscribers in the extended information area in the CM Application Self-service Manager user interface.
SCHEDOK	**Update Schedule Locally [Y/N]** *For CM Application Self-service Manager only.* Specify **`Y`** to allow the subscriber to change the update schedule locally. Specify **`N`** to maintain control on the CM Configuration Server.
VERSION	**Version Description** Version of the software. This appears in the properties for the service in CM Application Self-service Manager user interface. The value is set initially in the Version field in the New Application Wizard.

Attribute	Description
NAME	**Friendly Name** This name appears in the properties for the service in the CM Application Self-service Manager user interface. The value is set initially in the Short Description field in the New Application Wizard.
OWNER	**Application Contact** *Reserved for future use.*
RUNDLG	**Dialog Processing [Y/N]** Specifies whether to enable processing of instances in the DIALOG Class during the installation of the service. Specify Y for Yes and N = No. Default: N
REBOOT	**Install/Update/Delete/Version Chang** Used to restart the agent computer based on application event. Specify your action by equating an application event to a reboot type, panel, or connect. Event to restart on: AI = Install AD = Deinstall AU = Update AR = Repair AV = Verify Type of reboot: S = Soft Boot (Default of type Y panel.) H = Hard Boot (Default of type A panel.) N = None Type of panel: Q = No panel. A = OK button only. Y = OK and Cancel button. Type of connect: None specified: Reboot on Machine connect (context = m). U = reboot on user connect only (context = u).

Attribute	Description
	MU = reboot when both machine and user parts of the service have been installed. Example: AI=S performs a soft boot on application installation.
EVENTS	**Events to Report** Indicates which events to report on. Specify your event by equating an application event to an event type. AI:Application Install AD: Application Deinstall AU: Application Update AR: Application Repair AV: Application Verify VA: Version Activation VD: Version Deactivation What to report on S: Success F: Failure B: Both Success and Failure N: None Default: AI=B,AD=B,AU=F,AR=N,VA=F,VD=F
ERTYPE	**Event Reporting Method [O/E/X]** Set this attribute to send an APPEVENT object. Currently, this supports object (O) format only. Default: O
ADAPTIVE	**Auto Adaptability [Y/N]** Indicates whether the installed package is dependent on client settings that must be monitored periodically, such as plug and play devices. If the settings change, the client must reconnect to the CM Configuration Server to get new or different components. Specify Y for Yes and N for No.

Attribute	Description
LREPAIR	**Local Repair [Y/N]** Enables local repair of broken applications. If an application is broken because of missing files, the files (stored locally) can be used to repair the application. Specify Y for Yes and N for No. Default: N
REMOVAL	**Un-Managed Behavior [A/D/U]** Controls how the application is managed when a service is removed. • Set REMOVAL to A (Abandon) to delete the service's objects on the client, but leave the application components. The service will no longer be managed by CM. • Set REMOVAL to D (Delete) to delete the service's objects and components. The service will still be managed by CM. • Set REMOVAL to U (Unmanage) to stop management of the service by CM. Neither the objects nor the components are deleted. This applies only to optional applications (ZVSCMO set to O) that are removed based on entitlement policy. If a subscriber removes an optional application, the service's objects are always removed no matter what REMOVAL is set to. Default: D
RECONFIG	**Reconfiguration Enabled [Y/N]** Indicates whether an application can be relocated after it has been installed. Specify Y for Yes and N for No. For example, this allows you to move an application that was installed on the C drive to the D drive without removing and re-installing the application.
ZSVCCAT	**Service Visible in Catalog [Y/N]** Specifies whether the service is visible in the CM Application Self-service Manager Catalog. For optional applications, the default is Y. For mandatory applications, the default is N. Specify Y for Yes and N for No if you want to override these defaults.

Attribute	Description
UIOPTION	**Progress Indicator [NONE/FULL/INFO]** Controls whether the service status window appears. Possible values are: **NONE** = No interface appears. **FULL** = Interface appears and Cancel button is available. **INFO** = Interface appears with no option to cancel.
CACHE	**App Element Caching [Y/N]** Enables element caching. Specify Y for Yes and N for No. Default: N
CACHELOC	**CACHE Location On Client** *For Windows Installer applications only.* Location of the folder on the agent computer that is used to cache the compressed application files needed for the product. CM support for Windows Installer tags the PRODGUID value to this value to create the folder. For example, If `CACHELOC=C:\progra~1\HP` and `PRODGUID = 12345_XXXX`, then the cache folder would be: `c:\progra~1\HP\12345_XXXX\cache`. Note: The folder `\cache` is automatically appended to PRODGUID. If you are not deploying a Windows Installer-enabled application, the files will be cached in `IDMDATA`. Default: _UNDEF_
CACHELIM	**Percnt Disk Limit For Cache** *For Windows Installer applications only.* Cache limit, which is defined as the percentage of used drive space. Type a number between 000 and 100. If the percentage of used space is greater than the cache limit, then all of the cached files for the product are removed and the cache folder is deleted. This is checked after every file is cached on the disk.

Attribute	Description
ZDISCONN	**Disconnect on Install [Y/N]** Allows the agent to disconnect from the CM Configuration Server if there is an open session with the CM Configuration Server. • Specify Y to disconnect the client from the CM Configuration Server. • Specify N to keep the client connected to the client from the CM Configuration Server. Default: N
ZSYSACCT	**Install under System Account[Y/N]** Specifies whether to install the service under the system account or the user's account. • Specify Y to install the application using the system rights. • Specify N to install the application using the rights of the logged on user. Default: N
MCELIGBL	**Service Multicast Eligible[Y/N]** Indicates if the application is eligible for multi-casting. Specify Y for Yes and N for No. Default: Y
RSTRSIZE	**Download restart threshold (bytes)** Use the RSTRSIZE attribute in the appropriate ZSERVICE Class instance to control which files are enabled for check point restart based on the amount of data being downloaded (in bytes).
ZSVCMODE	**Application Context [M/U/MU/EMU]** • Set ZSVCMODE to M if the service has only machine components. This service will be ignored if context is set to u on the radskman command line. • Set ZSVCMODE to U if the service has only user components. This service will be installed if context is set to u or is left blank on the radskman command line. You may want to set ZSVCMODE to u if the application consists only of user registry changes or user desktop shortcuts. • Set ZSVCMODE to MU if the service has both

Attribute	Description
	machine and user components. The user connect will verify that the machine components have been installed before installing the user components. You will need to run two radskman connects, one with context set to m and one with context set to u. • Set ZSVCMODE to EMU if the agent connect is being made in the user context, but the machine side of the application has not yet been installed, this will force the machine connect. After the machine connect completes successfully, the user connect is initiated to install the user components. Use this for optional applications that the user controls through the CM Application Self-service Manager. • Leave ZSVCMODE blank to treat the service as single mode that can be installed independently by the machine or the user. In other words, install the entire service ignoring the component's ZCONTEXT.

Reporting Attributes in ZSERVICE

Some of the attributes in the ZSERVICE Class are calculated. They are updated when the service is installed, verified, updated, repaired, or deleted and reported in the agent computer's service objects. These attributes should *not* be modified using CM Admin CSDB Editor.

Table 26 Calculated ZSERVICE Attributes – DO NOT MODIFY

Attribute	Description
ZSVCCSTA	**Service Status on Client** Status code for the service. Used to determine why files for a service may not be deployed correctly. Values range from 000-999.
SIZE	**Application Size - Uncompressed** The size of the uncompressed application displayed to the subscribers in the extended information area in the CM Application Self-service Manager user interface. Since this is a calculated field, do not modify it. It is the cumulative value of the SIZE defined in the PACKAGE Class.

Attribute	Description
COMPSIZE	**Application Size - Compressed** The size of the compressed application displayed to the subscribers in the extended information area in the CM Application Self-service Manager user interface. Since this is a calculated field, do not modify it. This is the cumulative value of the COMPSIZE defined in the PACKAGE Class.
ZAVIS	**Available, Verified, Installed, Sync F** The CM agent manages and maintains this attribute to show the different states of the application in the catalog. The four states are: • **Available** indicates whether a service is available from the CM Configuration Server. • **Verified** indicates whether a service has been verified. • **Installed** indicates whether the service has been installed. • **Synchronized** indicates whether the installed service has all of the latest changes from the CM Configuration Server. The possible values for each are: Y = Yes N = No X = Unknown
VERDATE	**Verified Date of Service** Indicates when the application was last verified (in local time) on the agent computer. The CM agent manages and maintains this attribute. This is displayed to the subscribers in the extended information area in the CM Application Self-service Manager user interface. This attribute is useful for reporting purposes. The value is in the format of MMM DD,YYYY HH:MM:SS. Example: Jul 28, 2003 16:10:00

Attribute	Description
UPGDATE	**When Application was Upgrade on De** The CM agent manages and maintains this attribute. It indicates when the application was last updated (in local time) on the agent computer. This attribute is useful for reporting purposes. The value is in the format of MMM DD,YYYY HH:MM:SS. Example: Jul 28, 2003 16:10:00
INSTDATE	**Installed Date** Indicates when the application was installed (in local time) on the agent computer. The CM agent manages and maintains this attribute. This is displayed to the subscribers in the extended information area in the CM Application Self-service Manager user interface. This attribute is useful for reporting purposes. The value is in the format of MMM DD,YYYY HH:MM:SS. **Example**: Jul 28, 2003 16:10:00
DELDATE	**Delete Date** Indicates when the application was removed (in local time) from the agent computer. The CM agent manages and maintains this attribute. This attribute is useful for reporting purposes. The value is in the format of MMM DD,YYYY HH:MM:SS. Example: Jul 28, 2003 16:10:00

Summary

- Set the appropriate context, M, U, MU, or EMU, for a service using the ZSVCMODE attribute.
- If a service requires a reboot of the agent computer, use the REBOOT attribute in the Application (ZSERVICE) Class.
- Be aware of all of your service options in the Application (ZSERVICE) Class.

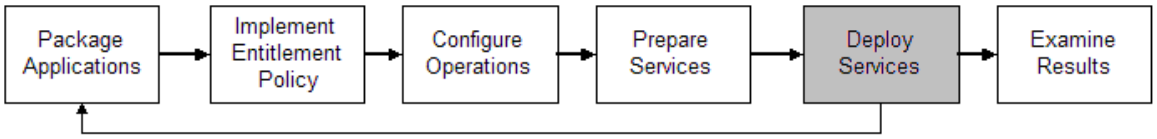
9 Deploying Services

At the end of this chapter, you will:

- Understand the different deployment methods available in CM and when to use each one.
- Be able to deploy a service at a predetermined time using the Scheduler.
- Know how to use the Notify function to update an application, remove an application, or send an e-mail message to a subscriber.
- Be familiar with key special case deployments.

This guide discusses the *suggested* implementation for the HP OpenView Configuration Management Application Manager (CM Application Manager). Although you will tailor this strategy to meet your organization's needs, we recommend that you review this guide for a comprehensive understanding of the CM Application Manager. This chapter describes deploying services.

Figure 16 Tasks described in this guide



About Deployment Methods

After creating a service using the CM Admin CSDB Editor for the HP OpenView Configuration Management Administrator Configuration Server Database Editor (CM Admin CSDB Editor), and deciding which users or groups will receive the application, you are now ready to deploy the service to your subscribers.

If you have used other systems management software, you may be familiar with the term job. A job is used to distribute a package. It includes a set of instructions to perform, a package containing the files or software, the targets for the job, and the schedule for carrying out the job.

In CM, you do not need to use a job. You can perform each of the steps – creating the package, defining targets (assigning users), and selecting a deployment method – individually. This provides flexibility because you can use multiple deployment methods to distribute a single package, based on the needs of your enterprise.

The following deployment methods are available in CM.

- **Scheduler**
  Installs the service at a specific time or sets any command line to run at an interval.
- **Notify**
  Forces one or more agent computers to connect to the CM Configuration Server to install, update, or remove an application, or sends an e-mail to the subscribers of a particular service.

The term **computer** is used to refer to a workstation or server.

Before selecting a deployment method, consider the following.

- Does the application need to be deployed at a certain time? If so, use Scheduler.
- Do you want to notify the users via e-mail when you are deploying the application? If so, use Notify.
- Do you want to install a new application, an update to an application, or remove an application? If so, use Notify.
- Are there multiple versions of the application? If so, use Version Groups. See Chapter 8, Preparing Services.

Testing Deployments

To ensure successful deployments, test your implementation rigorously.

- Publish and deploy software in a test environment before making the software available for live deployment.
- Test deployments to all target operating systems.
- Test all major capabilities of the deployment, including updates to the application, removing the application from the subscriber's desktop, customized installations, and variations in hardware configurations that might affect deployment, such as shortage of disk space, physical memory, and similar constraints.

Connection Parameters (Radskman)

No matter which deployment method you choose, you will need to create a radskman command line. Some deployment methods will create the command line for you. However, you should be aware of your options. Use radskman to:

- Check the status of all existing mandatory applications.
- Add new mandatory applications.

- Remove any mandatory applications that are no longer assigned to the subscriber.

You can specify your radskman command line from a command prompt, Scheduler (TIMER) instance, or Notify command. Before using any of these methods in a production environment, you should test the command line parameters you choose. The parameters can be divided into five categories:

- Core
- Operations
- Machine/User
- Client Operations Profiles
- Process
- In the tables below, the possible parameters for radskman are described. CM-AM stands for CM Application Manager and CM-ASM stands for CM Application Self-service Manager. After the tables, there are examples of radskman lines for common situations.

Core

Core parameters are used in most radskman lines. These parameters include the location of your CM Configuration Server, and how to identify the agent computer for policy.

Table 27 Radskman Core Parameters

Parameter	Explanation
cat	• Set `cat = prompt` to run self-maintenance, display the logon panel, and check the status of other services. • Set `cat = y` to simply check the status of services. • Set `cat = m` (4.0 feature) to use the local machine catalog for resolving the user's service list. This is used with context = u. Usually this is also used with local = y. CM-AM default: prompt. CM-ASM default: Depends on request type.

Parameter	Explanation
dname	The CM Configuration Server DB domain name for the services. This is the directory under which the service catalog (`ASERVICE.EDM`) is stored. For example, dname=SOFTWARE. Software CM-AM default: SOFTWARE. If preload=y, then the default is RADSTAGE. CM-ASM default: SOFTWARE. If preload=y, then the default is RADSTAGE.
IP	IP address of the CM Configuration Server. For example, IP = 10.10.1.001 Note: If you do not specify the IP address, CM uses the IP address specified in the ZMASTER object stored in IDMLIB (by default, /opt/HP/CM/Agent/lib). CM-AM default: NOVARCS (only defaults if no arguments are passed). CM-ASM default: NOVARCS (only defaults if *no* arguments are passed).
mname	Name of the CM Configuration Server. For example, mname=RADSVR01. CM-AM default: CM (defaults to RADSTAGE for preload). CM-ASM default: CM (defaults to RADSTAGE for preload).
port	CM Configuration Server port. The default for this is 3464. Note: If you do not specify the port, CM uses the port specified in the ZMASTER object stored in IDMLIB (by default, /opt/HP/CM/lib). CM-AM default: 3464. CM-ASM default: 3464.
sname	Specifies the service that you want to process. If you do not specify a service, then all mandatory services are processed.

Parameter	Explanation
startdir	Specifies the `IDMLIB` starting directory. Note: If uid is set on the command line, and startdir is not, then the startdir will be set to the same value as uid. If you specify a UID on the command line, we recommend specifying the STARTDIR as well. • Set `startdir = $MACHINE` to use the computer name. • Set `startdir = $USER` to use the currently logged on subscriber. • Set `startdir = value` to specify a custom starting directory. If *value* contains embedded spaces, enclose the entire name in double quotes. CM-AM default: $USER if started in a user context (context=u). SYSTEM if started in machine context (context=m). CM-ASM default: $USER if started in a user context (context=u). SYSTEM if started in machine context (context=m). CM Application Self-service Manager does not pass a context by default.
uid	Identification used to identify the current session. Note: If uid is set on the command line, and startdir is not, then the startdir will be set to the same value as uid. If you specify a uid on the command line, we recommend specifying the startdir as well. • `uid = $MACHINE` identifies the current session by the name of the computer. • `uid = $USER` identifies the current session by the name of the user currently logged on. • `uid=custom` is used to identify the current session by a custom value that you specify. CM-AM default: $USER if started in a user context (context=u). SYSTEM if started in machine context (context=m). CM-ASM default: $USER if started in a user context (context=u). SYSTEM if started in machine context (context=m). If you do not specify a context, the user ID, CM uses the LOCALUID specified in the ZMASTER object stored in IDMLIB (by default, `/opt/HP/CM/Agent/lib`).

