HP Configuration Management

OS Manager

for the Windows® operating system

Software Version: 5.10

Hardware Configuration Management Guide

hp ®

i n v e n t

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

OpenLDAP is a registered trademark of the OpenLDAP Foundation.

PREBOOT EXECUTION ENVIRONMENT (PXE) SERVER
Copyright © 1996-1999 Intel Corporation.

TFTP SERVER
Copyright © 1983, 1993
The Regents of the University of California.

OpenLDAP
Copyright 1999-2001 The OpenLDAP Foundation, Redwood City, California, USA.
Portions Copyright © 1992-1996 Regents of the University of Michigan.

OpenSSL License
Copyright © 1998-2001 The OpenSSLProject.

Original SSLeay License
Copyright © 1995-1998 Eric Young (eay@cryptsoft.com)

DHTML Calendar
Copyright Mihai Bazon, 2002, 2003

Documentation Updates

The title page of this document contains the following identifying information:

- Software Version number, which indicates the software version.
  - The number before the period identifies the major release number.
  - The first number after the period identifies the minor release number.
  - The second number after the period represents the minor-minor release number.
- 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:

http://h20230.www2.hp.com/selfsolve/manuals

You will also receive updated or new editions if you subscribe to the appropriate product support service. If you have additional questions, contact your HP Sales Representative.

Table 1 indicates changes made to this document since the last released edition.

Table 1   Document Changes

Chapter	Version	Changes
Chapter 1	5.10 September 2008	Updated the Prerequisites on page 12 to remove HP ProLiant Servers.
Chapter 2	5.10	In To configure the Hardware Configuration Element on page 20, added information about changing the Service OS as needed.
Appendix A	5.10	About the Hardware Configuration Element Class on page 43, updated the information about Service OS Needed to Run Method on page 47.
Appendix Appendix D	5.10	Added information about Maintenance on page 57.
Appendix E	5.10	Added Pre-configured Hardware Configuration Elements for use with HP ProLiant Hardware on page 59.

Support

You can visit the HP Software support web site at:

www.hp.com/go/hpsoftwaresupport

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

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

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

Most of the support areas require that you register as an HP Passport user and sign in. Many also require an active support contract. To find more information about support access levels, go to the following URL:

http://h20230.www2.hp.com/new_access_levels.jsp

To register for an HP Passport ID, go to the following URL:

http://h20229.www2.hp.com/passport-registration.html

Contents

1 What is Hardware Configuration Management?

Hardware Configuration Management allows you to create and apply operations to hardware on a target device. For example, although some target devices may be ready to have the operating system installed out of the box, there may be other situations when you need to identify and apply critical operations before proceeding with the operating system installation. Examples of the types of operations necessary are upgrading the BIOS firmware or configuring a disk array controller (DAC). These operations prepare the hardware configuration on the target device. After the hardware configuration is complete (i.e., the hardware configuration is in its desired state), the target device is ready for the operating system to be installed.

You may also use Hardware Configuration Management in situations where you need to apply hardware configuration changes to a target device *after* an operating system has been installed.

About the Process Flow

Below is an overview of the process flow of Hardware Configuration Management.

1   Create one or more Hardware Configuration Elements (HWCEs).

Each element contains information about the resources required for the operation, the sequencing of operations, and how the operation should be carried out.

a   Gather the resources for your HWCEs. You may need to go to the vendor's web site to get the correct binaries and create the appropriate configuration files to apply to the target device.

b   Use the CM Administrator Publisher to publish the HWCEs to the HP Configuration Management Configuration Server Database (CM-CSDB).

c   Use the HP Configuration Management Portal (CM Portal) to create and configure the HWCE instances.

2   Create a Hardware Configuration object.

A Hardware Configuration object contains the information about how a target device's hardware must be configured in order for it to be ready for operating system installation. It contains one or more HWCEs.

a   Use the CM Portal to create a Hardware Configuration object and specify the filters that identify to which devices to apply the hardware configuration.

b   Connect the appropriate HWCEs to the Hardware Configuration object.

3   Set policy to ensure deployment of the hardware configuration and operating system.

> When you use Hardware Configuration Management, Model and Manufacturer are the two most often used policy classes.

4   The next time the target device connects; it detects the hardware configuration and applies the tasks connected to this hardware configuration.

— If there is more than one task, they will be applied in the order specified by the administrator.

— If a task needs a reboot, the HP Configuration Management OS Manager (CM OS Manager) will reboot the target device and proceed to the next task.

— If the operating system needs to be re-installed after a change, then it will be re-installed.

Audience

Only experienced Configuration Management (CM) and system administrators who are very familiar with manipulating hardware and system components should use Hardware Configuration Management.

Prerequisites

• CM OS Manager 5.00 or higher

- HP Configuration Management Configuration Server (CM Configuration Server) 5.00 for Windows or higher

- HP Configuration Management Administrator (CM Admin) 5.00 for Windows or higher

- CM Portal 5.00 for Windows or higher

Chapter Summaries

Chapter 1, What is Hardware Configuration Management?

This chapter provides an introduction to Hardware Configuration Management and its process flow, as well as descriptions of the target audience and prerequisite knowledge.

Chapter 2, Using Hardware Configuration Management

This chapter includes an example of how to create and apply operations to hardware on a target device.

Chapter 3, Repairing Devices

This chapter discusses how to use the Repair Device task to fix a problem with a specific machine. It also covers the use of Shadow HWCEs and the special circumstances in which these are used.

Appendix A, About the Hardware Configuration Element Class

This appendix provides detailed information about the fields used to create HWCEs.

Appendix B, About the Hardware Configuration Class

This appendix provides detailed information about the fields used to create Hardware Configuration objects.

Appendix C, Best Practices

This appendix provides some best practices to follow when using Hardware Configuration Management.

Appendix D, Maintenance

This appendix specifies where to place fixes received from HP.

Appendix E, Pre-configured Hardware Configuration Elements for use with HP ProLiant Hardware

This appendix describes the pre-configured Linux-based Hardware Configuration Elements (HWCEs) that are available to allow you to perform various low-level configuration changes on HP ProLiant hardware.

Related Documents

HP Configuration Management OS Manager System Administrator Guide

2 Using Hardware Configuration Management

At the end of this chapter, you will:

- Be able to create and apply operations to hardware on a target device.

In this chapter, you will learn how to create and apply operations to hardware on a target device, such as a ProLiant server. In this example, you must make the following changes to the hardware configuration before you can install the operating system:

- Update an old version of BIOS firmware (BIOS).

- Configure a disk array controller (DAC).

- Assemble several raw disks into logical volumes (LVOLS).

Step 1: Create one or more HWCEs.

Step a: Gather the Resources for your HWCEs

Table 2 on page 17 defines each of the HWCEs to be used in this example. First, let us review the columns in the table.

- **Hardware Configuration Element** provides a general description for this HWCE. You should type a more descriptive instance name and friendly name for the element in the CM Portal.

- **Operation** describes the type of operation that will be applied to the target hardware.

- **Resource files** lists the files that must be gathered into a single folder. These are the files that will perform the operations.

> Be sure that text files are in the UNIX text file format.

- **Apply methods** are the commands that initiate the operations.

- **Post-requisite HWCE** is also known as a consequence. If a HWCE is applied to a target device, some HWCEs might have to be reapplied after the original is applied. For example, if you use a HWCE to change the BIOS firmware, the settings will most likely be reset to their default values. Therefore, you would need to use a HWCE to reset the default values back to the settings needed for your environment. This element would be considered the consequence (or Post-requisite HWCE) for the BIOS firmware HWCE.