Operations

These parameters influence how client will connect. Features include computer restart handling, log specifications, and the display options for the subscriber.

Table 28 Radskman Operations Parameters

Parameter	Explanation
ask	Set `ask = y` to prompt the subscriber before restarting the computer. This allows subscribers to save their work and close applications before the computer restarts. Set `ask = n` to restart the computer without prompting the subscriber. This is useful for unattended computers. CM-AM default: Y if CM System Tray is running. N if CM System Tray is *not* running or there are no users logged on. CM-ASM default: Y
hreboot	Set `hreboot = y` to allow radskman to handle a computer restart if it is required by the service. Set to p to power off the computer. If set to p, the agent computer will shut down no matter what the reboot settings are for a particular service. Note: This replaces handle_reboot. CM-AM default: Y CM-ASM default: N
ind	Set `ind=n` to hide the status indicator for each service. Set `ind=y` to show the status indicator for each service. CM-AM default: Y CM-ASM default: Y
jobid	Use jobid to further describe the source of this command line. It shows up in the APPEVENT, IDENTITY, PREFACE, SYNOPSIS as JOBID. CM-AM default: UserConnect if started in a USER context. MachineConnect if started in SYSTEM context. CM-ASM default: UserConnect if started in a USER context. MachineConnect if started in SYSTEM context.
log	Specifies the name of the log stored in the `IDMLOG` directory.

Parameter	Explanation
logsize	Specifies the size of the log file in bytes. When the logsize is reached, a backup file (`.BAK`) is created. By default, this file is `connect.bak`. If a backup file already exists, it will be overwritten. CM-AM default: 1000000 bytes CM-ASM default: 1000000 bytes
rtimeout	Specify number of seconds to wait if a reboot panel has been requested for a service before rebooting the agent computer. This will allow a subscriber time to save and close applications before a reboot.

Machine/User

Use these parameters when using applications with machine and user components, or when you have multiple users on the same agent computer. These parameters can control frequency of connection to the CM Configuration Server, display of the user logon panel, and when to send objects to the CM Configuration Server.

Table 29 Radskman Machine/User Parameters

Parameter	Explanation
cat	Set `cat = prompt` to display the logon panel, and check the status of other services. Set `cat = y` to simply check the status of services. Set `cat = m` (4.0 feature) to use the local machine catalog for resolving the user's service list. This is used with `context = u`. Usually, this is also used with `local = y`. CM-AM default: prompt CM-ASM default: Depends on request type

Parameter	Explanation

context	Set `context = m` when installing an application in the machine context.
	If `context = m` then the following defaults are assumed:
	`uid=$machine`
	`startdir=system`
	`cat=prompt`
	`ulogon=n`
	Set `context = u` when installing an application in the user context.
	If `context = u` then the following defaults are assumed:
	`startdir=$user`
	`uid=$user`
	`cat=prompt`
	`ulogon=y`
	CM-AM default: If started with a user logged on, the context defaults to u. If no user is logged on, then context defaults to m.
	CM-ASM default: There is no default for CM Application Self-service Manager. All components are processed.

flushu	If you are using `local=y`, set `flushu=y` on user connects (`context=u`) to send reporting objects up to the CM Configuration Server at the end of the local connect for immediate feedback. This is the default behavior on user connects. If you are using `local=y`, set `flushu = n` on a user connect (`context=u`) if you do not want the objects sent to the CM Configuration Server. Be aware that the user's objects will continue to grow until they are sent to the CM Configuration Server. On a machine connect (`context=m`), set `flushu=a`, if you want to send all user's reporting objects to the CM Configuration Server. CM-AM default: Y CM-ASM default: Y
local	Set this to y to install resources for the user's services from the local agent computer. Use this only with `context = u`. Usually, this is used with `cat = m`.

Parameter	Explanation
machfreq	(4.0 Feature) Use this variable to prevent CM from running every time an agent computer reboots. Set this to a positive integer, n, to run a machine connect only if it has been n hours since the last time a machine connect ran. This value ensures that the CM agent will not run more than once within the specified timeframe to reduce the number of ROM commits on a thin agent computer. If you set MACHFREQ to 0, the machine connect will run on *every* reboot of a thin client.
ulogon	*Only used if cat = prompt.* Set `ulogon = n` to hide the logon panel. Note: If using CM System Tray, set ulogon = n. This will display the CM logon panel, which is not supported by CM System Tray. CM-AM default: N CM-ASM default: Y
userfreq	*Only used if context=u.* Use this variable to prevent CM from running every time a user logs into the agent computer. Set this to 0 to run a user connect only if a machine connect has run since the last user connect. Set this to a positive integer, *n*, to run a user connect if a machine connect has run *or* it has been *n* hours since the last time a user connect ran. If the value of userfreq is blank or not supplied, then a user connect will run every time an agent connect is run with `context = u`.

Client Operations Profiles

These parameters are used for specifying how to use Client Operations Profiles. For more information, see Chapter 7, Configuring CM Client Operations Profiles.

Table 30 Radskman Client Operations Profiles Parameters

Parameter	Explanation
cop	(4.0 Feature) Set to Y to enable Client Operations Profile resolution for this agent connect only. Set to N to disable Client Operations Profiles resolution for this agent connect only. If the RADSEUP object exists the methods will not be run, but the other settings (from the CLIENT.SETTINGS class) will be used. Set to M to run a "method" connect. In other words, use the attributes specified in the RADSETUP object, including EXBSETUP, EXASETUP, EXBEXIT, EXBOUTBX, and CMETHOD, but do not do Client Operations Profiles resolution. CM-AM default: N CM-ASM default: N
datauri	(4.0 Feature) If you want to override the use of the SAP object for the Data Type, add `datauri` to the radskman command line. Datauri should be in the same format as the Universal Resource Identifier. For the syntax of this parameter, see Table 17 on page 124.
product	(4.0 Feature) If you used the SAP.PRODUCT attribute to identify that a SAP can only be used with a specific product, specify that product using this parameter. For example, if SAP.PRODUCT is set to IM, set `product=RIM` on the radskman command line. Specify multiple product filters separated by a comma.
rcsuri	(4.0 Feature) If you want to override the use of the SAP object for the RCS Type, add `rcsuri` to the radskman command line. `Rcsuri` should be in the same format as the Universal Resource Identifier. For the syntax of this parameter, see Table 17 on page 124.

Process

Process parameters involve service processing such as whether to repair or add applications on the current agent connect. These parameters also allow you to specify criteria for service processing, sending application data to a CM Proxy Server, and handling SSL security for your CM agents.

Table 31 Radskman Process Parameters

Parameter	Explanation
Add	Set this to N if you do not want to install applications during this agent connect. CM-AM default: Y CM-ASM default: Y
autofix	Set `autofix = y` to automatically repair any broken applications. Set `autofix = n` to prevent broken applications from being fixed. CM-AM default: Y CM-ASM default: Y
catexp	(4.0 Feature) Use this parameter to process applications based on a particular attribute in the ZSERVICE Class. Use the format *`attribute name`*`:`*`value`*. Specify multiple OR conditions with a forward slash (/). For example, to process only applications that have the ZSERVICE.CATGROUP attribute set to finance, set catexp=catgroup:finance.
del	Set this to N if you do not want to delete applications during this agent connect. CM-AM default: Y CM-ASM default: Y
merge	Set merge equal to an object name to have all variables in that object includes in the ZMASTER object. Do this to send the variables to the CM Configuration Server.
mnt	(4.0 Feature) Set to Y to process Agent Self Maintenance on this connect. Note: Agent Self Maintenance will not be deployed unless you set `mnt=Y`. CM-AM default: N CM-ASM default: N
preload	Use this for staging server preload. Specify the location of directory to copy the files to. If you do not need or want to specify a different data directory, setting `preload=y` uses the `IDMDATA` directory specified in `NVD.INI`.

Parameter	Explanation
rep	Set this to N if you do not want to repair applications during this agent connect session. CM-AM default: Y CM-ASM default: Y
sendcat	Set this to **y** to send the service list, stored in the agent computer's ASERVICE object, to the CM Configuration Server at the end of the agent connect so that additional analysis can be done on the service list.
sslmgr	Specifies the hostname or IP address of the CM Configuration Server. Note: To perform agent self-maintenance over a secure channel (SSL), add the `flag, ::sm`, to the end of the SSL Manager IP address.
sslport	Specifies the port for SSL communications (normally, 443).
upd	Set this to N if you do not want to update applications during this agent connect session. CM-AM default: Y CM-ASM default: Y
ver	Set this to N if you do not want to verify applications during this agent connect session. CM-AM default: Y CM-ASM default: Y

radskman Examples:

The following examples are provided to illustrate common uses of radskman.

```
radskman
ip=10.10.10.15,port=3464,mname=cmcs,dname=software,cat=prompt
```

Performs a first catalog refresh that brings down the catalog (`aservice.edm`), runs self-maintenance, does not display the user logon panel if using CM Application Manager, and processes all mandatory applications:

```
radskman
ip=test.corp.com,port=3464,mname=cmcs,dname=software,cat=prompt,u
```

```
id
=$machine,ulogon=n,ind=n
```

Performs a full connect for user <machine name> silently with no user logon panel or progress indicator panels. This is a typical command used by a daily timer. Note: the ip= parameter can be a DNS name or IP address.

```
radskman
ip=10.10.10.15,port=3464,mname=cmcs,dname=software,cat=n,autofix=
n
```

Verify mandatory applications *without* updating the catalog, running self-maintenance, or repairing broken applications. Note: This machine must perform a first refresh catalog using cat=prompt at least once to bring down the catalog prior to using cat=n:

```
radskman
ip=10.10.10.15,port=5004,mname=cmcs,dname=software,cat=y,sname
=WINZIP
```

Install a single application with the service name of WINZIP, while only updating the catalog. Note: In this example, the CM Configuration Server uses a custom port number.

```
radskman
ip=10.10.10.15,port=3464,mname=cmcs,dname=software,cat=prompt,
hreboot=Y,ask=Y
```

Process all mandatory applications, handle reboot requests, and prompt the user with a panel to confirm the reboot request:

```
radskman ip=10.10.10.15,port=3464,uid=STAGER,preload=Y
```

Silently preload a CM Staging Server using the default location of RADSTAGE. This syntax is often run by a daily or weekly timer instance that is deployed to a CM agent that is co-located on the same machine as the CM Staging Server. Note: The preload parameter automatically suppresses the user logon panel and progress indicator panels. The preload parameter also does not run any methods (ZCREATE, ZVERIFY, etc.) or evaluate any expressions on the CM Configuration Server.

```
radskman
ip=10.10.10.15,port=3464,uid=STAGER,preload=d:\stager,ind=Y
```

Preload a stager using a location of `d:\stager` and display the progress indicator panels. Note: If the CM System Tray feature is enabled, then the progress indicator will be displayed in the CM System Tray information bubble. If the CM System Tray is disabled, then the progress indicator will be displayed in a separate panel.

```
radskman context=m
```

Perform a machine connect. Because context was specified as m, and no other parameters were passed the following default values are used: `ip=NOVARCS`, `port=3464`, `uid=$machine`, `startdir=system`, `cat=prompt`, `ulogon=n`, `mname=cmcs`, `dname=software`.

```
radskman context=u
```

Perform a user connect. Because context was specified as u, and no other parameters were passed the following default values are used: `ip=NOVARCS`, `port=3464`, `uid=$user`, `startdir=$user`, `cat=prompt`, `ulogon=y`, `mname=cmcs`, `dname=software`.

```
radskman context=u,userfreq=12
```

Perform a user connect only if a machine connect has occurred since the last user connect and there has been at least 12 hours since the last user connect.

Deployment Methods

The following section covers each of the deployment methods in detail. Select the appropriate method for your subscribers. Remember, you can use multiple deployment methods to distribute a single application.

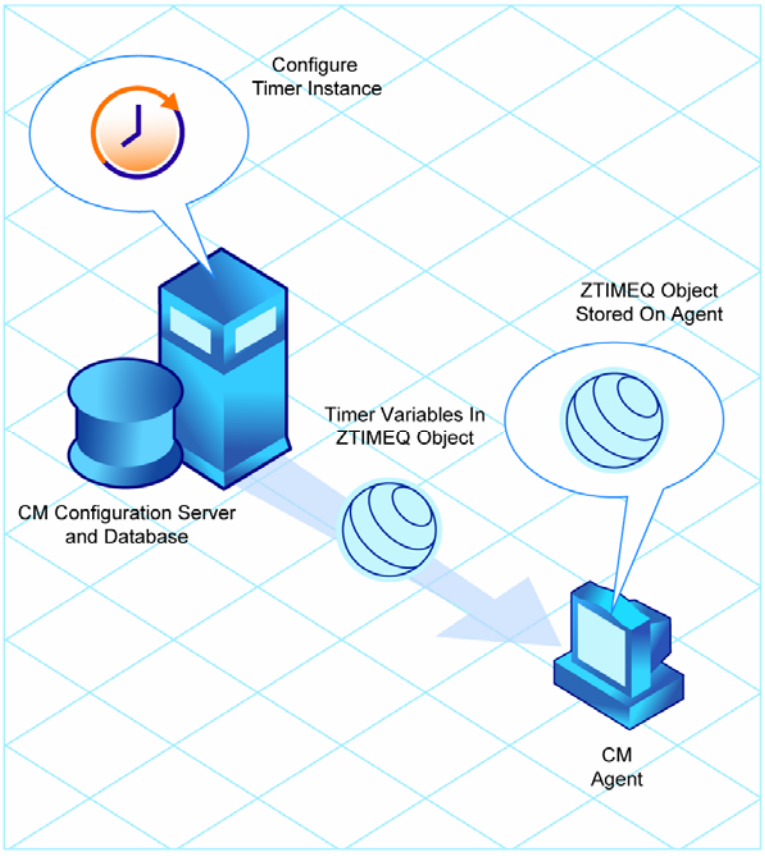
Scheduling (TIMER)

The CM Scheduler service, **radsched**, is installed with the CM Application Manager. The CM Scheduler allows you to deploy a service at a specific time. It wakes up once a minute to see if there are any scheduled items to execute.

The radsched daemon may be started as a service on UNIX workstations. We recommend running the radsched daemon as root. Consult your company's CM system administrator for more information.

The information about when to deploy the service is stored in two places. First, the time and date are configured in the Scheduler (TIMER) instance in the SOFTWARE Domain. The next time the agent computer connects to the CM Configuration Server, the ZTIMEQ object is created on the agent computer, and the timer variables are transferred to the ZTIMEQ object.

Figure 17 Transferring the timer instance



This section describes how to create and configure a timer, and then connect it to the service that you want to deploy. However, before creating and configuring a timer, consider the following.

- What time of day should the timer expire? Be sure to consider network traffic.
- How often do you want the timer to expire? Do you want the timer to expire daily, weekly, hourly, etc.?
- Does the timer need to expire more than once? For example, do you need to install the application only one time? Or, are you creating a timer that will check for mandatory applications every so often?

- What should happen when the timer expires? For example, do you want to launch, install, remove, or update an application?

Scheduled Deployment Strategy

One of the *suggested* strategies for implementing the CM Application Manager agent is the scheduled deployment strategy. This strategy installs an initial set of mandatory applications when you install the CM Application Manager, and transfers a timer to the agent computer that checks for new mandatory applications at the specified interval.

In this section, we will create a sample timer that updates all mandatory services on a weekly basis. In order to alleviate network congestion, the timer will expire randomly between 5:00 PM and 7:00 PM. Use the information in this section to configure timers based on your needs.

Creating a Timer

To create a timer, use the CM Admin CSDB Editor to create a Scheduling (TIMER) instance in the SOFTWARE Domain.