- **Conditional Pre-requisite HWCE** is also known as a dependency. Use this field to set up the order in which the HWCEs are applied. This is most effective when you are first applying your hardware configuration and need to apply the HWCEs in a specific order.

Table 2   Hardware Configuration Elements example definitions

Hardware Configuration Element	Operation	Resource Files	Apply Method	Post-requisite HWCE	Conditional Pre-requisite HWCE
Update BIOS	Update firmware	`Biosfw.img` `Biosflash.exe`	`./biosflash.exe biosfw.img`	None	None
Disk Array Controller (DAC)	Configure DAC	`Array.conf` `Acrutil.exe` `arraycfg.sh`	`./arraycfg.sh` Note: This script contains the command `acrutil.exe array.conf`	LVOL	None
Logical Volumes (LVOL)	Create logical volumes	`Volume.conf` `Volutil.exe` `volume.sh`	`./volume.sh` Note: This script contains the command `volutil.exe volume.conf`	DRIVE MAP	None

The BIOS HWCE will update the firmware on the target device. To perform this operation, the specified resource files will be initiated using the command line listed in the Apply Method column. The BIOS does not have any dependencies or consequences, therefore changes to the BIOS may happen at any time in the sequence of events.

However, the DAC HWCE specifies LVOL as a consequence. This means that the DAC must be configured on the target device before the LVOL and that LVOL must be reapplied after any change to the DAC.

The LVOL HWCE has a consequence of DRIVEMAP. This consequence is a reserved type (see Appendix A, About the Hardware Configuration Element Class for more information about reserved types) that causes an operating system installation after the OS drive is repartitioned. Therefore, this operation will invalidate any preexisting operating systems.

When all three operations have been completed successfully, the target device is considered to have its hardware configuration completed and is ready for its operating system to be installed.

Step b: Publish the HWCEs

In this section, you will use the CM Publisher to publish the HWCEs to the CM Configuration Server DB (CM-CSDB).

> Before you publish your HWCEs, gather your resource files into a single folder.

To publish a HWCE

1   Go to **Start→All Programs→HP OVCM Administrator→CM Admin Publisher.** Refer to the *HP CM Administrator User Guide* for details on how to use the CM Publisher.

2   Type your User ID and Password.

3   From the Type of Data to Publish drop-down list, select **Hardware Configuration**.

4   Click **OK**.

    The Select window opens.

5   Select the folder that contains the resources needed to create your HWCE. In our example, we selected C:\HWCEs\BIOS.

> Make sure that you gathered the correct files that match the system to which you intend to deploy this. If you choose the wrong files you may leave your system in a damaged state.
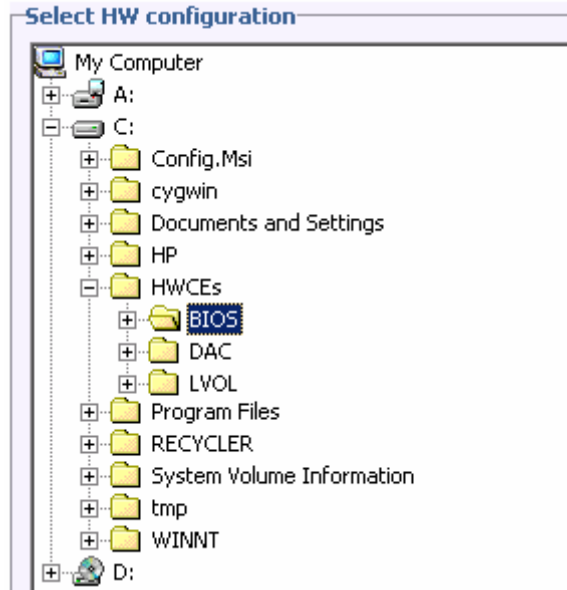
6 In the Description field, type a description of the elements that you are publishing. For this example, type **Pro32 WS Bios Rev 1.00 Resources**.

7 In the Package Instance Name field, type the instance name for the package. For this example, type **P32_BIOS_100**.

8 Click **Next**.

The Publish window opens.

9 Review the information and then click **Publish**. The package resources will be published in a non-compressed format.

10 When the CM Publisher is done, click **Finish**.

11 Click **Yes** to confirm that you want to close the CM Publisher.

Repeat the steps above to create packages for the Disk Array Controller (DAC) and the Logical Volumes (LVOL). Be sure to use descriptive instance names and provide detailed descriptions. For example:

For the Disk Array Controller, set the:

- Description to Pro32 WS Array Mirrored (1 Spare) Rev 1.00 Resources
- Instance Name to P32_DAC_MIRR_100

For the Logical Volume, set the:

- Description to Pro32 WS Array 1 Drive Rev 1.00 Resources
- Instance Name to P32_LVOL_1DRV_100

Step c: Create and configure the HWCEs

In this section, you learn how to configure HWCEs that identify the resources and methods to use to bring the hardware configuration for a target device to its desired state.

This section provides simple step-by-step instructions to help you to gain a basic understanding about how to create and configure hardware elements. See Appendix A, About the Hardware Configuration Element Class for detailed information about the fields and the information you enter into the fields.

To configure the Hardware Configuration Element

1   Open your web browser and go to the CM Portal (http://*ipaddressORhostname*:3466).

2   Log on as the CM OS Manager administrator (by default, the user ID is romadmin and the password is secret).

3   In the workspace, click the appropriate CM Configuration Server service.

4   In the workspace, click **HW Config Element**.

5   From the CM OS Manager Administration task group, click **Create Instance**.

    The Create Instance window opens.

6   Type the Instance Name and the Friendly Name and click **Create**.

When creating the HWCE instances, you may want to use the following Instance and Friendly Names.

For the BIOS, set the:

- Instance Name to P32_BIOS_100
- Friendly Name to Pro32 WS Bios Rev 1.00

For the Disk Array Controller, set the:

- Instance Name to P32_DAC_MIRR_100
- Friendly Name to Pro32 WS Array Mirrored (1 Spare) Rev 1.00

For the Logical Volume, set the:

- Instance Name to P32_LVOL_1DRV_100
- Friendly Name to Pro32 WS Array 1 Drive Rev 1.00

7 If the HWCE must be run in the WinPE 2.0 Service Operating System environment, you may need to change the default Service OS (_SVC_LINUX)_ to _SVC_PEX86_. To do this, click **Modify Instance**.

8 In the Service OS Needed to Run Method text box, type **_SVC_PEX86_.**

   Be sure that you set the Service OS for each HWCE to ensure it will run in the required Service Operating System.

9 From the CM OS Manager Administration task group, click **Connect Packages**.

   The Connect Packages window opens.

Connect Packages to Pro32 WS Bios Rev 1.00

① Select — ② **Add** — ③ Summary

Selection

Class: Os - OS Packages

Packages

Available

```
-----------------------------------
DAC
GARHOPT
Local Service Boot
P32_BIOS_100
P32_DAC_MIRR_100
P32_LVOL_1DRV_100
ROM Client Methods
WIN2K
```

Selected

```
-----------------------------------
```

Next | Reset | Cancel

10 From the Available list, select the OS Packages that you want to connect to the HWCE and click the arrow. For example, if you are creating the Pro32 WS Bios Rev 1.00 instance, you would select the P32_BIOS_100 package.

11 Click **Next**.

12 Click **Commit**.

13 From the CM OS Manager Administration task group, click **Modify Instance**.

The Modify window opens.

Modify Pro32 WS Bios Rev 1.00

Basic | *Advanced*

* *Default Values*

Properties

Friendly Name	Pro32 WS Bios Rev 1.00
Version *	1.00
Class *	_REGULAR_ ▼
Type *	BIOSFLASH
Instance Number (0 -> apply to all) *	0
Service OS Needed to Run Method *	_SVC_LINUX_
Apply Method *	
Apply Behavior *	_NEVER_ ▼
Post Behavior *	_CONTINUE_ ▼
Failure *	_CRITICAL_ ▼
Post-requisite HWCE *	
Conditional Pre-requisite HWCE *	

[Modify] [Cancel]

14 Enter the information needed to configure the BIOS element as described
Table 3 on page 24.

> See Appendix A, About the Hardware Configuration Element
> Class for a complete description of each of the fields below.
>
> As identified in this appendix, you *must* type entries for the
> following fields.
>
> - Friendly Name
> - Class
> - Type
> - Instance Number
> - Apply Method
> - Apply Behavior
> - Post Behavior
> - Failure

Table 3   Configuring the BIOS HWCE

Field	Value
Friendly name	Pro32 WS Bios Rev 1.00
Version	1.00
Class	_REGULAR_
Type	BIOSFLASH
Instance Number	0
Service OS Needed to Run Method	_SVC_LINUX_
Apply Method	`./biosflash.exe biosfw.img`
Apply Behavior	_NEVER_
Post Behavior	_CONTINUE_
Failure	_CRITICAL_
Post-requisite HWCE	N/A
Conditional Pre-requisite HWCE	N/A

15 Click **Modify**.

The Hardware Configuration Element Properties window opens.
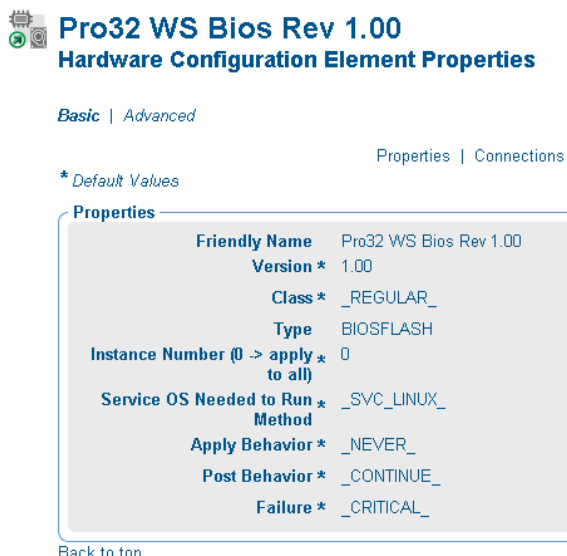
16 Repeat the steps above to configure the HWCEs for the Disk Array
   Controller (DAC) and the Logical Volumes (LVOL). Use Table 4 and
   Table 5 below as a guide for the information to enter.

Table 4   Configuring the DAC HWCE

Field	Value
Friendly name	Pro32 WS Array Mirrored (1 Spare) Rev 1.00
Version	1.00
Class	_REGULAR_
Type	ACRCONFIG
Instance Number	0
Service OS Needed to Run Method	_SVC_LINUX_
Apply Method	`/arraycfg.sh`
Apply Behavior	_NEVER_
Post Behavior	_CONTINUE_
Failure	_CRITICAL_
Post-requisite HWCE	LDRIVCFG
Conditional Pre-requisite HWCE	N/A

Table 5   Configuring the LVOL HWCE

Field	Value
Friendly name	Pro32 WS Array 1 Drive Rev 1.00
Version	1.00
Class	_REGULAR_
Type	LDRIVCFG
Instance Number	0
Service OS Needed to Run Method	_SVC_LINUX_
Apply Method	`./volume.sh`
Apply Behavior	_NEVER_

Field	Value
Post Behavior	_CONTINUE_
Failure	_CRITICAL_
Post-requisite HWCE	DRIVEMAP
Conditional Pre-requisite HWCE	N/A

Step 2: Creating the Hardware Configuration object

The hardware configuration is the configuration that a target device needs in order to be in its desired state. In this section, we will configure the Hardware Configuration object.

To create the hardware configuration

1   Open your web browser and go to the CM Portal (http://*ipaddressORhostname*:3466).

2   Log on as the CM OS Manager administrator (by default, the user ID is romadmin and the password is secret).

3   Go to the appropriate CM Configuration Server.

4   In the workspace, click **HW Config**.

The Hardware Configuration objects appear in the workspace.
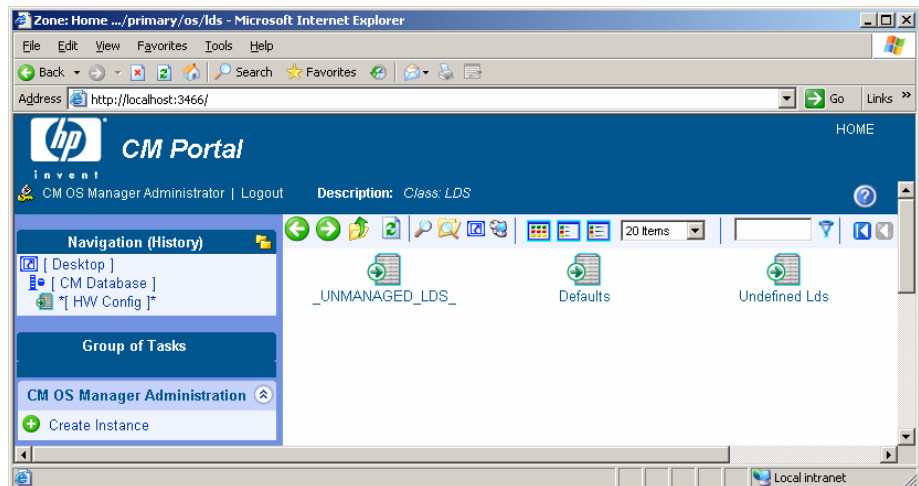
5   From the CM OS Manager Administration task group, click **Create Instance**.

6   Type the Instance name, such as `P32_CAD_100` and the Friendly name, such as `Pro32 WS CAD Setup Rev 1.00.`

7   Use the Reg Expr to Filter Manufacturer field *and* the Reg Expr to Filter Model field to restrict the hardware that the hardware configuration is applied to.

> You *must* enter information in the filter fields as a regular Tcl expression otherwise the hardware configuration will never be applied to your target device. Filter entries are case-sensitive.
>
> If you need assistance with regular Tcl expressions, refer to a manual such as *Practical Programming in Tcl and Tk* by Brent B. Welch, Prentice Hall PTR (3rd edition).

a   In the Reg Expr to Filter Manufacturer field, type a regular Tcl expression to indicate the manufacturer that you want to apply this hardware configuration to. If you do not enter information in this field, no match will be made.

–   If you type a value of `.*` the hardware configuration will be applied to any target device. *We do not recommend this because your hardware configuration may cause damage to some of your target devices.*

–   List the items as a series of regular expressions separated by a space.

Some example values for the manufacturing filter might be:

```
HP.* Compaq.* COMPAQ.*
```

or

```
Dell.* DELL.*
```

b   In the Reg Expr to Filter Model field, type an expression to indicate the model that you want to apply this hardware configuration to.