To create a new timer in the SOFTWARE Domain

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

The CM Admin CSDB Editor Security Information dialog box opens.

The User ID, as shipped from HP, is RAD_MAST. No password is necessary. This might have been changed during installation. You can also change this by selecting the **Change Password** check box and typing the new password in the New Password and Verify New Password text boxes.

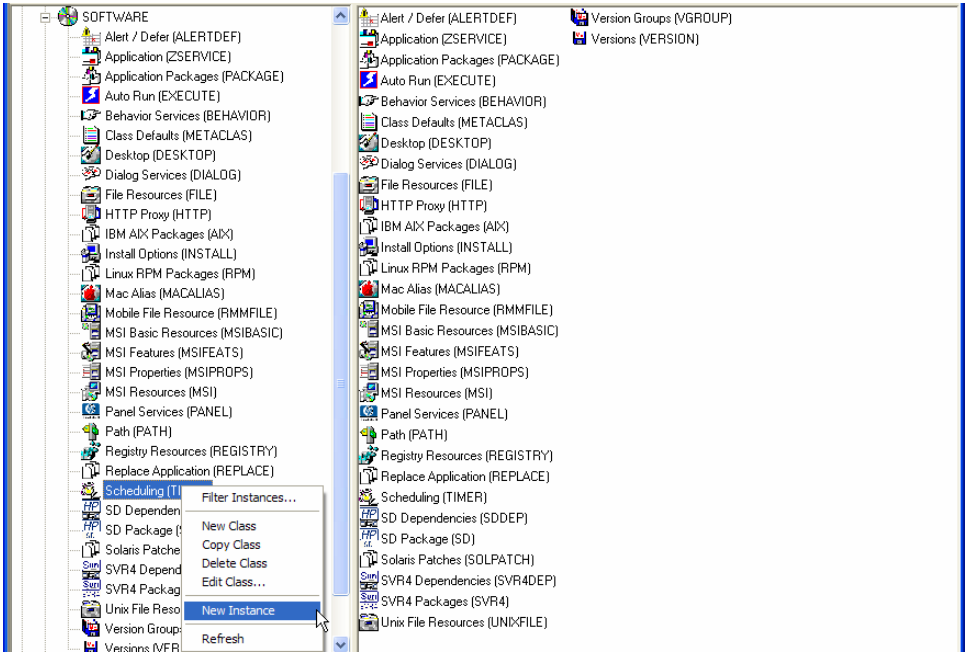
2 If necessary, type a User ID and Password, and then click **OK**.

3 The CM Admin CSDB Editor window opens.

4 Double-click **PRIMARY**.

5 Double-click **SOFTWARE**.

6 Right-click **Scheduling (TIMER)**.



7 Select **New Instance**.

8 The Create Instance dialog box opens.

9 Type a name for the new timer instance, such as **`Mandatory Apps Timer`**.

10 Click **OK**.

The TIMER instance appears in the Scheduling (TIMER) Class.

The Scheduling (TIMER) Class Attributes

The attributes in the TIMER instance contain the information needed to execute the timer on the agent computer. The following table describes these attributes.

Table 32 Scheduling (TIMER) attributes

Attribute	Usage
ALWAYS	Stores connections to other instances.
NAME	The friendly name for this instance.

Attribute	Usage
NETAVAIL	If set to Y, check for network availability before executing the TIMER instance. If the network is not available, network availability will be checked every time the timer wakes up until the network is available. If set to N, the TIMER instance will be executed without checking for network availability. If set to W, check for network availability before executing the TIMER instance. If the network is not available and the time window's end limit has been reached, the timer will wait until the next time window before checking for network availability again. The default is N.
PINGDLAY	If ZNOPING is set to N, PINGDLAY specifies the time in milliseconds between pings. The default is 2000.
PINGCNT	If ZNOPING is set to N, PINGCNT specifies number of ping attempts. The default is three attempts.
RETRYFLG	Set to Y to retry the command up to the number of times specified in RETRYLMT, ignoring the end time for the timer. Set to W to retry the command up to the number of times specified in RETRYLMT, but stop retrying after the specified limit time has passed. Set to N to not retry. Note: a return code other than 200 will indicate success, and stop the retries.
RETRYINT	Specify number of minutes to wait between command executions. RETRYFLG must not be set to N.
RETRYLMT	Specify the number of times it to retry the command. Set this to 0 will retry until the command succeeds. RETRYFLG must not be set to N.
RETRYRC	Specify return codes that qualify for the retry logic. If this variable does not exist or is blank, RETRYRC will default to 200. A return code of 200 means that there was a fatal error due to a network connection failure with the CM Configuration Server. If you populate this attribute, and a return code of 200 qualifies for a retry, be sure to specify 200 in the list. Example: RETRY = 200, 202, 209
ZNOPING	Use this attribute to control automatic sensing of a

Attribute	Usage
	network connection between the agent computer and the CM Configuration Server. The default is Y. An expired timer continually evaluates whether communications with the CM Configuration Server can be established. When communications are established, the command line associated with the timer is executed. After executing the command line, the Scheduler service resumes normal evaluation of whether the timer has expired again. If the ZNOPING attribute *does not exist* in the ZTIMEQ object, the Scheduler service does *not* ping the CM Configuration Server. Set ZNOPING to Y if you want to prevent the Scheduler service from pinging the CM Configuration Server. This is especially useful for mobile users. Set ZNOPING to N if you want the Scheduler service to ping the CM Configuration Server. If the CM Configuration Server is pinged successfully, the command in ZRSCCMDL executes and the ZPENDING attribute, in the client's ZTIMEQ object, is set to N, to indicate that the Scheduler service does not need to ping the CM Configuration Server again. If the CM Configuration Server is *not* pinged successfully, the timer is not processed any further, and the ZPENDING attribute value remains Y, to indicate that the next time the Scheduler service "pops", it should ping the CM Configuration Server again. Set ZNOPING to W if you are specifying an end limit in the ZCHDEF attribute. The Scheduler will ping the CM Configuration Server before executing the command. If the CM Configuration Server is unavailable, then the ZPENDING flag will be set to "W". If the ZSCHEDEF has a limit time, then when that time passes, the ZPENDING flag will be set to N, and the Scheduler will not attempt to execute the command until its next scheduled time.

Attribute	Usage
ZRSCCMDL	Use this attribute to specify the command line that is executed on the agent computer when the timer expires. Use radskman to verify and update CM-managed mandatory applications. See Connection Parameters (Radskman) on page 183 for a complete list of the parameters and examples.
ZSCHDEF	Use this attribute to specify when the timer expires. The syntax for this attribute varies depending on the frequency, which can be DAILY, HOURLY, INTERVAL, MONTHLY, MONTHDAY, NUMDAY, STARTUP, WEEKDAY, WEEKLY. See Specifying When the Timer Expires on page 203 for instructions on how to set ZSCHDEF.
ZSCHFREQ	Use this attribute to specify how often the timer should expire. Set ZSCHFREQ to ONCE if you want the timer to expire one time. Set ZSCHFREQ to PERIODIC if you want the timer to expire repeatedly. Set ZSCHFREQ to RANDOM if you want the timer to expire in random intervals. See Deploying Applications over a Period of Time on page 206 for more information.
ZSCHTYPE	*Used only when ZSCHFREQ = PERIODIC.* Set ZSCHTYPE to DEFERRED to indicate that the first time an event is attempted to be launched, it will be deferred until the *next* scheduled time, no matter when the timer instance is evaluated. This was designed to handle the case of a daily 4 A.M. (non-peak) scheduled event that is sent to the agent computer during the day. If it was not deferred, it would launch during the day instead of "waiting" until the next morning. Example 1: Suppose you create and deploy a timer with the ZSCHDEF = DAILY(&ZSYSDATE,4:00:00) If ZSCHTYPE = IMMEDIATE and it is: Before 4:00:00, the command in the instance will be

Attribute	Usage
	executed the same day at 4:00:00. After 4:00:00, the command in the instance will be executed immediately. If ZSCHTYPE = DEFERRED and it is: Before 4:00:00, the command in the instance will be executed the *next* day at 4:00:00. After 4:00:00, the command in the instance will be executed the *next* day at 4:00:00. Example 2: Suppose you create and deploy a timer with the ZSCHDEF = WEEKDAY(FRIDAY,4:00:00) If ZSCHTYPE = IMMEDIATE and it is: Not Friday or Friday and before 4:00:00, the command in the instance will be executed on Friday at 4:00:00. Friday and after 4:00:00, the command in the instance will be executed immediately. If ZSCHTYPE = DEFERRED and it is: Not Friday or Friday and before 4:00:00, the command in the instance will be executed a week later on Friday at 4:00:00. Friday and after 4:00:00, the command in the instance will be executed a week later on Friday at 4:00:00.
ZSTOP	Expressions evaluating to "true" in ZSTOP attributes cause resolution of the instance to be skipped. If left blank, the instance is accepted, and resolution continues. This is useful if you want to set conditions on which of your subscribers receive the timer.
The values for the following attributes are set from the BASE INSTANCE of the TIMER class and should not be edited.	
RUNSYNC	Specifies if synchronous timer execution will take place. The default value is Y.
ZOBJPRI	Indicates the priority for deployment of the ZTIMEQ object, relative to the other elements deployed during the agent connect. Elements with priority numbers lower than the value of ZOBJPRI are deployed before this ZTIMEQ object. A value of 90 is inherited from the BASE INSTANCE.

Attribute	Usage
ZSCHMODE	Specifies the timer owner. Leave as Default.
ZSVCOID	Specifies the object ID of the Application instance that this Scheduling instance is connected to. The value is inherited from the BASE INSTANCE.
ZCHNNAME	Specifies the name of the domain in the CM Configuration Server DB where the Application instance to which this Scheduling instance is connected. The value is inherited from the BASE INSTANCE.
ZPRVNAME	The name of the CM Configuration Server that the subscriber receiving this timer instance is connected to. The value is inherited from the BASE INSTANCE.
ZCREATE	The Scheduler Create method that runs on the agent computer. The value is inherited from the BASE INSTANCE.
ZVERIFY	The Scheduler Verify method that runs on the agent computer. The value is inherited from the BASE INSTANCE.
ZUPDATE	The Scheduler Update method that runs on the agent computer. The value is inherited from the BASE INSTANCE.
ZDELETE	The Scheduler Delete method that runs on the agent computer. The value is inherited from the BASE INSTANCE.

Configuring the Timer

Now that you are familiar with the attributes in the timer instance, we will review the syntax used to configure these attributes. Then, we will configure the sample timer using the appropriate values to deploy mandatory applications every week at random intervals 5:00 PM and 7:00 PM.

Specifying When the Timer Expires (ZSCHDEF)

Use the ZSCHDEF and ZSCHFREQ attributes respectively to specify when the timer should expire, and how often. ZSCHDEF indicates when the timer should expire and ZSCHFREQ indicates how often the timer should expire.

If you want your timer to expire only one time (ZSCHFREQ = ONCE), or repeatedly (ZSCHFREQ = PERIODIC) at a certain time, use the following table to determine the appropriate syntax for the value of ZSCHDEF.

Table 33 Syntax of ZSCHDEF Attribute

* Name of Weekday is the name of a specific weekday, e.g., Monday.

Attribute	Usage	Timer Expires
DAILY	DAILY(&ZSYSDATE, 24:00:00)	Daily at midnight on system's date.
WEEKLY	WEEKLY(&ZSYSDATE, 01:00:00)	Every 7 days at 1:00 AM.
WEEKDAY	WEEKDAY(Name of Weekday*,01:00:00)	Every *Name of Weekday** at 1:00 AM. The weekday must be specified in uppercase.
HOURLY	HOURLY(&ZSYSDATE, 08:41:00)	Hourly starting at 8:41 AM on system's date.
INTERVAL	INTERVAL(&ZSYSDATE, 08:41:00,,30)	Every 30 minutes starting at 8:41 A.M. based on system's date. Note: When setting ZSCHFREQ to RANDOM, there are four possible arguments for INTERVAL. The extra comma is required whether ZCHFREQ is set to RANDOM or not. Use the second argument to set the start time and the third argument to set the end time of the random interval.

Attribute	Usage	Timer Expires
MONTHDAY	MONTHDAY(*Name of Weekday**,01:00:00,,2)	Every *Name of Weekday** at 1:00 A.M. on the second week of the month. The weekday must be specified in uppercase. If the last argument is not specified, then the timer will run on the first week of the month. Note: When setting ZSCHFREQ to RANDOM, there are four possible arguments for MONTHDAY. The extra comma is required whether ZCHFREQ is set to RANDOM or not. Use the second argument to set the start time and the third argument to set the end time of the random interval.
MONTHLY	MONTHLY(20040215, 01:00)	Runs on the 15th of the month starting in February at 1:00 AM. If setting ZSCHFREQ to RANDOM, use the second argument to set the start time and add a third argument to set the end time of the random interval. Note: MONTHLY reschedules differently than other schedule frequencies MONTHLY will reschedule from the original day it was scheduled for, instead of the day it ran. For example, if ZSCHDEF was MONTHLY(20040116,05:30:00) and the agent device was off on the 16th of January, and didn't execute until the 18th of January, the new schedule would be MONTHLY(20040216,05:30:00) instead of MONTHLY(20040118,05:30:00).
NUMDAYS	NUMDAYS(20000803, 08:00:00,,14)	Every 14 days starting on August 3, 2000 at 8:00 AM.

Attribute	Usage	Timer Expires
STARTUP	STARTUP	When the CM Scheduler starts on the agent device, it will immediately execute all Timer instances with ZSCHDEF of STARTUP. It will check for all special conditions (NETAVAIL, ZNOPING and RETRYFLG). After it executes all the STARTUP instances, RADSCHED then goes back into its regular timer loop. It will only execute STARTUP instances in the regular timer loop if the ZPENDING flag on that instance was set (because NETAVAIL or ZNOPING could not get through or RETRYFLG is on and the return code was 200 during startup run).

Deploying Applications over a Period of Time

You can also deploy applications over a specified range of time. This varies the load on the CM Configuration Server and alleviates network congestion by spreading out the deployment over a period.

To do this, indicate that you want your timer to expire randomly (ZSCHFREQ = RANDOM). Then, use ZSCHDEF to specify the range of time over which the applications should be deployed.

If the ZSCHDEF is WEEKDAY(TUESDAY,<start>,<end>) then the Scheduler is going to execute it on TUESDAY, and only on TUESDAY. If <start> is 22:00:00 and <end> is 05:00:00, then the <end> time is no longer on TUESDAY, and therefore not a valid time. The RADTIMEQ create method enforces this by not allowing a RANDOM time to be created for WEEKDAY (or MONTHDAY) if the time spans midnight.

Note: If the ZSCHDEF is WEEKDAY(TUESDAY,<start>,<end>) then the Scheduler is going to execute it on TUESDAY, and only on TUESDAY. If <start> is 22:00:00 and <end> is 05:00:00, then the <end> time is no longer on TUESDAY, and therefore not a valid time. The RADTIMEQ create method enforces this by not allowing a RANDOM time to be created for WEEKDAY (or MONTHDAY) if the time spans midnight.

To do this, the syntax for ZSCHDEF is:

```
ZSCHDEF =<frequency>(<date>, <from_time>, <to_time>, <limit>)
```

Table 34 ZSCHDEF Parameters for ZSCHFREQ set to RANDOM

Parameter	Description
<frequency>	Any of the supported frequency values, including DAILY, HOURLY, INTERVAL, NUMDAY, WEEKLY, and WEEKDAY.
<date>	Date when the event should be initiated. The format is YYYYMMDD.
<from_time>	Beginning time for randomization. The format is HH:MM:SS.
<to_time>	Ending time for randomization. The format is HH:MM:SS.
<limit>	The optional parameter that prevents initiation after this time (HH:MM:SS). The format is HH:MM:SS.

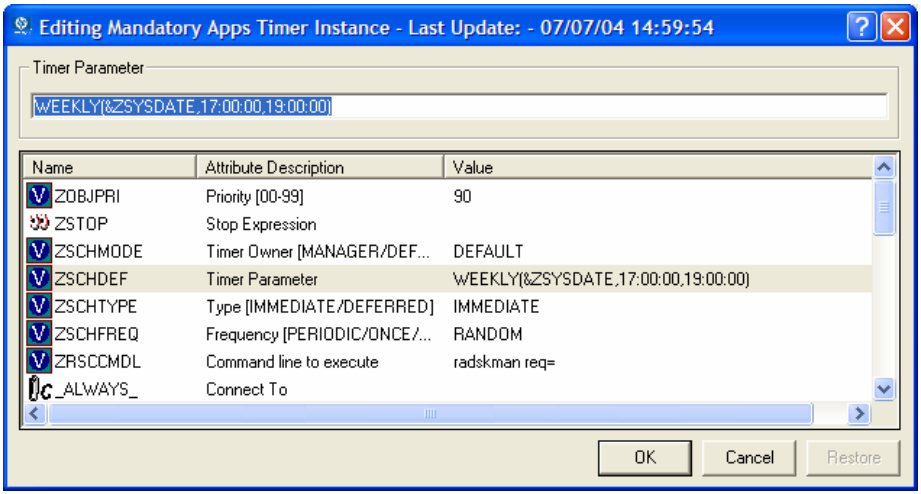
In our example, we want to configure the timer to deploy mandatory applications on a weekly basis. However, to alleviate network congestion, we are going to schedule deployments to run between 5:00 P.M. and 7:00 P.M.

To specify when the timer expires

1. After navigating to the timer instance, double-click **ZSCHFREQ** in the list view of the CM Admin CSDB Editor.

   The Editing Instance dialog box opens.

2. In the Frequency drop-down list, select **RANDOM**.
3. Click **ZSCHDEF**.
4. In the Timer Parameter text box, type **`WEEKLY(&ZSYSDATE,17:00:00, 19:00:00)`**.



5 Click **ZSCHTYPE**.

6 In the Type (Immediate/Deferred) drop-down list, select **IMMEDIATE**.

7 If you are done editing the attributes for the timer instance, click **OK**, and then click **Yes** when you are prompted to confirm your changes.

or

8 Select the next attribute to edit.

Specifying the Command Line (ZRSCCMDL)

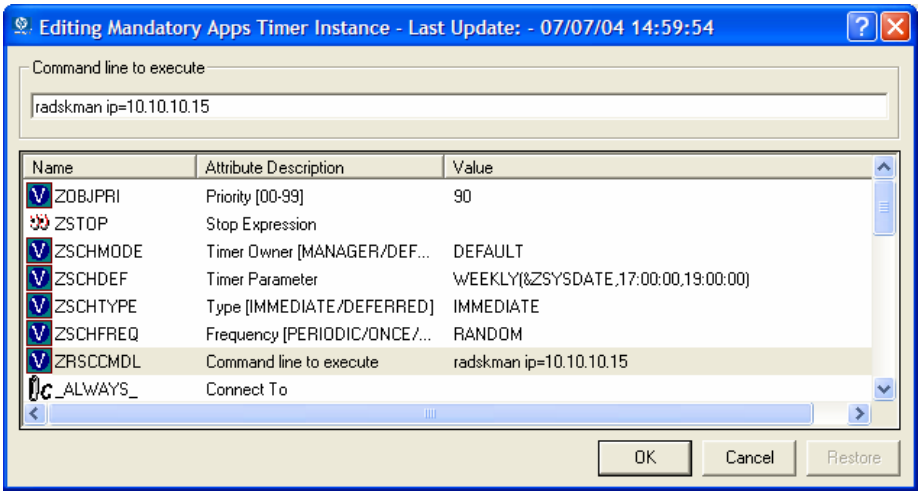
When the timer expires, it executes any command line that you specify on the agent computer.

If you want to see how timers work, you might create a timer that runs a command line such as *SystemDrive*:\Notepad.exe. Remember to configure the timer to expire immediately, and attach it to a service. Then, deploy the service. When the timer expires on the agent computer, the Notepad application opens.

To specify a command line

1 Navigate to the timer instance and then double-click **ZRSCCMDL** in the list view of the CM Admin CSDB Editor.

The Editing Instance dialog box opens.



2 In the Command line to execute text box, type the command line to execute the appropriate program for your needs.

3 Click **OK**.

4 Click **Yes** when you are prompted to confirm your changes.

In our example, we indicated that we would be deploying new mandatory applications to your subscribers on a weekly basis. The following procedure will show you how to specify a command line that will update *all* mandatory services and perform self-maintenance.

Connecting the Timer to a Service

Once you have created the timer, you must connect it to a service. Each subscriber that receives the service to which the timer is connected will receive the timer information in the ZTIMEQ object the next time his CM agent connects to the CM Configuration Server.

In the example we have used throughout this section, we created a timer intended to deliver mandatory applications. We will connect the sample timer to the GS-Calc service. We assume that all subscribers are receiving this service.

1 To connect the timer to the GS-Calc service,

2 In the CM Admin CSDB Editor, double-click **PRIMARY**.

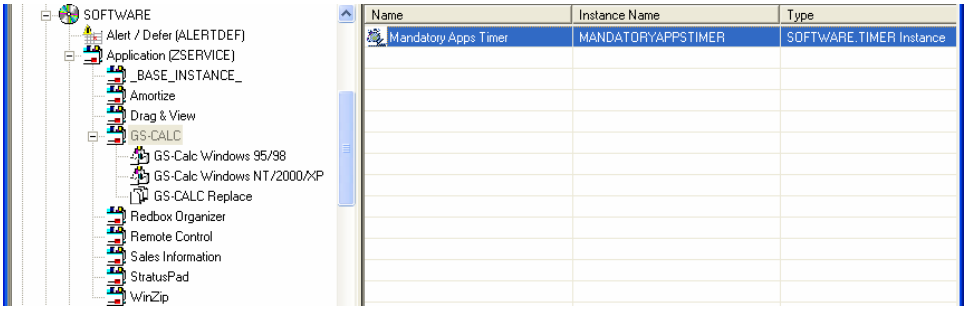
3 Double-click **SOFTWARE**.

4 Double-click the **Application (ZSERVICE)** class.

5 Right-click on the Application class (in this example, GS-CALC). A menu opens.

6 Select **Show Connections**. The SOFTWARE.ZSERVICE Connections dialog box opens.

7 Select **Scheduling (TIMER)**, and then click **OK**. The Timer class instances appear in the list view.



8 Click **Mandatory Apps Timer** in the list view and drag it to the appropriate Application (ZSERVICE) instance (in this example, GS-Calc). When your cursor changes to a paper clip, release the mouse button. The Select Connection Attribute dialog box opens.

9 Click **Copy**.

10 Click **Yes** to confirm that you want to connect the GS-Calc service to the Mandatory Apps Timer.

11 Click **OK** to close the confirmation message.

Testing the Timer Deployment

The first time an agent computer connects to the CM Configuration Server after the timer is created, the timer information is transferred to the agent computer in the ZTIMEQ object. The ZTIMEQ object is located in the IDMROOT directory

In this section, we will force the agent computer to connect to the CM Configuration Server so that you can view the ZTIMEQ object. For testing purposes, we will run a command line on the agent computer that updates *all* mandatory services and performs any agent self-maintenance.

To connect to the CM Configuration Server

1. On the agent computer, go to a command prompt and change the directory to `/opt/HP/CM/Agent`. This is the default location for radskman.
2. Type **`radskman ip=<manager ip>,port=<mgr_port>`**

   See Connection Parameters (Radskman) on page 183 for information about radskman and the parameters above.

   > If you plan to do more testing, consider creating a batch file that contains the command line. Save the file in `IDMSYS` on the agent computer. Then, create a shortcut on the desktop of the agent computer.

3. Press **Enter**. Once the agent connect is finished, you can view the ZTIMEQ object on the agent computer.

Viewing the Timer Object (ZTIMEQ.EDM)

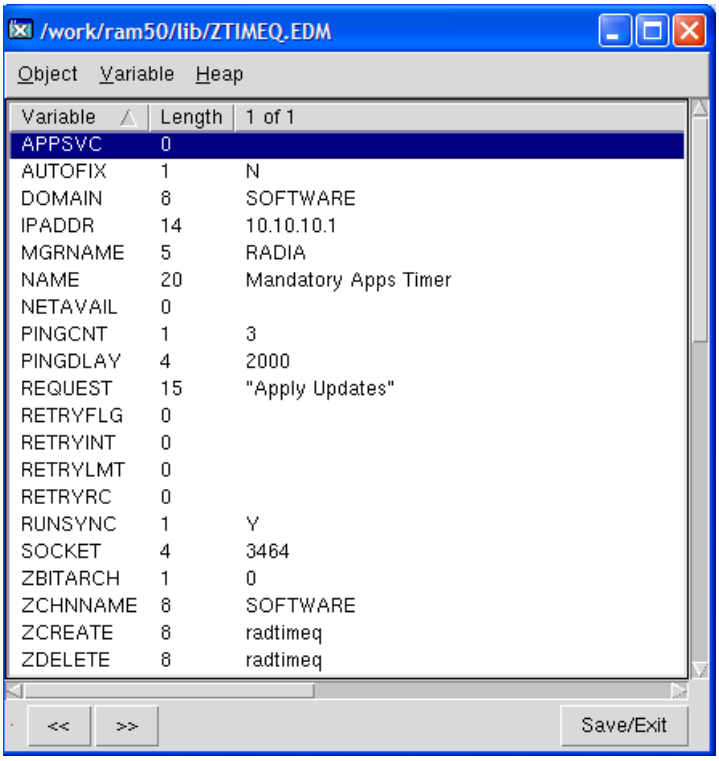
Now that we have forced the agent computer to connect to the CM Configuration Server, the ZTIMEQ object is stored on agent computer. Use the CM Admin Agent Explorer, installed as part of the CM Administrator, to view or modify the ZTIMEQ object.

> After the timer expires, the ZTIMEQ object is removed from the agent computer during the next agent connect.
>
> If the timer is configured to expire only one time, in the TIMER.ZSCHFREQ attribute, it will be removed immediately after the timer expires, during the next agent connect. If the timer is configured to expire more than one time, the ZTIMEQ object will be removed after the timer expires for the last time, during the next agent connect.

The ZTIMEQ object contains one instance for each Scheduling (TIMER) instance in the CM Configuration Server DB. For example, if two different services each have timer instances associated with them, then there will be two instances in the ZTIMEQ object.

To view the ZTIMEQ object on the agent computer

1. Go to the directory where you installed the CM Administrator.
2. Type **`./radobjed`** and press **Enter**.
3. Double-click the **ZTIMEQ** object. The ZTIMEQ object opens.



Experimenting with Timers

If you want to experiment with timers, you can modify the ZRSCCMDL, ZSCHDEF, ZSCHFREQ, and ZSCHTYPE attributes in the ZTIMEQ object on the agent computer to see what happens in various situations.

To edit an attribute in ZTIMEQ

1 Double-click the attribute that you want to edit.

The Change Variable dialog box opens.

2 Type the new value.

3 Click **Save/Exit**.

To quickly determine whether the timer expires, you can change ZRSCCMDL to run any executable, such as Notepad. When the timer expires, Notepad opens, confirming that the timer expired.

Timer Logs

Timer events are tracked in three logs, stored in the `IDMLOG` directory (by default, `/opt/HP/CM/Agent/log`).

The following table describes the timer logs.

Table 35 Timer Logs

Log File	Usage
`RADSCHED.LOG`	Lists the results of the most recent Scheduler expiration. The Scheduler, radsched, runs in the background. It wakes up once a minute and examines the ZTIMEQ agent object to see if a timer has expired. This log only retains information from the most recent expiration.
`RADSHIST.LOG`	Lists all of the programs dispatched because a timer instance expired. It reflects all activity since radsched was started last.
`RADTIMEQ.LOG`	Lists the events that occurred during the last execution of the radtimeq method. This method executes when the application to which the timer is attached is created, updated, verified, or deleted. Only the last execution's events appear in the log, with an indication of what activity took place regarding the application.

Notifying Subscribers

Use CM Notify to force one or more agent computers to connect to the CM Configuration Server to install, update, or remove an application. Each agent computer runs the CM Notify service in the background. This service waits to receive a Notify message from the CM Configuration Server. When a message is received, the agent computer connects to the CM Configuration Server and performs the action initiated by the Notify operation. CM Notify can also send e-mail notification to agent computers.

If you are using the HP OpenView Configuration Management Portal, you can use the Notify Task. See the *HP OpenView Configuration Management Portal Installation and Configuration Guide (CM Portal Guide)* for details.

You can initiate a Notify by:

- Selecting **Notify Subscribers** from the shortcut menu for an Application (ZSERVICE) instance. Use this option t *only* to update or remove applications. You cannot use this type of Notify to install an application because this option notifies *existing* subscribers.
- Creating a Drag-and-Drop Notify command. Use this option to install, update, or remove an application. The benefit of this type of Notify is that the application does not have to be installed on the agent computer to perform the Notify.

Drag-and-Drop Notify is intended for use in environments with a single CM Configuration Server. If you are working in an environment with multiple CM Configuration Servers, consider using the CM Portal. Contact your sales representative for details.

Requirements for Using Notify

To use Notify

- The agent computer must connect to the CM Configuration Server prior to the notification. This populates the PROFILE File, which contains the agent computer's network address, used by Notify.

Notify is designed to notify only subscribers whose information is in the PROFILE File in the CM Configuration Server DB.

- Confirm that the CM Configuration Server settings file, `EDMPROF.DAT`, is configured properly, as shown in the code sample below. This file and the relevant lines are created in the CM Configuration Server settings file when the server is installed.

```
[MGR_ATTACH_LIST]
ATTACH_LIST_SLOTS = 15
RESTART_LIMIT = 7
VERIFY_INTERVAL = 5
CMD_LINE=(zutilmgr) RESTART=YES
CMD_LINE=(zrexxmgr) RESTART=YES
CMD_LINE=(ztcpmgr PORT=3464,NAME=tcpmgr_3464) RESTART=YES
CMD_LINE=(znfytmgr NAME=NotifyManager) RESTART=YES
```

```
CMD_LINE=(zrtrymgr) RESTART=YES
```

- If you are using Drag-and-Drop Notify to run a command, you must store the program that you want to execute in the `IDMSYS` directory (by default `/opt/HP/CM/Agent`).
- If you are using e-mail to notify subscribers, be sure that the correct e-mail address for the subscriber is stored in the EMAIL attribute of the USER instance in the USER Class in the POLICY Domain.
- If you are using e-mail to notify subscribers, be sure that the CM Configuration Server is properly configured for email.

Initiating a Notify from a ZSERVICE Instance

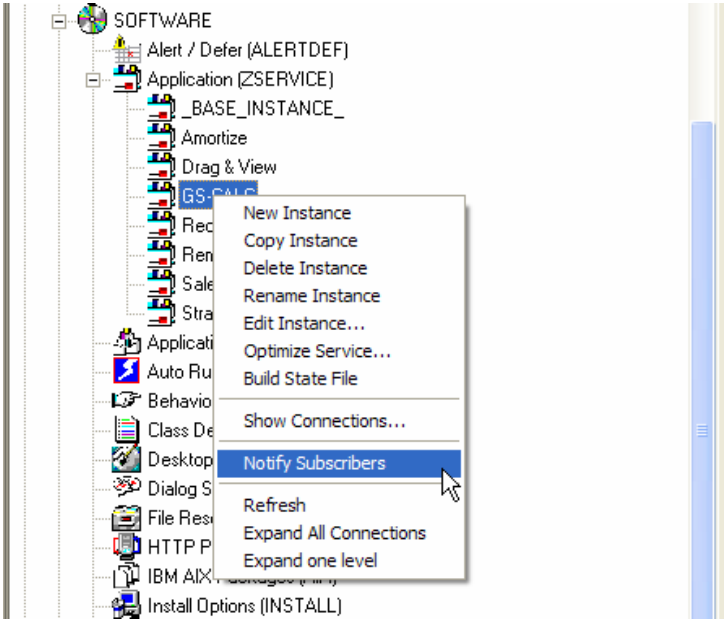
To update or remove a service, initiate the Notify from the shortcut menu for the Application (ZSERVICE) instance.

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.