–   If you type a value of `.*` the hardware configuration will be applied to any target device. *We do not recommend this because the hardware configuration targeted at a particular manufacturer may cause damage to some of your target devices. It is important consider how to use your filters and be sure to filter based on both manufacturer and model.*

Some example values for the model filter might be:

```
DL360-G2.* Dl360.* PROLIANT.*DL360.*
```

or

```
Poweredge.* .*2450.*
```

These patterns match the models of DL360 servers or Poweredge 2450 server models.

> ⊘ Be careful not to type an expression such as DL360-C2* because in this case the asterisk is not used as a wildcard as you would expect. The asterisk will cause a match to any model that has repeated 2s in it. For example, some potential matches would be DL360-C22, DL360-C222, DL360-C2222222222 and so on.

> ⚠ The manufacturer and model filters must match for the hardware configuration to be applied to the target device.

8 Click **Create**.

If you want to specify a version number to track the versions of your configuration elements from the CM OS Manager Administration task group, click **Modify Instance** and type a version number in the appropriate field.

9 Click **Modify**.
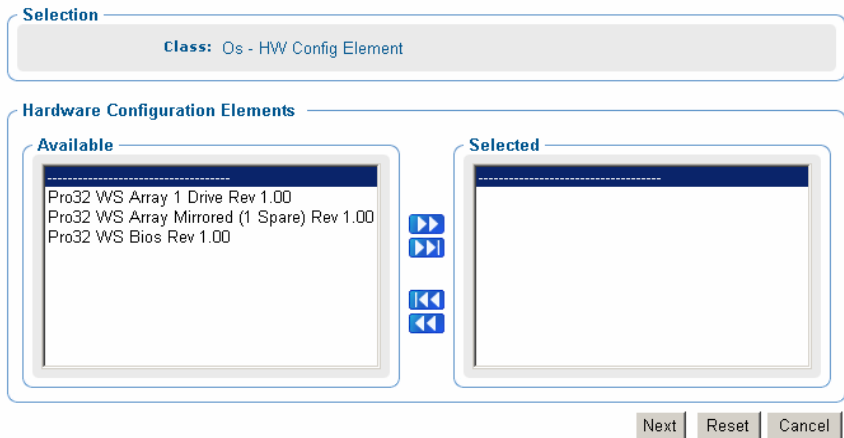
To connect the HWCEs to the Hardware Configuration Package

1 Open your web browser and go to the CM Portal (http://*ipaddress*OR*hostname*:3466).

2 Log on as the CM OS Manager Administrator (by default, the user ID is romadmin and the password is secret).

3 In the workspace, click **Hardware Configuration**.

4 In the workspace, click the instance such as Pro32 WS CAD Setup Rev 1.00.

5 In the CM OS Manager Administration task group, click **Connect Hardware Configuration Elements**.

The Connect Hardware Configuration Elements window opens.

6   From the Available list, select the elements that you want to connect to your hardware configuration and click . In this example, select the three elements listed.

7   Click **Next**.

8   Click **Commit**.

Step 3: Set Policy

Before you set policy that will deploy hardware configurations to your target devices, you must understand how policy works.

This section provides an overview of the steps needed to set policy. For details about how policy works in the CM OS Manager, see About Policy in the *HP Configuration Management OS Manager System Administrator Guide*.

To set policy

1   Open your web browser and go to the CM Portal (`http://`*ipaddress*OR*hostname*`:3466`).

2   Log on as the CM OS Manager administrator (by default, the user ID is romadmin and the password is secret).

3   In the workspace, click the appropriate CM Configuration Server.

4   Go to the appropriate policy class such as Machine Manufacturer or
    Machine Model.

5   Select the appropriate instance and from the CM OS Manager
    Administration task group, use the Connect Hardware Configurations
    task to select the hardware configuration object you want to deploy to the
    target device.

6   From the Available list, select the appropriate hardware configuration
    and click ⏭

7   Click **Next**.

8   Click **Commit**.

Reviewing the Current Hardware Configuration Elements

After a CM OS connect has occurred and the hardware configuration has
been deployed to a target device, you can review the list of the HWCEs that
have been successfully applied. To do this, go to the ROM object for the device
and review the Hardware Configuration area.

🖥 **Rom**
ROM object Properties

Basic | *Advanced*

Properties | Hardware Configuration | Resultant Policy | Events | Computer Information

Properties

OS State	_DESIRED_
Current OS	WIN2K
Chosen OS	WIN2K
Last Resolved OS(es)	WIN2K

Back to top

Hardware Configuration

Current Hardware Configuration	P32_CAD_100
Chosen Hardware Configuration	P32_CAD_100
Resolved Hardware Configurations	P32_CAD_100
Current Hardware Configuration Elements	P32_BIOS_100
	P32_DAC_MIRR_100
	P32_LVOL_1DRV_100

Back to top

In this area, you can see both the Hardware Configuration object that has
been applied, as well as all HWCEs that are currently remembered by the
target device.

Modifying the Hardware Configuration on a Device

If we want to continue with our example, consider that once your hardware configuration is in its desired state, it may remain that way for quite a while. However, at some time in the future, you may need to change the hardware configuration. For example, you may want to update your BIOS with the latest version of firmware. If you want to do this, you would:

1   Publish the resources for the updated firmware. For example, you would publish the resources with the description Pro32 WS Bios Rev 2.00 Resources and the instance name P32_BIOS_200.

2   Create the appropriate HWCE for the updated firmware, such as Pro32 WS Bios Rev 2.00 (instance name P32_BIOS_200).

3   Create a new Hardware Configuration object, such as Pro32 WS CAD Setup Rev 1.10 (instance name P32_CAD_110), and connect the following HWCEs to this object

    —   Pro32 WS Bios Rev 2.00

    —   Pro32 WS Array 1 Drive Rev 1.00

    —   Pro32 WS Array Mirrored (1 Spare) Rev 1.00

4   Change policy to deploy the new Hardware Configuration object Pro32 WS CAD Setup Rev 1.10.

5   The next time the target device does a CM OS connect, it detects the hardware configuration and applies the tasks connected to this hardware configuration.

6   Go to the ROM object for the target device to view the Current Hardware Configuration again. Notice that the new Hardware Configuration object, P32_CAD_110, is listed, as well as the Current HWCEs. In this example, the elements, P32_DAC_MIRR_100 and P32_LVOL_1DRV_100 remain unchanged and only the new BIOS, P32_BIOS_200 will be applied to reach the new desired state.

The ROM Object Properties window opens.

Rom
ROM object Properties

Basic | *Advanced*

Properties | Hardware Configuration | Resultant Policy | Events | Computer Information

Properties

OS State	_DESIRED_
Current OS	WIN2K
Chosen OS	WIN2K
Last Resolved OS(es)	WIN2K

Back to top

Hardware Configuration

Current Hardware Configuration	P32_CAD_110
Chosen Hardware Configuration	P32_CAD_110
Resolved Hardware Configurations	P32_CAD_110
Current Hardware Configuration Elements	P32_BIOS_200
	P32_DAC_MIRR_100
	P32_LVOL_1DRV_100

Back to top

If there were no consequences or dependencies, the elements DAC and LVOL were not modified at all. However, if there was some consequence, then one or both of the elements may have been reapplied to the device.

3 Repairing Devices

At the end of this chapter, you will:

- Be able to repair a device by reapplying a Hardware Configuration Element (HWCE) to your system.

- Understand what a Shadow HWCE is and when to use one.