To initiate a Notify from a ZSERVICE Instance

1 Right-click the **Application (ZSERVICE)** instance, such as **GS-CALC**.

A shortcut menu opens.



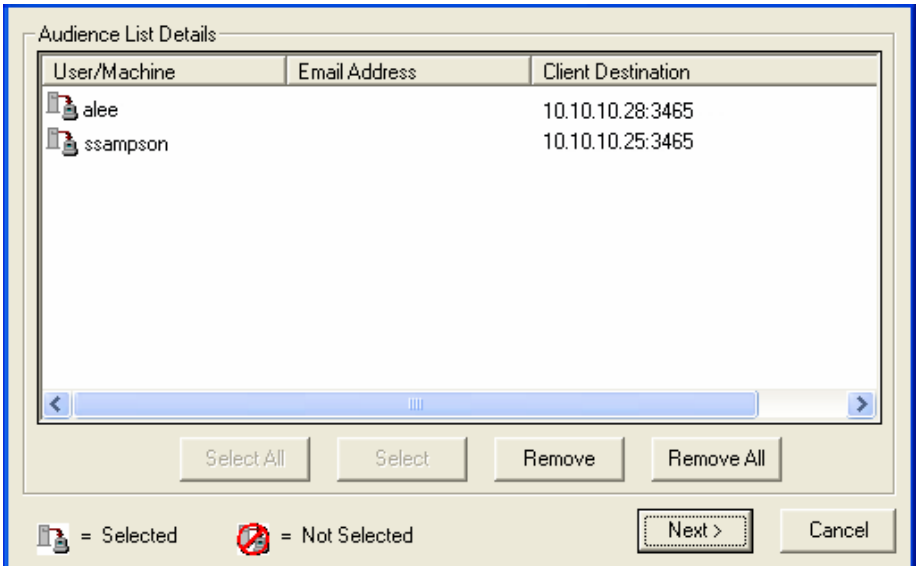
2 Select **Notify Subscribers**.

The Notify retrieves the list of the subscribers from the POLICY Domain. If the selected application does not have any subscribers, the following message appears.

If the selected application does have subscribers, a message appears to confirm that you want to build an audience list.

3 Click **Yes** to confirm that you want to build an audience list.

The CM Notify Manager opens.



A list of the subscribers to the application appears in the Audience List Details area.

— By default, the *entire audience* will be notified.

— To select individual subscribers, click **Remove All**. Then, select the appropriate subscribers and click **Select**.

— To remove an individual subscriber, select the appropriate subscriber, and then click **Remove**.

As shown in the figure above, the symbols to the left of the subscriber indicate who has been selected or not. The total number of agent computers in the audience list and the number of agent computers selected are displayed at the bottom left of the dialog box.

4 Click **Next** when you are done selecting agent computers in the Notify audience list.

5 Select the **Notification Type** for all members of the audience list.

— **Send an Email**
Select this option to inform subscribers of an application's status. See the *HP OpenView Configuration Management Application Self-service Manager Installation and Configuration Guide (CM Application Self-service Manager Guide)* for more information.

The subscriber's e-mail attribute (EMAIL) in the user instance *must* contain a valid entry. Be sure to complete the Subject and Message fields.

— **Update the Application on the target machine(s)**
Select this option to install updates or new versions of an application on the agent computers.

— **Remove the Application on the target machine(s)**
Select this option to remove an application from the agent computers.

Normally, Notify removes the application without requesting permission from the client. This allows removal of applications from unattended agent computers.

If you want to require the subscriber to give permission to remove an application, select the **Prompt for deletion on client** check box.

6 Click **Next** to continue.

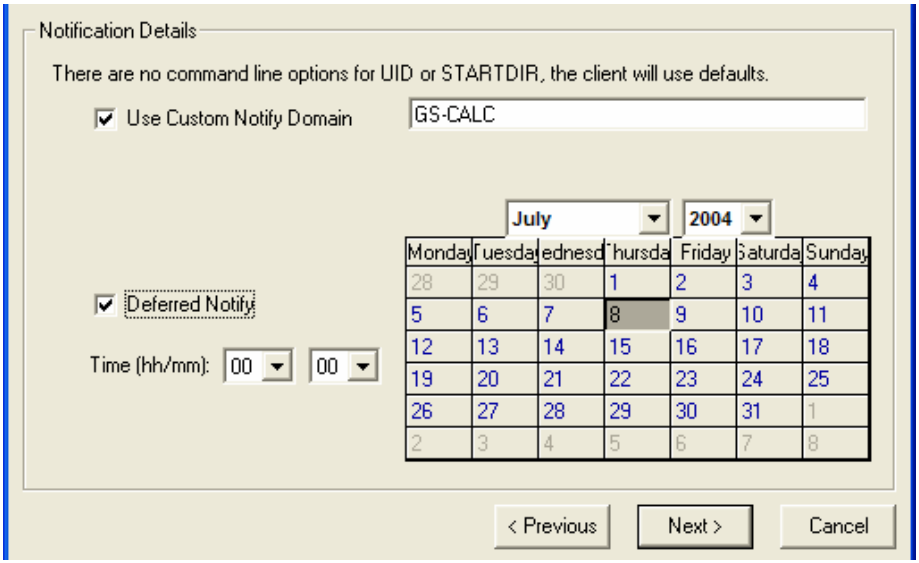
The Notification Details dialog box opens.

By default, the Notify occurs immediately and generates an object in the NOTIFY File in the CM Configuration Server DB. The object is named according to the date and time of the Notify action in the following format: YYYY_MM_DD_HH_MM_SS.

Use this dialog box to create a custom domain to store the object or to defer the notify action to a later date and time.

— Select **Use Custom Notify Domain** and type a name in the text box for the new domain, located in the NOTIFY File, in the CM Configuration Server DB.

— Select **Deferred Notify** and use the Time (hh/mm) drop-down lists and the calendar controls to schedule the Notify.



7 Click **Next**.

The Notification Summary dialog box opens.

8 Click **Finish** to begin the Notify.

A message asks if you want to start the status monitor.

9 Click **Yes** to view the status of the Notify.

A dialog box opens with a list of the subscribers and the status of the notification.

10 Click **Refresh** to update the Status Monitor.

11 Click **Close** when you are done.

Creating a Drag-and-Drop Notify Command

Use a Drag-and-Drop Notify command to initiate a Notify to one or more subscribers immediately. The benefit of this type of Notify is that the application does not have to be installed on the agent computer to perform the Notify. You might use this type of notify to install software, update all mandatory services, or even run an executable on the agent computer.

The Drag-and-Drop Notify works only if the agent computer has connected to the CM Configuration Server prior to the notification. This populates the PROFILE File, which contains the agent computer's network address, used by Notify.

The Drag-and-Drop Notify is intended for use in environments with a single CM Configuration Server. If you are working in an environment with multiple CM Configuration Servers, consider using the Push Manager. Contact your sales representative for details.

In the following example, we will create a command that will update all mandatory services on your agent computers.

To create a Drag-and-Drop Notify

1 Go to **Start** → **Programs** → **CM Administrator** → **CM CSDB Editor**.

The CM Admin CSDB Editor Security Information dialog box opens.

The User ID, as shipped from HP, is RAD_MAST. No password is necessary. This might have been changed during installation. You can also change this by selecting the **Change Password** check box and typing the new password in the New Password and Verify New Password text boxes.

2 If necessary, type a User ID and Password, and then click **OK**. The CM Admin CSDB Editor window opens.

3 Double-click **PRIMARY**.

4 Double-click **SYSTEM**.

5 Double-click **Application Manager (ZCOMMAND)**.

6 Right-click the **Mandatory** Instance, and select **Copy Instance**.

The Copy Instance dialog box opens.

7 Type a display name and name for the instance in the appropriate text boxes. For this example, we named the instances RefreshCatalog.

8 Click **OK**.

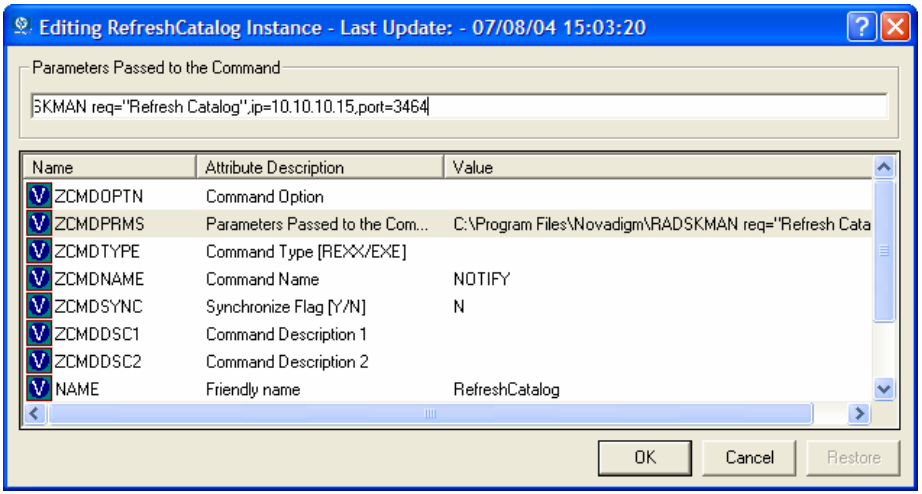
The **RefreshCatalog** Instance appears in the list of ZCOMMAND Class instances.

9 Double-click the instance, such as RefreshCatalog, in the tree view.

The attributes appear in the list view.

10 Double-click the **ZCMDPRMS** attribute.

The Edit Instance dialog box opens.



11 Type the command line that you want to execute on the agent computer. For this example, we will type:

`radskman ip=<mgr_ip>,port=<mgr_port>`

This command line updates or installs *all* new and old mandatory applications. See Specifying the Command Line on page 208 for more information.

To use a Drag-and-Drop Notify to run a command, you must store the program that you want to execute in the `IDMSYS` directory (by default `/opt/HP/CM/Agent`).

12 Click **OK**.

13 Click **Yes** to confirm that you want to save your changes.

14 From the POLICY Domain, select a User, Workgroup, or Department instance and drag it to the RefreshCatalog command. The cursor changes to a wand.

15 Release the mouse button.

The Notify is sent immediately to the specified subscribers and the command line in ZCMDPRMS is executed.

Retrying a Notify

Sometimes a subscriber cannot be notified. This may occur for one of the following reasons:

- The agent computer may be turned off.

- The subscriber does not have a valid e-mail address listed in the CM Configuration Server DB.
- The agent computer is not running the CM Notify service.
- The agent computer may not be accessible via the normal communication channel.

An unsuccessful Notify attempt creates an instance in the RETRY Domain of the NOTIFY File. The RETRY Domain is created the first time a Notify fails.

By default, CM automatically retries the Notify operation for failed attempts. To do this, the CM Configuration Server is started with the Notify Retry Manager (zrtrymgr module), as indicated in the following excerpt from the CM Configuration Server settings file, `EDMPROF.DAT`, located in the bin directory of your CM Configuration Server installation directory (by default, `/opt/HP/CM/ConfigurationServer/bin`).

```
[MGR_ATTACH_LIST]
ATTACH_LIST_SLOTS = 15
RESTART_LIMIT = 7
VERIFY_INTERVAL = 5
CMD_LINE=(zutilmgr) RESTART=YES
CMD_LINE=(zrexxmgr) RESTART=YES
CMD_LINE=(ztcpmgr PORT=3464,NAME=tcpmgr_3464) RESTART=YES
CMD_LINE=(znfytmgr NAME=NotifyManager) RESTART=YES
CMD_LINE=(zrtrymgr) RESTART=YES
```

If you make any changes to the CM Configuration Server Settings file, `EDMPROF.DAT`, you must restart the CM Configuration Server service.

For more information on editing the CM Configuration Server Settings file, see the *HP OpenView Configuration Management Configuration Server User Guide (CM Configuration Server Guide)*.

The Notify Retry Manager periodically examines the NOTIFY File's RETRY Domain, based on VERIFY_INTERVAL in `EDMPROF.DAT`. The default interval

is every five minutes. The Retry Manager attempts the Notify operation for each instance it finds in the RETRY Domain.

Viewing the Results of a Notify

You can find information about a Notify:

- In the CM Admin CSDB Editor in the NOTIFY File.
- In the Status Monitor accessed from the NOTIFY File in the CM Admin CSDB Editor.

You can also find information about a Notify operation in the CM Configuration Server log. The log file is stored on the CM Configuration Server in the `LOG` directory (by default, `/opt/HP/CM/ConfigurationServer/log`).

Viewing an Instance in the NOTIFY File

The NOTIFY File is created after the first Notify is initiated. Each Notify operation creates a single object in the NOTIFY File in the CM Configuration Server DB. The objects are named according to the date and time of the Notify action in the following format: YYYY_MM_DD_HH_MM_SS. In each object, there is an instance for each subscriber that was notified. Each instance contains important information about the subscriber and the notify operation.

To view an instance in the NOTIFY File

1. In the CM Admin CSDB Editor, double-click **NOTIFY**.

   Notice in the image above, there are several default Notify objects. Each object represents a single Notify operation. The objects are named according to the date and time of the Notify action in the following format: YYYY_MM_DD_HH_MM_SS.

   The custom NOTIFY domain, REDBOX, is also a Notify object. However, this Notify was given a custom domain name in the Notification Details dialog box.

   Finally, notice the RETRY Domain. An unsuccessful Notify attempt creates an instance in the RETRY Domain of the NOTIFY File. The RETRY Domain is created the first time a Notify fails.

2. Double-click the Notify object that you want to review.
3. Double-click **NOTIFY**.

The NOTIFY File is divided into domains, where each domain represents one Notify operation. The name of the domain is in the form YYYY_MM_DD_HH_MM_SS, representing the date and time when the Notify operation was initiated.

Each NOTIFY domain has one NOTIFY class. Each NOTIFY class contains an instance for each subscriber that was notified. The instances are named with eight-digit numbers starting with 00000001 and running sequentially up to the total number of notified subscribers.

Each instance contains attributes that identify the subscriber, the kind of Notify operation, and the results of the Notify operation for that subscriber.

4 Review the attributes in the list view. The following table describes possible attributes.

Table 36 Attributes in the NOTIFY instance

Attribute	Description
ZUSERID	The USER, WORKGRP, or DEPT that you notified.
ZCIPADDR	The IP address of the agent computer.
EMAIL	The subscriber's e-mail address, if using e-mail notification.
NTFYTYPE	Indicates the type of notify, such as E for e-mail notification.
NTFYDATE	The date of the Notify.
NTFYTIME	The time of the Notify.
NTFYMSG	Message indicating the status of the Notify, such as "Successfully notified."
NTFYRC	The return code generated for a Notify.
NTFYCMDL	The command line that the Notify executed.
NTFYSUBJ	The subject of the email that is sent, if using e-mail notification.
LOCALUID	The user ID for the subscriber that is currently logged on to the computer.
NTFYRTIM	The time at which the notification should execute.
NTFYRNUM	The number of times to retry the Notify.

Attribute	Description
NTFYDOMN	The name of the domain where this instance is stored.
NTFYINS	The name of this instance.
NTFYPORT	The registered port for Notify.
NTFYPWD	The encrypted password for the Notify.
NTFYUINF	The user information passed to the Notify operation from the Notify Manager.
NTYFYRMAX	The maximum number of times to retry the Notify.
NTYFYDLAY	The amount of time (in seconds) to wait before retrying the Notify.
NTYFYMAC	The physical address of the agent computer. Used for Wake-on-LAN support.
NTYFYMASK	The network mask used for Wake-on-LAN support.

Viewing Results of a Notify or Retry in the Status Monitor

Use the Status Monitor to review the results of a Notify or Retry operation for all of your subscribers.

To see the status of a Notify or Retry

1. In the CM Admin CSDB Editor, double-click **NOTIFY**.
2. Right-click the appropriate Notify domain or the **RETRY Domain** for which you want to see the status.
3. Click **Status Delete** to delete the status information.

   or

   Click **Status Display** to display the Status Monitor.
4. Click **Refresh** if you think the status might have changed.

   or

   Click **Close** to close the Status Monitor.