Hardware Configuration Management operates based on the Last Remembered State (also known as the Current Hardware Configuration Elements). This means that the actual state may be different from what the CM OS Manager remembers as the last state for the hardware configuration.

If you determine that the actual state of your Hardware Configuration is different from the Current Hardware Configuration, you can use the Repair Device task to correct conflicts. Note that the system will not recognize that there is a problem. This task allows you to reapply a HWCE to your system.

Some examples of when you may need to repair a device are:

- If you flashed the BIOS, but someone flashed it back. You may want to use Repair Device to flash the BIOS again.

- If your HWCE configures the BIOS with a password and someone disables it, you can use this task to repair it.

The Repair task is typically targeted toward a specific device and therefore, rather than assigning policy, we access this task by going to the ROM object for the device that you want to repair.

To repair a device

1 Open the CM Portal and log on as a CM administrator (by default romadmin, password is secret).

2 Go to the ROM object for the target device.

3   From the CM OS Manager Administration task group, click **Repair Device**.

The Repair Device – Select element window opens.

In this window, you see what the CM OS Manager recognizes as the current state of the target device. For this topic, let us look at on the Hardware Configuration Element area, which lists the elements that were successfully deployed.

You may know that this does not match reality and want to repair the hardware configuration. For example, the user of this device may have flashed the BIOS.

4   Select the check boxes for the Hardware Configuration Elements that you want to repair, such as Pro32 WS Bios Rev 2.00.

5   Click **Next**.

The Repair Device – Repair option window opens.

6   Select one of the following options:

> ⚠ We recommend that you choose **Select this option if you want to perform other tasks such as data personality capture before repairing the device**.
>
> The second option is a last resort that you should use only if your device is severely damaged and a CM OS connect is not possible.

— **Select this option if you want to perform other tasks such as data personality capture before repairing the device.**
This option defers the repair until you notify the device. It operates similar to the Re-evaluate/install OS task.

— **Select this option if you want to repair the device immediately.**
This option repairs the device during the next connect. If the device is not booted, the repair occurs when the device boots. If the device is booted, the repair occurs during the next CM OS connect.

7 Click **Next**.

8 Review the summary and click **Submit**.

The ROM object properties window opens.

At this time, you will notice that the HWCEs that you selected to repair are no longer listed in the Hardware Configuration area next to the Current Hardware Configuration Elements field. This is because the state of this element is no longer known.

9 After the next CM OS connect, the HWCE is reapplied and this can be seen in the list of the Current Hardware Configuration Elements.

Rom
ROM object Properties

Basic | *Advanced*

Properties | Hardware Configuration | Resultant Policy | Events | Computer Information

Properties

OS State	_DESIRED_
Current OS	WIN2K
Chosen OS	WIN2K
Last Resolved OS(es)	WIN2K

Back to top

Hardware Configuration

Current Hardware Configuration	P32_CAD_110
Chosen Hardware Configuration	P32_CAD_110
Resolved Hardware Configurations	P32_CAD_110
Current Hardware Configuration Elements	P32_DAC_MIRR_100
	P32_LVOL_1DRV_100
	P32_BIOS_200

Back to top

Using Shadow Hardware Configuration Elements

If you cannot repair your device using the Repair Device task to reapply a HWCE, then you may need to use a Shadow HWCE. A Shadow HWCE is an exception to how a device's state is typically managed. A Shadow HWCE is applied to your target device once and is then removed from the hardware configuration. In other words, the operation is run once on a device, but does not become part of the Current HWCEs (also known as the Last Remembered State).

This is used in rare cases where external factors or physical changes to a target device might require that a maintenance task is performed. You are not reapplying the entire hardware configuration, but are simply tweaking the configuration.

> Note that there is *no way to check if a BIOS setting that you applied still exists*. You must determine whether to reapply the HWCE, apply a Shadow HWCE, or re-install the operating system.

For example, use a Shadow HWCE if you set up a disk array. If a drive fails and is replaced by a new one, the target device is still operable and the disk array HWCE is intact. However, the controller might require a trigger to synchronize the new drive as a spare. To do this, a Shadow HWCE must be defined and added to the Last Remembered State, so that the Last Remembered State reflects reality.

Shadow HWCEs:

- Must be appended to the Current Hardware Configuration in a machine object.

- Are always applied before a regular HWCE.

- Cannot have any consequences (Post-requisite HWCEs).

- Can only have dependencies (Conditional Pre-requisite HWCEs) on other Shadow HWCEs.

- Are only applied once, and then their association to the target device is removed.

- Does not alter the other Current HWCEs.

Shadow HWCEs are different from regular HWCEs because they are not assigned by policy. They are assigned to a specific device to perform a one-time operation. You are temporarily adding an element to the hardware configuration in order to put the target device back to its desired state.

To create and apply Shadow Hardware Configuration Elements

1 Use the CM Admin Publisher to publish the resources for the Shadow HWCE to the CM Configuration Server Database (CM-CSDB). For this example, you can name the package P32_DAC_MIRR_NS1_100 and type the following description: **Pro32 WS Array Mirrored Rev 1.00: Activate New Spare Resources**.

2 Open the CM Portal and log on as a CM OS Manager administrator (by default romadmin, password is secret).

3 Go to the appropriate CM Configuration Server.

4 In the workspace, select **HW Config Element**.

5 From the CM OS Manager Administration task list, select **Create Instance**.

6 Enter the Instance name, such as P32_DAC_MIRR_NS1_100, and friendly name, such as Pro32 WS Array Mirrored Rev 1.00: Activate New Spare, and click **Create**.

7 Click **Modify Instance**.

8 From the Class drop-down list, select _**SHADOW**_.

9 In the Type field, type **ACRCONFIG**.

10 In the Apply method field, type the apply method to be used.

11 Click **Modify**.

12 Go to the ROM object for the target device.

13 From the CM OS Manager Administration task group, click **Repair Device**.

The Select Element window opens.
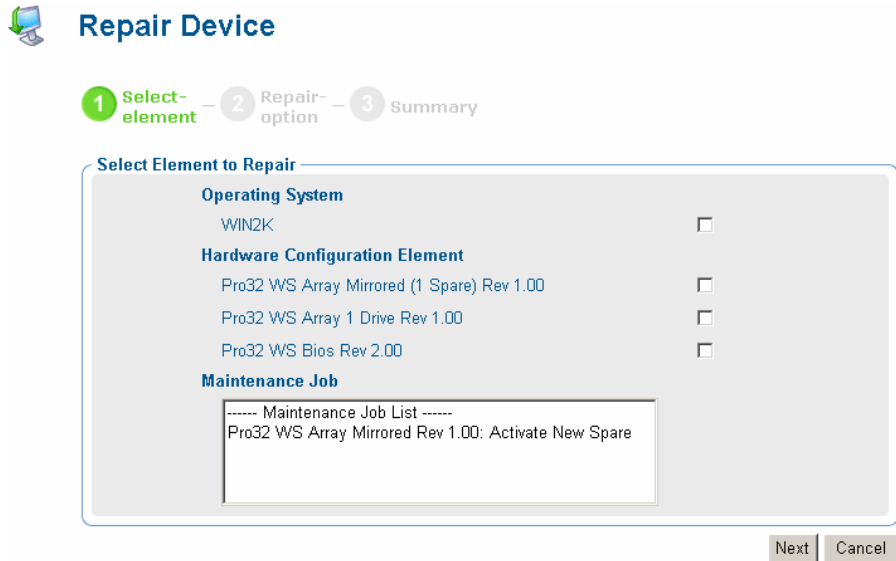
10 From the Maintenance Job list, select the Shadow HWCEs that you want to apply to this device, for example, Pro32 WS Array Mirrored Rev 1.00: Activate New Spare.