CM Self Maintenance

Maintenance for the CM agents is available from Technical Support. The maintenance will include import decks for the CM Configuration Server DB. New instances are created in the PRDMAINT class in the PRDMAINT Domain. There will be one PRDMAINT instance for each PRODUCT_PLATFORM_RELEASE combination. These instances will be connected based on the agent's platform and current product level. Once you have decided to roll out the maintenance to the agent computers, you can add the service to the user's entitlements.

To minimize the need for separate PRDMAINT bundles based on different operating systems where the actual maintenance is the same, the ZMASTER.ZOSTYPE variables identify the operating system type or family. The valid values for this variable are:

- WIN32_NT (including Windows 2000, XP, and 2003)
- WINX64_NT
- WINIA64_NT
- UNIXAIX
- UNIXHPUX
- UNIXLNUX
- UNIXSOL
- UNIXSX86

Usage Notes

- All packages are disabled by default. This is accomplished by setting a ZSTOP expression to "1" to prevent deployment. Either remove this value for general deployment, or use this ZSTOP expression to restrict its deployment to certain groups.
- The first REQUIRES connection is reserved for any possible hot fix, a fix sent to you directly by Technical Support is not yet available in a fix or service pack. This package, _HOTFIX, will be used to chain any required fixes (and/or enhancements) and will be maintained by the customer. The second connection is for any locally customized code to be included as part of maintenance.

- Use the ACTMAINT attribute in the SETTINGS Class of the CLIENT Domain to specify how you want maintenance processed. You can choose to immediately download and install maintenance (I), download only and install later (D), or prompt users to install maintenance at another time (D). Maintenance runs *only* when the mnt parameter of radskman is set to Y. See ACTMAINT on page 137 and the mnt parameter on page 192 for more information.

We will provide an updated PRDMAINT instance with each new maintenance pack. The customer is not required to apply all maintenance.

To deploy agent maintenance packages

1. A maintenance package is made available on the HP OpenView web site in the form of an export deck.
2. Download the files. There should be at least an `xpi` and `xpr` file.
3. Stop the CM Configuration Server service and copy the export files to the CM Configuration Server `bin` directory.
4. Import the files using the ZEDMAMS utility. For detailed information on the use of this utility, see the *CM Configuration Server Guide*.

   For example, if you were given two files, `MAINT_RAM_40_RC3.XPI` and `MAINT_RAM_40_RC3.XPR`, you might use the following two command lines.

   ```
   ZEDMAMS VERB=IMPORT_INSTANCE,FILE=
   MAINT_RAM_40_RC3.XPI,PREVIEW=NO

   ZEDMAMS VERB=IMPORT_RESOURCE,FILE=
   MAINT_RAM_40_RC3.XPR,PREVIEW=NO
   ```

   Your command line may vary depending on a number of factors. For detailed information on the use of this utility, see the *CM Configuration Server Guide.*
5. Restart the CM Configuration Server.
6. Assign the Maintenance Service to the appropriate users in the POLICY Domain.

   To run the maintenance portion of an agent connect process, the `mnt` parameter of the radskman command line, must be set to Y.

During catalog processing, the agent will first process all services found in the PRDMAINT Domain, perform arbitration to determine appropriate maintenance, and deploy the maintenance to the maintenance staging directory.

About CM Proxy Servers

Use CM Proxy Servers to load a portion of the work required to deploy applications from the CM Configuration Server to another server computer. You may want to do this for the following reasons:

- The CM Proxy Server may be closer to the clients on the network.
- You may want to reduce the load on the CM Configuration Server.

When using CM Proxy Servers, the software to be distributed is copied to the CM Proxy Server. The CM Proxy Server then provides the software to those CM agents that are not required to obtain their software from the CM Configuration Server. The potential benefit of CM Proxy Servers must be evaluated individually for each server and its CM subscriber computers.

For more information, refer to the *HP OpenView Configuration Management Proxy Server Installation and Configuration Guide (CM Proxy Server Guide).*

Summary

- Carefully plan and test your application deployment strategy to determine the best distribution method for your subscribers.
- Use the Scheduler service to deploy an application at a specific time or interval.
- Use the Notify function to update or remove an application that has already been deployed, or to notify users via e-mail of an update.
- Consider if you have any special cases for deployment that may need further configuration.
- Use the Version Group Editor when you have multiple versions of the same application. You can use the Version Group Editor to schedule deployments, and set versions to activate.
- Consider using CM Staging Servers to minimize network traffic or to minimize work on the CM Configuration Server.

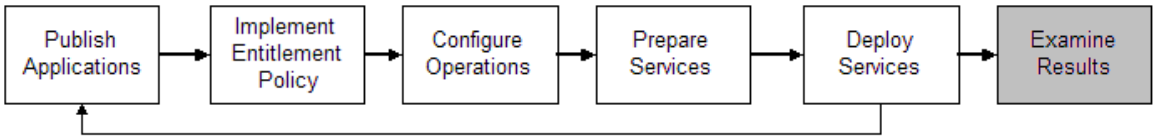
10 CM Agent Objects and Directories

At the end of this chapter, you will:

- Know the directory structure of the CM agent.
- Be familiar with core CM agent objects.
- Know where client objects are stored.
- Know how Open Database Connectivity (ODBC) can help you generate reports with information from the objects.

Although you will tailor this strategy to meet your organization's needs, we recommend that you review this guide for a comprehensive understanding of the CM Application Self-Service Manager. This chapter covers CM agent objects.

Figure 18 Tasks completed in this guide



CM Agent Directory Structure

Below is a table of the directory structure on an agent computer following a forced system connection from the client to the CM Configuration Server, and the installation of a mandatory application. (Directories in Table 37 below are preceded with `/opt/HP/CM/Agent/` by default).

Table 37 Agent directories

Directory (/opt/HP/CM/Agent/..)	Description
/opt/HP/CM/Agent	Agent Directory (IDMSYS)
../CACertificates	SSL Certificates
../lib	LIB Directory (IDMROOT)
../lib/MAINT	Maintenance Storage Directory
../lib/BACKUP	Upgrade Maintenance Backup folder.
../lib/data	Data Storage (IDMDATA)
../lib/SYSTEM	Starting Directory (startdir) created during connect. (Name will vary).
../lib/SYSTEM/CM-CS	CM Configuration Server name (mname).
../lib/SYSTEM/CM-CS/SOFTWARE	Directory Name (dname)

Directory (/opt/HP/CM/Agent/..)	Description
../lib/SYSTEM/CM-CS/SOFTWARE/ZSERVICE	ZSERVICE Class
../lib/SYSTEM/CM-CS/SOFTWARE/ZSERVICE/DRAGVIEW	Sample application directory
../log	Log directory (IDMLOG)

The term **computer** is used to refer to a workstation or server.

CM Application Manager Directories

The initialization settings for the CM Application Manager are located in the `.nvdrc` file on the agent computer. This is located, by default, in the home directory of the account used to install the agent.

Table 38 NOVAEDM parameters

Parameter	Description
IDMDATA	When CM installs software, the CM agent temporarily stores compressed files received from the CM Configuration Server in this folder. Once the files are decompressed and installed on the agent computer, the compressed files are erased. Default: `/opt/HP/CM/Agent/lib/Data/`
IDMLIB	Dynamic directory that stores the objects for the service currently being managed. Default: `/opt/HP/CM/Agent/lib/`
IDMSYS	Stores the CM agent executables, such as `.EXE` and `.DLL` files. Default: `/opt/HP/CM/Agent/`
IDMROOT	The base directory for `IDMLIB`. This is a static path. Default: `/opt/HP/CM/Agent/lib/`

Parameter	Description
IDMLOG	Stores the CM agent logs. Default: `/opt/HP/CM/Agent/log/`

About CM Agent Objects

When an agent computer connects to the CM Configuration Server, information is exchanged between the agent and the CM Configuration Server. This exchange is called **resolution**. During resolution, CM checks the status of services, and updates the CM Configuration Server with information from objects stored on the agent computer.

CM agent objects are stored in the `IDMLIB` directory on the agent computer. After installing the CM Application Self-service Manager agent and connecting to the CM Configuration Server, you can use CM agent objects to answer questions such as:

- What is the hardware configuration of the agent computer?
- Was the service successfully installed?
- When was the service installed?
- What is the agent computer's name, and who was the last user logged on?
- What are the possible data sources for this agent computer?

While there are multiple CM objects on an agent computer at any time, there is a core group of objects that supply information about and the status of the current agent connect. Table 39 on page 233 lists these core objects. The table includes information on when the object is created or updated, and a brief summary of what the object includes. Each object listed has its own section in this chapter including a table listing its attributes. There are other objects created during the agent connect, but only the most commonly used ones are noted here. Check the HP OpenView web site for information on other agent objects.

Table 39 Core Client Objects

Object	When created or updated	Type of Information included
ZCONFIG	ZCONFIG is created at start of agent connect process. See Table 40 on page 236 for more information.	Contains basic hardware information for the agent computer such as processor, operating system, and drives.
SYNOPSIS	This object is transferred to the CM Configuration Server at the end of the agent connect. Note: Client Operations Profiles must be enabled for this object to be present. See Table 41 on page 237 for more information.	RADSKMAN stores a job summary in the SYNOPSIS object. It reports some of the parameters from the RADSKMAN command line and information on the number of files and bytes added, removed, and repaired.
SAPSTATS	Updated by any network bound modules that need to access the Server Access Profile (SAP) such as RADCONCT, RADSTGRQ, and RADSTGMS. RADSKMAN deletes the SAPSTATS object at the beginning of the job. Note: Client Operations Profiles must be enabled for this object to be present. See Table 42 on page 239 for more information.	The SAPSTATS object has one instance for each of the agent computer's Server Access Profiles (SAP). It summarizes information for each SAP such as speed, number of files sent and received, and the role of the SAP.

Object	When created or updated	Type of Information included
PREFACE	PREFACE is sent to the CM Configuration Server at every phase of a radskman process including: • Client operations profile resolution • Self Maintenance resolution • Catalog resolution • Single service resolution (This can happen multiple times depending on what services are processed.) • Outbox flush See Table 43 on page 241 for more information.	Contains core information about each invocation of radskman.
SMINFO	SMINFO is created during Client Operations Profiles resolution, but does not require Client Operations Profiles. See Table 44 on page 242 for more information.	SMINFO collects information that is independent of the hardware and software installed on the computer, and some network information.

CM Agent Version

Some of the objects described in this book apply only to CM agents version 3.1 and above. To verify the agent's versions, open the `connect.log` file in the `IDMLOG` directory on any operating system using a text editor and search for "version." On Windows Operating systems, you can also check the Version tab of the Properties of radskman in the `IDMSYS` directory.

Using the CM Admin Agent Explorer to View Objects

The CM Admin Agent Explorer is installed as a component of the HP OpenView Configuration Management Administrator (CM Administrator).

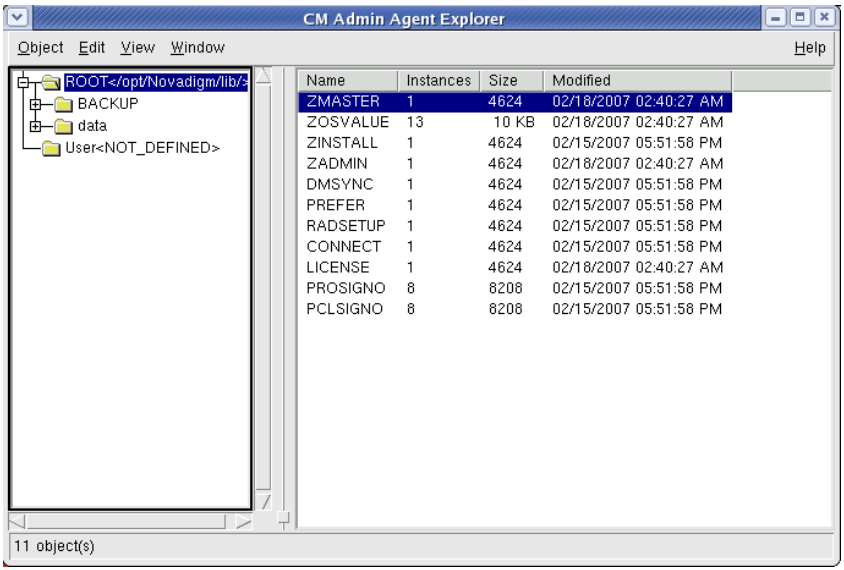
Use it to view objects in the IDMLIB directory. You can view any object if you have access to the agent computer's IDMLIB directory. Otherwise, you may need to manually retrieve the object file, and store it on your administrator computer.

To view an object using the CM Admin Agent Explorer

1 Go to the directory where you installed the CM Administrator.

2 Type **./radobjed** and press **Enter**.

The CM Admin Agent Explorer opens.



3 Navigate to the agent computer's IDMLIB directory or to the directory where the object is stored.

4 Double-click the object's name in the list view.

The CM Agent Explorer displays the selected object.

5 Click **Save/Exit** to close the dialog box.

Hardware Configuration Information (ZCONFIG)

The ZCONFIG object stores hardware configuration information from the agent computer. Use the CM Agent Explorer to view the ZCONFIG object. Table 40 on page 236 describes the attributes of ZCONFIG arranged in

alphabetical order. These attributes may vary depending on the configuration of the agent computer.

The ZCONFIG object is sent to the CM Configuration Server automatically for viewing with the CM Admin CSDB Editor. If you do not want this object sent to the CM Configuration Server, set the POLICY.USER.ZCONFIG attribute to N in the base instance. This will stop collection from ALL users. The object will still exist on the agent computer.

Table 40 ZCONFIG Attributes

Attribute	Description	Example
RUNLEVEL	Current run level at time of scan	5
ZHDWCOMP	Computer Name	qalinux
ZHDWCPU	CPU type	I686
ZHDWD00	Drive name for drive 00	/dev/hda2
ZHDWD00F	Current free space on drive 00	26913026048
ZHDWD00M	Mount Point for Drive 00	/
ZHDWD00T	Total space for drive 00	35152932864
ZHDWDNUM	Number of drive letters assigned	3
ZHDWMACH	Machine Type	i686
ZHDWMEM	Total physical memory (RAM)	133,619,712
ZHDWOS	Operating system	Linux
ZHDWSVCP	Service pack applied	2.4.20-8
ZHDWXHID	Host ID (output of hosted command)	771039E4
ZHDWXHN	Host Name	qalinux
ZOBJNAME	Name of Object	HARDWARE_SCAN
ZOSMAJOR	Major Component of OS version	2
ZOSMINOR	Minor Component of OS version	4
ZOSREV	OS revision (output of uname –v)	#1 Thu Mar 13 17:54:28 EST 2003
ZOSVER	OS version (output of uname –r)	2.4.20-8

Attribute	Description	Example
ZUSERID	User ID or computer name	LINUXUSER

Client Operations Profile Summary (SYNOPSIS)

The SYNOPSIS object is created on clients using Client Operations Profiles. The SYNOPSIS object summarizes the most recent agent connect. Use the SYNOPSIS object to confirm the success or failures of the agent connect process. See Chapter 7, Configuring CM Client Operations Profiles for more information.

Table 41 SYNOPSIS Attributes

Attribute	Description
STARTIME	Start time in ISO8601 time format, e.g., 1997-08-15T11:12:00-0400
ENDTIME	End time in ISO8601 time format
EXITCODE	Exit code from the job
ERRORMSG	Text message corresponding to the EXITCODE described in the *CM Management Applications Messages and Codes Guide*.
PRIORAPP	Total number of applications that existed in the service list (installed/not installed) before this job started
PRIORINS	Total number of installed applications that existed in the service list before this job was started
PRIORERR	Total number of applications in the service list that have errors before this job started
CURRAPP	Number of applications in the service list after the job completed
CURRINS	Number of applications in the service list that have been installed
UPDNUM	Number of updates found in the service list
UPDSKIP	Number of updates skipped
UPDDONE	Number of updates processed
UPDFAIL	Number of updated that failed

Attribute	Description
ADDNUM	Number of new applications found in the service list
ADDSKIP	Number of installs skipped (possibly optional applications)
ADDDONE	Number of installs processed
ADDFAIL	Number of installs that failed
DELNUM	Number of deletes found in the service list
DELSKIP	Number of deletes skipped
DELDONE	Number of deletes processed
DELFAIL	Number of deletes that failed
VERNUM	Number of applications that were verified
VERSKIP	Number of verifications skipped
VERDONE	Number of verifications processed
VERFAIL	Number of verifications that failed
REPNUM	Number of applications that were repaired
REPSKIP	Number of repairs skipped
REPDONE	Number of repairs processed
REPFAIL	Number of repairs that failed
CREFRESH	Catalog Refreshed (Y/N)
JOBID	Jobid passed in on the command line via notify
ZUSERID	Userid for this job
ZCONTEXT	Context of this job (M – Machine or U – User)
MACHNAME	Machine name of the agent computer from where this was run
USEREXEC	User that executed the job
CMDLINE	Command line parameters used to execute this job

Service Access Profile Status (SAPSTATS)

The SAPSTATS object is generated on agents using Client Operations Profiles, and is used to report the Server Access Profile (SAP) status and usage statistics from the CM agent. The SAPSTATS object contains all the variables defined in the SAP Class in the CM Configuration Server DB along with the following usage related variables. For more information on the SAP Class, see Chapter 7, Configuring CM Client Operations Profiles for more information.

Table 42 SAPSTATS object attributes

Attribute	Description
BANDWDTH	Percentage of bandwidth to use between 1 and 99.
BYTERCVD	Bytes received
BYTESENT	Bytes sent
ENABLED	Specifies if this SAP is enabled. Y for enabled, N for disabled.
ERRCOUNT	Number of errors
FILEMISS	Number of files not found
FILERCVD	Number of files received
FILESENT	Number of files sent
LASTAXSD	Last Date/Time Accessed in ISO format
NAME	Friendly name of the SAP
OBJRCVD	Number of objects received
OBJSEND	Number of objects sent
PRIORITY	Priority for this SAP obtained from the CLIENT.LOCATION Class instance
PROXY	The internet proxy URI through which the agent will connect to the SAP. Maintained by agent.

Attribute	Description
ROLE	Role of the SAP. Possible values are: **O**: Client **O**perations Profiles **M**: Self **M**aintenance **S**: **S**ervices **D**: **D**ata **A**: **A**ll of the above roles
SPEED	Speed to the SAP from the agent computer measured in Bytes per second
STATUS	Status of this SAP 000= SAP was accessed successfully 920 = SAP could not be accessed 999 = SAP was not used
STREAM	Specifies if streaming is used. Y for enabled. This overrides the client setting in ZMASTER.ZNORSPNS.
THROTYPE	Type of bandwidth throttling used. Possible values are NONE, ADAPTIVE, and RESERVED.
TIMEOUT	Communications timeout in seconds.
TYPE	Type of SAP. Possible values are: RCS – CM Configuration Server DATA – CM Proxy Servers, Staging Servers or a CD-ROM.
URI	Universal Resource Identifier for the SAP

Radskman Execution (PREFACE)

The PREFACE object contains information about each execution of radskman. PREFACE is sent to the CM Configuration Server at every phase of a radskman process including:

- Client Operations Profile resolution
- Self Maintenance resolution
- Catalog resolution

- Single service resolution (This can happen multiple times depending on what services are processed.)
- Outbox flush

At each new phase of the agent connect, the PREFACE object is updated. The variables in the PREFACE object can be used for resolution and reporting. For resolution, use the attributes of the PREFACE object for ZSTOP expressions, for symbolic substitution, and for dispatching messages. For reporting, you can combine MACHNAME, ZUSERID, ZCONTEXT, JOBID, and CTYPE to know which user ran the agent connect, the type of connect, and the context.

Table 43 PREFACE Object Attributes

Attribute	Description
CMDLINE	The radskman command line parameters used for the current agent connect.
COMPDN	The distinguished name of the computer in the Active Directory format. This field will be blank if the system is not part of an Active Directory or a Domain environment. Windows operating systems that do not authenticate to Active Directory would show this as their DomainName/MachineName. Example: CN=ALEE,CN=Computers,DC=usa, DC=asdfoods, DC=com
CTYPE	Type of CM agent. The possible values are: RSM CM Application Self-service Manager RAM CM Application Manager RPS CM Proxy Server or CM Staging Server (for preloading application resources)
JOBID	The jobid specified on the command line for this connect.
LOCALUID	The starting directory under `IDMROOT` on the agent computer. LOCALUID contains the value derived from the STARTDIR radskman parameter. For example, if startdir = $USER then LOCALUID would contain the user's ID. If STARTDIR = SYSTEM then LOCALUID would contain SYSTEM. UID stands for user's initial directory *not* the user's identification.
MACHNAME	Agent computer's machine name.

Attribute	Description
USEREXEC	The user who is currently logged on and who executed the command. For Notify and Timers, this would be SYSTEM. For logon scripts, this would be the subscriber's network account name.
ZCONTEXT	The context for this connect. Possible values are: M – Machine, U – User, or blank.
ZDOMNAME	The CM Configuration Server DB's domain specified in the DNAME parameter of the radskman command line. If DNAME is not specified in the command line, the default is SOFTWARE.
ZMGRNAME	The CM Configuration Server's name specified in the MNAME parameter of the radskman command line.
ZUSERID	The ZUSERID field contains the same value found in ZMASTER.ZUSERID of the client. In most scenarios, this represents the machine name of the agent computer, but may also contain the current user name or another value. The value found in this field is often used as the key for policy resolution or reporting. The UID radskman parameter sets this value.

Systems Management Information (SMINFO)

The Systems Management Information (SMINFO) object is created on all AM agent computers. The SMINFO object summarizes hardware specific information that is independent of what operating system or software is installed on the agent computer. CM uses SMBIOS standards to access data about the BIOS. SMINFO also includes some network and user ID information.

Table 44 SMINFO Attributes

Attribute	Description
ASSETTAG	Unique Asset Tag number of the agent computer from the BIOS
COMPDOMN	Computer Domain
COMPNAME	Computer Name

Attribute	Description
IPADDR	Agent computer's IP address
MACADDR	Agent computer's MAC address
MACHUUID	Unique machine user ID
SNENCLOS	Serial Numbers for the system enclose structures from the BIOS
SNSYSTEM	Serial Numbers for the system structures from the BIOS
SUBMASK	Subnet Mask
SUBNET	Subnet
SYSMANUF	System manufacturer from the BIOS
SYSPROD	System manufacturer product information from the BIOS

Controlling Default Permissions for Directories and Objects

Directories, objects, and log files created by CM are assigned permissions based on current umask settings and execute permissions on objects and log files are removed. In order to change the default permissions assigned when new directories, objects, and log files are created by CM within `IDMLIB`, you can use environment variables or you can create a `DEFAULTS.EDM` file in `IDMROOT`.

Note that environment variables will always take precedence. If the environment variables are set and a `DEFAULTS.EDM` file exists, values defined using the environment variables are used.

These methods for controlling permissions apply only to newly created, service-related directories and objects within `IDMLIB`. For example, `/opt/OVCM/lib/SYSTEM/CMCS/SOFTWARE/ZSERVICE/SAMP_APP/00000000.000`.

To control permissions using environment variables

- Set the following environment variables with the permissions you want assigned by default:
  - For directories: IDMLIBPERM
  - For objects: IDMOBJPERM
  - For log files: IDMLOGPERM

To control permissions using DEFAULTS.EDM.

1. Within `IDMROOT`, create an object, `DEFAULTS.EDM`.
2. Add the following variables with the permission value to be used when new objects, log files, or directories are created by CM.
   - For directories: LIBPERM
   - For objects: OBJPERM
   - For log files: LOGPERM

   For example, to exclude write permissions for objects for group and other, create a `DEFAULTS.EDM` file with the following:

   ```
   OBJPERM     0644
   ```

   To exclude write permissions for logs for group and other, create a `DEFAULTS.EDM` file with the following:

   ```
   LOGPERM     0644
   ```

   To set the default permissions of directories to read and write for everybody, create a `DEFAULTS.EDM` file with the following:

   ```
   LIBPERM     0777
   ```

The PROFILE File

Some agent objects such as ZCONFIG and ZMASTER are sent to the CM Configuration Server during an agent connect. The objects received from the agent computer are stored in the PROFILE File in the CM Configuration Server DB. Within the PROFILE File, each agent computer is stored as a domain. By default, each agent computer is identified by the subscriber who is currently logged on. The subscriber may be either a computer name or a user name.

The following table describes some of the objects that you might find in the PROFILE File, although the objects may vary based on your configuration.

Table 45 Objects in the PROFILE File

Instance	Information Recorded
ZCONFIG	Contains basic hardware information for the agent computer such as processor, operating system, and drives.

Instance	Information Recorded
ZMASTER	Contains information used to run the CM agents, such as user ID and operating system.
ZSVCSTAT	Contains information about the service after it has been successfully installed on the agent computer. This is useful for reporting purposes such as determining which users have the application or when it was installed. One instance is created for each service.
ZSTATUS	Contains information about the most recent agent connect, such as the number of objects going to and from the agent computer.

Each domain contains several classes, which represent the objects received from the agent computer. Use the CM Admin CSDB Editor to view the PROFILE File.

Reporting

CM Reporting Server

As part of the CM extended infrastructure, the web-based CM Reporting Server allows you to query the combined data in existing CM Inventory Manager, CM Patch Manager, and CM Usage Manager databases and create detailed reports. In addition, you have the option of mounting an existing LDAP directory, which allows you to filter your data using your LDAP directory levels.

The CM Reporting Server interface provides a dynamic and intuitive way to use CM SQL data for reporting and overall environmental assessment.

For additional information refer to the *HP OpenView Configuration Management Reporting Server Installation and Configuration Guide*.

CM Messaging Server

The CM Messaging Server is a generic messaging service that can be used with many CM Infrastructure modules. Its job is to continually monitor a predefined data queue and dynamically route data objects to one or more

external destinations. The CM Messaging Server provides retry, rerouting, and failover capabilities to ensure all data is transferred efficiently and reliably.

On a CM Configuration Server, the CM Messaging Server operates with the QMSG executable to handle the transfer of reporting data obtained from clients to the appropriate external databases.

For additional information refer to the *HP OpenView Configuration Management Messaging Server Installation and Configuration Guide*.

Agent Logs

One log is created from the three main agent modules, radskman, radpinit, and radconct. The default name for the log is `connect.log`, and its default location is /opt/HP/CM/Agent/log. When `connect.log` reaches a size of 1 MB, a backup is created called `connect.bak`.

For diagnosing problems, you may want to name a new log for debugging deployments or for collecting information from your agent computers. Each of the three main agent modules can be directed to use a particular log file, by adding the log parameter to its command line. For example, you might name your logs based on the date and time. Then, if you notice a problem occurring on a certain date, you can retrieve only the logs that you need to review. Add the log parameter to a radskman command line to specify a particular log name as shown below:

```
radskman log=notify10012003.log
```

The value for the log parameter is stored in the LOGNAME attribute, located in the ZMASTER object in the catalog and application directories.

Diagnostic Module (radstate)

Radstate is a diagnostic module designed to give an overview of the current state of the CM agent. The information in the radstate output is based on data retrieved from numerous CM agent objects.

When radstate is run with the Verbose parameter (mode **v**), it provides basic information regarding the CM agent environment:

- Global object statistics

- Current date and time
- Current operating system
- Locations of the `IDMSYS`, `IDMLIB`, and `IDMLOG` directories
- Environment settings
- Emulator settings
- Trace levels
- Timeout settings
- All Service status including component totals by User and Service including instance totals and byte totals.
- Agent Timer Information

Use radstate at anytime to check CM agent configurations. For example, run radstate at the end of each agent connect. After radstate is run using mode option o, the ZRSTATE and ZRSTATES objects are built and can be sent to the CM Configuration Server as needed.

Radstate should be run:

- Whenever CM agent-specific information is required.
- If it is suspected that some files may not have deployed correctly.
- If desktop updates have not occurred.

Manual execution of radstate produces a summary style report, `radstate.log`, regarding the current state of services and resources installed on the client desktop. Radstate is executed from the command line using the appropriate parameters, separated by a comma, for example:

```
radstate mode=vo, IDMROOT=/opt/HP/CM/Agent/lib
```

See the HP OpenView web site for a technical note with additional information.

Summary

- The ZCONFIG object stores hardware configuration information from the agent computer.
- Use the CM Admin Agent Explorer to view objects stored on the agent computer.
- The PROFILE File in the CM Configuration Server DB stores objects received from the agent computer.

A Naming Conventions

This appendix discusses the use of naming conventions to help you organize the software stored in the CM Configuration Server DB.

When publishing applications, subscribers may have different requirements such as:

- Different operating systems.
- Different amounts of free space on their hard drives.
- Different processors, memory, and so on.
- Different data or applications, depending upon their job function or other factors.

Due to these varying requirements, you might need to create several packages for a single application. To keep your data organized in the CM Configuration Server DB, we recommend that you create a naming convention to be used within your organization.

This section provides some recommendations that you can use as a starting point to create your own standards.

Categorizing Information

In general, consider using unique high-level identifiers with an underscore (_) to categorize information in the CM Configuration Server DB. HP OpenView Configuration Management Administrator Configuration Server Database Editor (CM Admin CSDB Editor) groups instances based on the identifier that precedes the underscore.

If you decide to use a high-level identifier *without* an underscore (_), you can use the CM Admin CSDB Editor's filtering capabilities to display only the instances with that identifier.

See the *HP Open View Configuration Management Administrator Configuration Server Database Editor Guide (CM Admin CSDB Editor Guide)* for more information.

For example, if you had a Windows 2000/XP version and a Windows Server 2003 version of an application to calculate loan amortizations, you might name the packages AMORTIZE_2000/XP and AMORTIZE_2003.

Naming Conventions for the POLICY Domain

We recommend that you use a variation of the following standards.

Table 46 Naming conventions for the USER Class

Format	Description	Example
USERID	Identifies the subscriber.	SJones

When naming instances in a workgroup, use information that groups your subscribers appropriately. For example, if your company is organized by division and location, you might use conventions such as the following:

Table 47 Naming conventions for the WORKGRP Class

Format	Description	Example
DIV_LOC_DESC	Defines ownership or assignment.	CTS_CLE_EVERYONE
DIV	Identifies the division.	CTS (Corporate Technology Services)
LOC	Identifies the location.	CLE (Cleveland)
DESC	Provides additional description of the group.	EVERYONE (all users)

Naming Conventions for the SOFTWARE Domain

In a company organized by division and location, you might organize your data using the following standards.

Table 48 Naming Conventions for the PACKAGE Class

Format	Description	Example
DIV_LOC_APPNAME _VER_OS	Defines the application.	CTS_CLE_WINZIP_ 80_WNT
DIV	Identifies the division.	CTS (Corporate Technology Services)
LOC	Identifies the location.	CLE (Cleveland)
APPNAME	Identifies the application.	WINZIP
VER	Identifies the version of the application.	80
OS	Identifies the operating system that the application runs on.	WNT

Table 49 Naming conventions for Delivery and Auditing Classes*

*All other classes in the SOFTWARE Domain.

Format	Description	Example
REG_DIV_LOC_ APPNAME _VER_OS	Defines the application.	NAM_CTS_CLE_WIN ZIP
REG	Identifies the region.	NAM (North America)
DIV	Identifies the division.	CTS (Corporate Technology Services)
LOC	Identifies the location.	CLE (Cleveland)
APPNAME	Identifies the application.	WINZIP

Determining the conventions that make sense for your organization may take some time. However, creating a convention up front and communicating it to all of your CM administrators will keep you organized in the future.

B Adding Attributes to the CM Configuration Server DB

The following procedure shows you how to add an attribute (also known as a variable) to your CM Configuration Server DB.

Be sure to create a backup of your CM Configuration Server DB before adding an attribute to it.

At a minimum, you will need the following information before you make the changes to your CM Configuration Server DB:

- The name of the class that you are editing.
- The name of the new attribute.
- The length of the new attribute.
- A description for the new attribute.

To add an attribute to a class template

1. Go to **Start** → **Programs** → **HP OVCM Administrator** → **CM Admin CSDB Editor**. The CM Admin CSDB Editor Security Information dialog box opens.
2. If necessary, type a User ID and Password, and then click **OK**.