11 Click **Next**.

The Repair option window opens.

12 Select one of the following options.

> ⚠ We recommend that you choose **Select this option if you want to perform other tasks such as data personality capture before repairing the device.**
>
> The second option is a last resort that you can use only if your device is severely damaged and a CM OS connect is not possible.

— **Select this option if you want to perform other tasks like data personality capture before repairing the device.**
This option defers the repair until you notify the device.

— **Select this option if you want to repair the device immediately.**
This option repairs the device (triggers the Shadow Hardware Configuration Element) immediately at the next reboot.

13 Click **Next**.

Repair Device

1 Select-element — 2 Repair-option — **3 Summary**

Selected Element to Repair

Hardware Configuration Maintenance : Pro32 WS Array Mirrored Rev 1.00: Activate New Spare

Selected Repair Option

Run other task like data personality capture before repairing the device.

Submit | Back | Cancel

14 Review the summary and click **Submit**.

No changes will be made to the target device until a CM OS connect occurs or you force the device to reboot. At this time, the system recognizes that the device needs to have the Shadow HWCEs applied. The Shadow HWCE is listed as part of the Current HWCEs for a short time only.

Rom
ROM object Properties

Basic | *Advanced*

Properties | Hardware Configuration | Resultant Policy | Events | Computer Information

Properties

OS State	_DESIRED_
Current OS	WIN2K
Chosen OS	WIN2K
Last Resolved OS(es)	WIN2K

Back to top

Hardware Configuration

Current Hardware Configuration	_NONE_
Chosen Hardware Configuration	P32_CAD_110
Resolved Hardware Configurations	P32_CAD_110
Current Hardware Configuration Elements	P32_DAC_MIRR_100
	P32_LVOL_1DRV_100
	P32_BIOS_200
	P32_DAC_MIRR_NS1_100

Back to top

After a CM OS connect occurs, the Shadow HWCEs are applied and deleted from the Current HWCEs. Notice that the P32_DAC_MIRR_NS1_100 is no longer listed in the Current HWCEs.

Rom
ROM object Properties

Basic | *Advanced*

Properties | Hardware Configuration | Resultant Policy | Events | Computer Information

Properties

OS State	_DESIRED_
Current OS	WIN2K
Chosen OS	WIN2K
Last Resolved OS(es)	WIN2K

Back to top

Hardware Configuration

Current Hardware Configuration	P32_CAD_110
Chosen Hardware Configuration	P32_CAD_110
Resolved Hardware Configurations	P32_CAD_110
Current Hardware Configuration Elements	P32_DAC_MIRR_100
	P32_LVOL_1DRV_100
	P32_BIOS_200

Back to top

A About the Hardware Configuration Element Class

This appendix provides detailed information about the fields used to create Hardware Configuration Elements (HWCEs).

Modify Pro32 WS Bios Rev 1.00

Basic | Advanced

* *Default Values*

Properties

Friendly Name	Pro32 WS Bios Rev 1.00
Version *	1.00
Class *	_REGULAR_
Type *	BIOSFLASH
Instance Number (0 -> apply to all) *	0
Service OS Needed to Run Method *	_SVC_LINUX_
Apply Method *	
Apply Behavior *	_NEVER_
Post Behavior *	_CONTINUE_
Failure *	_CRITICAL_
Post-requisite HWCE *	
Conditional Pre-requisite HWCE *	

Modify   Cancel

Table 6   Hardware Configuration Element Class

Field	Description	Default Value	Potential Values
Friendly Name (required)	Friendly name	N/A	N/A
Version	A user-defined version number. For reference only.	1.00	N/A

Field	Description	Default Value	Potential Values
Class (required)	Identifies whether the HWCE is a Regular HWCE or a Shadow HWCE.	_REGULAR_	_REGULAR_ A regular HWCE is an object that identifies the resources and methods you use to bring a target device the state where it is ready to have an operating system installed. _SHADOW_ A Shadow HWCE is an exception to how a device's state is typically managed. A Shadow HWCE is applied to your target device one time and is then deleted. See Using Shadow Hardware Configuration Elements on page 37 for more information.
Type (required)	Generally a user-defined identifier. A hardware configuration would typically have only one HWCE with this type. If there is more than one HWCE with the same type, then only the first encountered will be applied. If a system has identical hardware components that a HWCE can be applied to, then you can set more than one HWCE to the same type and set the INSTANCE attribute to 0.	N/A	**Examples of user-defined types** **BIOSFLASH** Flashes the BIOS of a device. **NVRCONFIG** Configures the non-volatile RAM of a system. The purpose of this is to batch configure a system's parameters such as bootpath, PXE enable, OS type, and so on. There are many different parameters that can be set on a ProLiant server to customize its configuration. **ACRFLASH** Flashes the firmware of an array controller.

Field	Description	Default Value	Potential Values
	See Instance Number on page 46 and About the Instance Number and Apply Method Attributes on page 50.		**ACRCONFIG** Configures the array controller with respect to RAID volumes that are built from raw local disks or raw fiber channel LUNs (==FC disk drives). **LDRVCONFIG** Creates the logical drives (e.g., C:) from either raw local disks or by carving off a part of a RAID volume that has been set up. **NICFLASH** Flashes the firmware of network interface controllers (NIC). **NICCONFIG** Configures any programmable elements of NICs. **HBAFLASH** Flashes the firmware of a SCSI or fiber channel host bus adapter (HBA) (disk controller). **HBACONFIG** Configures SCSI or fiber channel host bus adapter with regard to getting raw disks or FC LUNs (disks) assigned. **READ_CONFIG** Used for pre-configured HWCEs that read or capture the array/hardware/iLO/system configuration of the target device into a file that is

Field	Description	Default Value	Potential Values
			uploaded to the CM OS Manager.
			Reserved types
			Do not use the OS or DRIVEMAP types for your HWCEs because they are reserved for internal use only.
			OS
			Causes an OS installation.
			DRIVEMAP
			Causes an OS installation.
Instance Number (0 -> apply to all) (required)	A unique number used to establish a convention between the apply methods and the application of the apply methods to a set of HWCEs. Information in this field does not *do* anything until the person writing the apply method scripts uses it to establish conventions. See About the Instance Number and Apply Method Attributes on page 50 for more information.	0	0 or greater depending on the apply method scripts that you write. Typically, the value of 0 is used to indicate that you want a HWCE to be applied to *all identical components* of a device. If you do this, the apply method script must interpret 0 as an indication to apply the script to all instances of the hardware configuration elements. If you set this to 1, the apply method may be applied to only one hardware configuration element. Again, the apply method script must support this. You can use values greater than 0 to indicate that the hardware configuration element should be applied to one or more identical pieces of hardware.

Field	Description	Default Value	Potential Values
			See About the Instance Number and Apply Method Attributes on page 50 for more information.
Service OS Needed to Run Method	System defined service OS identifier.	_SVC_LINUX_	_SVC_LINUX_ to run the Linux Service Operating System. _SVC_PEX86 to run the WinPE 2.0 Service Operating System.
Apply Method (required)	A command line that initiates the HWCE. Note that text files must be in the UNIX text file format. Also, please note that if you have an apply method that reboots the machine and you want messages to be seen by the user, you must redirect any output to /dev/console. For example, if your apply method flashes BIOS and you want the user to see a message indicating that the BIOS is being flashed and not to touch the power button, redirecting the output would ensure that the user sees the message before the reboot. See About the Instance Number and Apply Method	N/A	N/A

Field	Description	Default Value	Potential Values
	Attributes on page 50 for more information.		
Apply Behavior (required)	Defines the reboot behavior upon completion of the HWCE's apply method. Note that apply behavior must be in sync with the apply method otherwise you may run into problems deploying your hardware configuration. An example of an inconsistency would be if your apply method includes a reboot command, but the Apply Behavior is set to _NEVER_.	_NEVER_	_ALWAYS_ The apply method always triggers a reboot after the operation without returning control to the OS Manager agent. _OK_ After success, the apply method triggers a reboot without returning control to the OS Manager agent only. _NEVER_ The apply method never reboots and always returns control to the OS Manager agent. _ERROR_ *For future use.*
Post Behavior (required)	After a HWCE has been processed successfully, this instructs the machine being configured what to do next with regards to its boot state.	_CONTINUE_	_REBOOT_ Upon successful completion of an apply behavior operation, this triggers a reboot of system. _CONTINUE_ Upon successful completion of an apply behavior operation, the system will not reboot. It will continue on to the next HWCE or OS installation operation after HWCE processing returns.

Field	Description	Default Value	Potential Values
Failure (required)	The command or shell script returned an error and this indicates how to deal with that error.	_CRITICAL_	_CRITICAL_ Any failure of this HWCE operation is considered critical and will terminate HWCE /OS install processing. _IGNORE_ Any failure of this HWCE operation is considered a success for the purposes of HWCE processing.
Post-requisite HWCE	The HWCEs listed here must be applied after the HWCE being configured. These may also be referred to as consequences.	N/A	Type a friendly name, Type, or a combination of Type_Instance Number. A space is used as the separator between multiple items.
Conditional Pre-requisite HWCE	If there are HWCEs that need to be applied, they must be applied before the HWCE currently being configured. These may also be referred to as dependencies.	N/A	Type a HWCE's friendly name, Type, or a combination of Type_Instance Number. A space is used as the separator between multiple items. For example, the following may be a list of dependencies: `BIOSFLASH  SMART5ICFG NVRAMFLASH_1` BIOSFLASH refers to the TYPE. SMART5ICFG refers to a specific HWCE. NVRAMFLASH_1 refers to the INSTANCE.

Field	Description	Default Value	Potential Values
Connections	Connections to published packages of resources.	N/A	N/A

About the Instance Number and Apply Method Attributes

You must understand the relationship between the instance number and the apply method.

The instance number is a unique, user-defined number used to establish a convention between an apply method (for a HWCE) and identical hardware in the target system. Information in this field does not *do* anything unless the person writing the apply method script uses it to establish conventions. For example, if you determine that setting the instance number to 0 means that the HWCE should be applied to all identical hardware in the target system, the person writing the apply method must establish this convention in the script, otherwise nothing will happen. Also, the person creating the HWCEs must define what is considered "identical hardware" based on the service OS, hardware configuration, and apply method.

Typically, instance values greater than zero designate that the hardware configuration element should be applied to one or more identical instances of a system's hardware components. The specific instances are defined by the script writer based on the service OS, hardware configuration, and apply method. For example, the apply method for a hardware configuration element with an instance number set to 1 may be intended to be applied to the first piece of matching hardware detected during an operating system boot. Or, it could mean that the apply method for the hardware configuration element with an instance number set to 2 may be intended to be applied to the second piece of matching hardware detected during an operating system boot. An instance number of 2 could also mean that this operation should be applied to all even numbered disks detected during operating system boot.

Remember, the value of the instance number is meaningless unless there is a convention between the CM use of the value and the apply method scripts that determine what is to be done with it.

Below are two examples of potential usage of these attributes.

Example 1

A system contains three identical disks that have been assigned the same HWCE. The instance number attribute in the HWCE is set to 1.

In this example, the apply method was written so that:

- An instance number set to 1 means that the operations should only be applied to one disk in the system. Whether the apply method is applied to disk 1, 2, or 3 depends on how the apply method was written.

- An instance number set to 0 means that the apply method is to apply the appropriate operations to each disk assigned to this hardware configuration element.

  To do this, the apply method script contains a loop that applies the method to all three disks. This loop ensures that the operation is applied to any number of disks that are assigned to this hardware configuration element.

If this same system contained 10 disks—five of type A and five of type B—you might want to apply the hardware configuration element only to type A disks. If this was the case, the script could be written so that if the instance number is set to 0, the HWCE would only be applied to type A disks.

Example 2

Two network interface cards (NICs) are eligible to PXE network boot but only one may be activated as PXE bootable at a time.

We could create two different Hardware Configuration Elements (HWCE) with the type set to NICCONFIG. The first HWCE enables the PXE boot and the second HWCE disables the PXE boot. All other parameters for the HWCEs are set identically.

In this case, you would not want to set the instance number to 0 for the HWCE that enables the PXE boot. If your script indicates that setting the instance number to 0 means that the operation is applied to all HWCEs, then all NICs would be enabled and this would cause issues.

However, you could set the instance number to 0 for the HWCE that disables the PXE boot. If your script indicates that setting the instance number to 0 means that the operation is applied to all HWCEs, then all NICs would be disabled and this may be useful.

Also, because you do not want the enable PXE and disable PXE HWCEs to overlap, you could create the apply method script to use an instance number set to 1 to designate the first NIC and you could use an instance number set to 2 to designate the second NIC.

B About the Hardware Configuration Class

This appendix provides detailed information about the fields used to create Hardware Configuration objects.

Table 7   The Hardware Configuration Class

RMP field	Description	Default value	Potential values
Friendly name	The name of the Hardware Configuration.	N/A	N/A
Version	A user-defined version number. For reference only.	1.00	N/A
Connections	Allows you to manually create connections to hardware configuration elements.	N/A	N/A
Reg Expr to Filter Manufacturer	Restricts the hardware that the hardware configuration is applied to based on manufacturer. You *must* enter information in the filter fields as a regular Tcl expression otherwise the hardware configuration will never be applied to your target device. *We strongly recommend that you create a thorough plan for the*	""	Any regular Tcl expression

RMP field	Description	Default value	Potential values
	proper filters. See Step 2: Creating the Hardware Configuration on page 26 for detailed information.		
Reg Expr to Filter Model	Restricts the hardware that the hardware configuration is applied to based on model. You *must* enter information in the filter fields as a regular Tcl expression otherwise the hardware configuration will never be applied to your target device. *We strongly recommend that you create a thorough plan for the proper filters.* See Step 2: Creating the Hardware Configuration on page 26 for detailed information.	""	Any regular Tcl expression
Service Create Ordering	For internal use only.	N/A	N/A
Stop Expressions	For internal use only.	N/A	N/A

C Best Practices

The following are some recommended best practices to follow when using Hardware Configuration Management.

- Deploy in a test environment before deploying Hardware Configuration Management in a production environment.

- Verify that the hardware configuration elements are correct with respect to scripts that you are running in the apply method.

D   Maintenance

If you receive a fix from CPE, place it in
`\IntegrationServer\OSM\SOS\linux\testing\work` and then reboot the
machine.