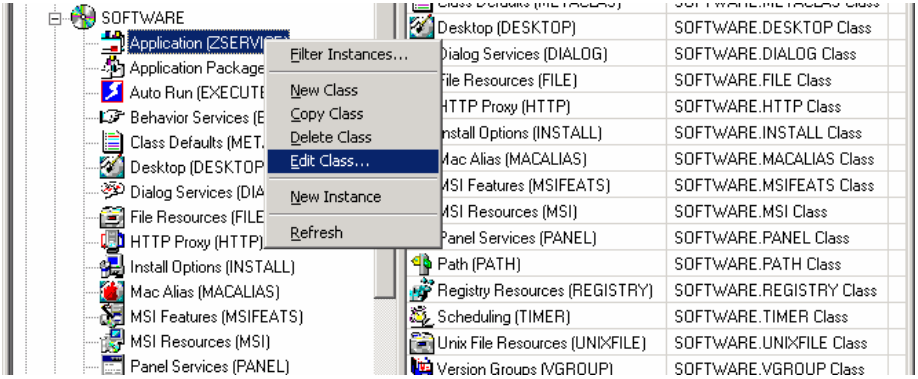
   The User ID, as shipped, is RAD_MAST. No password is necessary. This may have changed in your installation. Check with your CM security administrator to obtain your own User ID and Password, if necessary.

   The CM Admin CSDB Editor window opens.
3. Navigate to the class that you want to edit. For example, you might go to PRIMARY.SOFTWARE.ZSERVICE.

If you do not see the name of the class (such as ZSERVICE) in the tree view, you can modify the CM Admin CSDB Editor options. To do this:

On the CM Admin CSDB Editor tool bar, click **View**, and select **Options**. In the Options dialog box, click the **General** tab, and then select the **Show Class Names Next to Descriptions** check box.

4 Right-click the class that you want to edit, such as Application (ZSERVICE).



5 Select **Edit Class**. The Editing Class dialog box opens. For detailed information about the Editing Class dialog box, refer to the *CM Admin CSDB Editor Guide*.

6 Determine where, in the Attribute List, the attribute should be inserted.

7 In the list of attributes, select the attribute adjacent to where you want to insert the new attribute.

If the Automatic Sequencing check box is not selected, attributes are processed during resolution in the order in which they appear in the Attribute List.

If the Automatic Sequencing check box is selected, the attributes of the class are processed in the following order: Expressions, Attributes, Classes (Connections), and then Methods.

Refer to the *CM Admin CSDB Editor Guide* for more information.

8 Click **Insert Before** to insert the attribute before the selected one.

or

Click **Add After** to add the new attribute after the selected one.

A blank attribute appears.

9 In the Name field, type the name of the new attribute.

10 In the Length field, type the length for the attribute.

11 In the Description field, type a description for the attribute.

12 In the Type drop-down list, select **Attribute**.

13 If the document that contains the information about the new attribute specifies CM Agent or Configuration Server properties, select the appropriate check boxes.

14 Click **OK**.

15 Click **Yes** to confirm the changes to the class.

C 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 50 below will help you identify naming changes that have been applied to the Radia brand.

Table 50 Product Name and Term Changes

New Name/Term	Old Name/Term
CM A gent Installation Wizard	Radia Client Installation Wizard
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 Solutions for Servers	Server Management

Glossary

Active Component Server

See CM Configuration Server.

Administrative Installation Point (AIP)

An AIP is a server share or local directory structure that contains all of the files needed to run setup for a Windows Installer-enabled application.

agent computer

An agent computer is a subscriber's computer that has the CM agent software installed on it.

agent object

An agent object is a file located on the client computer that contains information about the configuration of services or hardware.

APPEVENT

APPEVENT is the client object that provides information about an application event, such as success or failure of the installation.

Application Manager

See CM Application Manager.

applications

Also called software, data, or services.

Applications are one type of content that CM can manage on subscriber computers. Use the CM Admin Publisher to create packages of data to be managed on your subscribers' computers.

attribute

Also called *field, variable,* or *property*.

An attribute is a single, descriptive data item in a class. The class template contains a definition (e.g., the name, data type, description, and length) for each attribute that makes up the class. Class instances contain a set of attributes and each attribute contains a value.

attribute property

An attribute property controls some aspect of how an attribute is processed on the CM Configuration Server and client computer. Each attribute defined in a class template has a set of CM Configuration Server properties and a set of client properties.

audience list

An audience list is a directory of the subscribers for an application used by CM Notify.

base instance

The base instance contains the default values for the attributes that make up a class. When you create a new instance in that class, the attributes in the new instance inherit the default values, as specified in the base instance.

byte-level differencing

Byte-level differencing is the process of publishing a patch containing updates or corrections to a resource. The patch is calculated by differencing an existing copy of the resources in the CM Configuration Server DB against the resources currently being published.

class

A class defines a category of the distribution model to be managed. It is conceptually similar to a schema in a relational database structure or a file layout in a traditional flat file. Each of the required elements of a distribution model (e.g., users, applications, etc.) is defined in the CM Configuration Server DB by its class.

class connection variable

A class connection variable determines the path of resolution for a client's distribution model during the Client Connect process. It is a branch in the resolution process.

A class connection is resolved and resolution continues using the target instance identified in the class connection variable if the class connection variable attribute's name is _ALWAYS_, INCLUDES, REQUIRES, or if the name of the attribute matches the current value of the system message.

class instance

Also called *instance*.

A class instance is an object in the CM Configuration Server DB that contains a specific occurrence of a class. This is analogous to a row in a relational data table or a record in a traditional flat file.

clean computer

A clean computer is a computer on which the operating system has just been installed, and no further changes have been made.

Client Explorer

See CM Admin Agent Explorer.

client object

See agent object.

component class

A component class is a type of class used to identify the items (files, registry entries, links, icons, and so forth) that make up the content identified by a CM Configuration Server class instance. Typically, this class' instances have distributable data associated with them such as FILE, REGISTRY, or DESKTOP.

Use the CM Admin CSDB Editor's Class Editor to set the class type to "Component".

configuration class

A configuration class identifies content to be managed on subscribers' computers by grouping together instances of component classes. Typically, a configuration class' instances do not have distributable data associated with them. They are connected to instances of one or more component classes, perhaps through an instance of another configuration class. Examples: ZSERVICE, PACKAGE, VGROUP, VERSION, and so forth.

Use the CM Admin CSDB Editor's Class Editor to set the class type to "Configuration."

CM Admin Agent Explorer

The CM Admin Agent Explorer (Object Editor) can be used to view or edit local objects, or create new objects. You can also use the CM Admin Agent Explorer to view objects located on a file server or on other computers to which you are connected via a local area network (LAN).

CM Admin Publisher

The CM Admin Publisher is used to create packages of data and store them in (i.e., promote them to) the CM Configuration Server DB.

CM agent

The CM agent runs on the subscriber's computer. It communicates with the CM Configuration Server to receive information about the desired state of the subscriber's computer, and compares that information to the actual state of the subscriber's computer. Then, the CM agent makes any adjustments necessary to make the actual state match the desired state.

CM Application Manager

The CM Application Manager Radskman is the CM agent executable that manages mandatory services. The CM administrator uses the CM Admin CSDB Editor to specify the services that the CM Application Manager manages on the subscriber's computer. No user interface is available.

CM Application Self-service Manager

The CM Application Self-service Manager is the CM agent used to manage optional services. The CM administrator uses the CM Admin CSDB Editor to specify the services that are available to the subscriber.

The subscriber installs and manages data that is available from the CM Application Self-service Manager user interface (Service List).

CM Configuration Server

Also called *Active Component Server* or *Manager*.

The CM Configuration Server distributes applications to client computers. It runs on the server and maintains the CM Configuration Server DB, which stores information that the CM Configuration Server needs to manage digital assets for distribution to client computers.

CM Configuration Server DB

The CM Configuration Server DB stores all of the information necessary to manage digital assets on a client computer, including:

- The software and/or data that CM distributes.
- The desired state of each client computer with respect to the CM - managed content.
- The policies determining which subscribers can subscribe to which packages.

- Security and access rules for CM administrators.

Use the CM Admin CSDB Editor to manipulate the CM Configuration Server DB.

CM Admin CSDB Editor

The CM Admin CSDB Editor is used to manipulate the contents of the CM Configuration Server DB.

CM Inventory Manager

The CM Inventory Manager is a policy-driven, inventory management tool that automatically discovers information about software and hardware, and consolidates the results into Web-based reports. The CM Inventory Management agent is a WBEM (Web-based Enterprise Management) consumer.

CM Scheduler

The CM Scheduler service (radsched), installed with the Application Manager, allows you to deploy a service at a specific time.

CM Staging Server

The Staging Server is used to store data required for deploying applications on a computer other than the computer with the CM Configuration Server.

Database

See *CM Configuration Server DB*.

desired state

The desired state embodies the content that CM manages for a specific subscriber's computer. A model representing the desired state for each subscriber's computer is stored in the CM Configuration Server DB. The desired state model is created and managed using the CM Admin CSDB Editor.

domain

A domain logically partitions a file in the CM Configuration Server DB to group "like" classes together.

Examples: POLICY Domain; SOFTWARE Domain; SYSTEM Domain

- The POLICY Domain contains the classes that identify users individually and by their association with groups of other users.

- The SOFTWARE Domain contains the classes needed to define and deploy applications. CM administrators will do most of their work in the POLICY and SOFTWARE Domains of the PRIMARY File.
- The SYSTEM domain contains the classes that contain administrative and process control definitions.

expression variable

An expression variable contains a single line REXX command that is executed during resolution. If the expression evaluates to **true** in an attribute named ZSTOP, it causes resolution of the current instance to end. Resolution continues in the calling instance with the variable following the one that called the instance containing the expression variable.

file

A file is the highest level in the hierarchy of the CM Configuration Server DB and it groups similar domains together.

Example: PRIMARY File

The PRIMARY File is used to define and maintain the distribution model. This is one of the pre-configured files distributed with the CM Configuration Server and installed when you first install CM. Others are the NOTIFY file and the PROFILE File. CM administrators will do most of their work in the PRIMARY File.

instance

Also called *class instance*.

An instance is a CM Configuration Server DB object containing a specific occurrence of a class. This is analogous to a row in a relational data table or a record in a traditional flat file. The attributes of an instance contain the data describing one specific entity of that class.

Manager

See CM Configuration Server.

mandatory service

A mandatory service is a service that is required on the subscriber's computer. Services are made mandatory by setting the ZSVCMO variable in the Application instance to M.

method

A method is a program that performs functions that are meaningful in the context from which they are called.

Methods can be written in REXX or in a language that produces an executable that can validly run on the platform where it is invoked. The HP-supplied REXX run-time environment interprets REXX methods.

Client methods run on the subscriber's computer, while CM Configuration Server methods run on the CM Configuration Server computer.

method variable

The method variable identifies the method, or program, to be executed as part of the resolution process.

For CM Configuration Server methods, it contains a reference to an instance of the SYSTEM domain PROCESS class that identifies the method to execute and the parameters to be passed to the method. CM Configuration Server methods are located in the CM Configuration Server BIN subdirectory for .exe methods or in the Manager REXX subdirectory for REXX methods.

For CM Agent methods, it contains the name of the method to execute on the subscriber's computer. The name of a method variable that executes a CM Agent method identifies the event (such as installing or removing software) for which the method should be executed. Client methods are located in the IDMSYS location on the subscriber's computer.

Notify

A notify forces one or more client computers to connect to the CM Configuration Server to update or remove an application or send an e-mail to subscribers of a particular service.

null instance

The null instance of a class is used when an instance of that class that does not exist. During resolution, if a connection is attempted to a non-existent instance of a class, the Null Instance is used. This provides a resolution path that handles broken connections.

object

An object is a data structure containing variables stored in a file with an `.EDM` suffix on the client computer. An object can consist of one or more instances. Each instance contains the same set of variables. The values held in the variables can vary from instance to instance.

Use the Client Explorer to view, edit, or create objects.

optional service

An optional service is a service that is available to subscribers via the Service List of the CM Application Self-Service Manager user interface. Services are made optional by setting the ZSVCMO variable in the Application instance to "O".

package

A package is the data that is published as an individual unit.

policy

A policy determines *which* subscribers (or computers) have access to *what* software. The POLICY domain class instances identify users. Connections to the POLICY class instances identify the content to be managed for those subscribers.

promote

When you promote a package that was created with the CM AdminPublisher, you are storing the package in the CM Configuration Server DB.

publish

To bundle a set of related data into a single unit that can be managed by CM.

Publisher

See CM Admin Publisher.

Software Manager

See CM Application Self-service Manager.

Staging Server

See CM Staging Server.

resolution

Resolution occurs when the CM Configuration Server accomplishes a unit of work in response to a service request. The unit of work is defined by the contents of the CM Configuration Server DB and parameters included in the service request itself.

In other words, what CM does depends upon what information is stored in the CM Configuration Server DB and what information accompanies the request for CM to perform some action.

For example, the CM Agent Connect submits service requests by sending an object to the CM Configuration Server. The CM Configuration Server then performs resolution in response to each request. The parameters that control the processing of the service request are in the input object.

resource

Also called *file*.

A resource is a single component that is bundled into a package. Examples of resources are files, desktop links, and sets of registry keys.

Scheduler

See CM Scheduler.

service

Also called a software application, application, or software.

A service is a group of related packages.

session

A session identifies a packaging exercise in CM Admin Publisher that results in the creation of one CM package.

Staging Server

See CM Staging Server.

subscriber

A subscriber is the person who uses CM -managed applications on a client computer.

symbol

A symbol is the name of a variable in global memory, preceded by an ampersand.

symbolic substitution

Database instances and client objects consist of variables that contain values. The value of a variable can contain a specification that refers to the value of another variable. During the resolution process, CM can 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, one of the _ALWAYS_ connection variables in the SYSTEM.PROCESS.ZMASTER instance of the Database contains the value POLICY.USER.&(ZMASTER.ZUSERID). The reference &(ZMASTER .ZUSERID) refers to the ZMASTER object's ZUSERID variable, which contains the user ID typed into the CM logon dialog box on the CM Agent, when the subscriber visits the CM Software Management Web page. If the user typed in JDOE for the user ID, symbolic substitution would render the effective value of the _ALWAYS_ connection variable as POLICY.USER .JDOE.

The substitution is not permanent, i.e., the value in the CM Configuration Server DB does not change. Only the value in the in-storage object derived from the CM Configuration Server DB instance for the current resolution process contains the substituted value.

The parentheses are required only if the reference is qualified, i.e., contains a period. If the reference is unqualified, the parentheses are optional.

For example, these symbolic substitution specifications are correct:

&(ZMASTER.ZUSERID)

&(ZUSERID)

&ZUSERID

and this is incorrect:

&ZMASTER.ZUSERID

System Explorer

See CM Admin CSDB Editor

Timer

See CM Scheduler.

variable

A variable is a piece of named storage that contains a changing value. The variable's value forms a part of the client's resolved distribution model and can influence the resolution process through messaging or symbolic substitution.

version group

A version group is a collection of one or more versions of one application that CM deploys and manages. Use version groups to roll out a new version of an application to the appropriate subscribers, and activate it upon delivery or at a pre-determined time.

Web-based Enterprise Management (WBEM)

Web-Based Enterprise Management (WBEM) is an initiative from the Distributed Management Task Force (DMTF) to develop standard technologies for accessing management information in an enterprise-computing environment.

Windows Management Instrumentation (WMI)

Windows Management Instrumentation (WMI) is the Microsoft implementation, for Windows platforms, of Web-Based Enterprise Management (WBEM). WMI provides support for WBEM's Common Information Model (CIM).

ZCONFIG

The ZCONFIG object contains basic hardware information for the client computer such as processor, operating system, and drives.

ZMASTER

The ZMASTER object contains information about the client computer that is necessary to run the Application Manager such as the identity of the subscriber and the IP address of the client computer.

ZTIMEQ

The ZTIMEQ object is created, based on information in the Scheduler (TIMER) instance, when a timer is deployed to the client computer.

.

Index

B

C

D

E

F

G

H

I

J

L

M

N

O

P

Q

R

S

T

U

V

W

Z