E Pre-configured Hardware Configuration Elements for use with HP ProLiant Hardware

A number of Linux-based Hardware Configuration Elements (HWCEs) are available to allow you to perform various low-level configuration changes on HP ProLiant hardware. These are supplied as a set of files that can be imported into your CM Configuration Server DB (CM-CSDB).

These HWCEs have been tested and verified on various HP ProLiant ML/DL hardware, however variations in the physical configuration of the hardware may require that you make adjustments.

These Hardware Configuration Elements must be connected to a Hardware Configuration Object and then to a behavior. If multiple Hardware Configuration Elements apply to a Hardware Configuration Object, you must specify the order in which the HWCEs should be executed using the pre-requisite/post-requisite parameters specified in the HWCEs.

Importing the HWCEs to the CM Configuration Server DB

To import the resources for the HWCEs

1 From the CM OS Manager media, copy the
   `hardware_configuration_management\HP\linux` directory which
   contains the resources to a temporary directory on the machine that has
   the CM Configuration Server installed.

2 Stop the CM Configuration Server.

3 Run `ImportLME.cmd`.

   This script populates the CM-CSDB with the new HWCEs. All of the
   required resources and `.sh` files are provided.

4 When the import is complete, start the CM Configuration Server.

5   Use the CM Portal and go to the CM-CSDB, HW Config Element to confirm that the HWCEs were imported.

About the Pre-configured HW Configuration Management Elements

In this section you will learn about each of the pre-configured HWCEs that have been provided.

1 - Read ProLiant Hardware Inventory [Server Specific]

This HW Configuration Element captures the hardware configuration of the system using the hwdisc3 utility. The captured file is *machineName*.dat by default and is sent to the /upload directory of the CM OS Manager Server.

> You can change the name of the file that gets uploaded when this HWCE is successfully executed. The name of the output file is machineName.dat by default. To change the name of the file from machineName.dat, modify the Apply Method as follows:
>
> hpdiscovery.sh *RequiredFileName*

Packages connected to this HWCE

- SSSTLIB, which contains ssstlibs.tar.gz

- HPDISCOVERY which contains the script hpdiscovery.sh, the utility hpdiscovery, the libraries hpdiscoverylibs.tar.gz and the plugins hpdisc_plugin.tar.gz

  (HPDISCOVERY version 2.00 from HP)

Properties for the Read ProLiant Hardware Inventory [Server Specific] HWCE

You can use the CM Portal to review the hardware configuration element as specified in Table 8 on page 61.

Table 8   Properties

Field	Value
Friendly name	Read ProLiant Hardware Inventory [Server Specific]
Version	1.00
Class	_SHADOW_
Type	READ_CONFIG
Instance Number	2
Service OS Needed to Run Method	_SVC_LINUX_
Apply Method	hpdiscovery.sh
Apply Behavior	_NEVER_
Post Behavior	_CONTINUE_
Failure	_CRITICAL_
Post-requisite HWCE	N/A
Conditional Pre-requisite HWCE	N/A

2 - Read ProLiant System Configuration

This HWCE captures the system configuration using the conrep utility. The captured file is system.xml by default and is sent to the /upload directory of the CM OS Manager Server.

> You can change the name of the file that gets uploaded when this HWCE is successfully executed. To change the name of the file from the default system.xml, modify the Apply Method as follows:
>
> **conrep.sh –s** *FileName*

Packages connected to this HWCE

- SSSTLIB contains `sstlibs.tar.gz`

- HW_INVENT_UTILS contains `conrep.xml`, `conrep.sh` and the utility conrep, version 1.81 from HP.

Properties for the Read ProLiant System Configuration HWCE

You can use the CM Portal to review the hardware configuration element as specified in Table 8 on page 61.

Table 9   Properties

Field	Value
Friendly name	Read ProLiant System Configuration
Version	1.00
Class	_SHADOW_
Type	READ_ CONFIG
Instance Number	4
Service OS Needed to Run Method	_SVC_LINUX_
Apply Method	`conrep.sh -s system.xml`
Apply Behavior	_NEVER_
Post Behavior	_CONTINUE_
Failure	_CRITICAL_
Post-requisite HWCE	N/A
Conditional Pre-requisite HWCE	N/A

3 - Read ProLiant Array Configuration

This HWCE captures the array configuration of the system using the acu utility. The captured file is `array.ini` by default and is sent to the `/upload` directory of the CM OS Manager Server.

You can change the name of the file that is uploaded when this HWCE is successfully executed. To change the name of the file from the default `array.ini`, modify the Apply Method as follows:

acu.sh –c *FileName*

Packages connected to this HWCE

- SSTLIB contains `sstlibs.tar.gz`

- ARR_CONFIG_COMMON contains `acu.tar.gz` (having `.acuxebin` utility, version 7.85.18.0) and `acu.sh`

Properties for the Read ProLiant Array Configuration HWCE

You can use the CM Portal to review the Hardware Configuration Element as specified in Table 8 on page 61.

Table 10   Properties

Field	Value
Friendly name	Read ProLiant Array Configuration
Version	1.00
Class	_SHADOW_
Type	READ_ CONFIG
Instance Number	1
Service OS Needed to Run Method	_SVC_LINUX_
Apply Method	acu.sh –c array.ini
Apply Behavior	_NEVER_
Post Behavior	_CONTINUE_
Failure	_CRITICAL_
Post-requisite HWCE	N/A
Conditional Pre-requisite HWCE	N/A

4 - Read ProLiant iLO Configuration

This HWCE captures the iLO configuration of the system using the hponcfg utility. The captured file is iLO.xml by default and is sent to the /upload directory of the CM OS Manager Server.

> You can change the name of the file that is uploaded when this HWCE is successfully executed. To change the name of the file from the default iLO.xml, modify the Apply Method as follows:
>
> **hpconfg.sh –w** *RequiredFileName*

Packages connected to this HWCE

- iLO_CONFIG_COMMON contains hponcfglibs.tar.gz, utility hponcfg (version 1.60 from HP), and hponcfg.sh

Properties for the Read ProLiant iLO Configuration HWCE

You can use the CM Portal to review the hardware configuration element as specified in Table 8 on page 61.

Table 11   Properties

Field	Value
Friendly name	Read ProLiant iLO Configuration
Version	1.00
Class	_SHADOW_
Type	READ_ CONFIG
Instance Number	3
Service OS Needed to Run Method	_SVC_LINUX_
Apply Method	hponcfg.sh –w iLO.xml
Apply Behavior	_NEVER_
Post Behavior	_CONTINUE_
Failure	_CRITICAL_
Post-requisite HWCE	N/A
Conditional Pre-requisite HWCE	N/A

5 – Write ProLiant System Configuration

This HWCE loads the system configuration as specified in an xml file used as input to the conrep utility.

Prerequisite

You must publish an .xml file containing the required system configuration (e.g., dl360g4.xml) as a package (e.g., WRITE_SYSCONFIG_FILE) and connect it to this HWCE.

> The file name in the Apply Method of the HWCE must match the name of the xml file that you publish. Modify the Apply Method as follows:
>
> conrep.sh –l *RequiredFileName.xml*

Packages connected to this HWCE

- SSSTLIB contains ssstlibs.tar.gz
- SYS_CONFIG_COMMON contains conrep (version 1.81), conrep.xml and conrep.sh
- WRITE_SYSCONFIG_FILE contains the xml input file (e.g., dl360g4.xml)

Properties for the Write ProLiant System Configuration HWCE

You can use the CM Portal to review the hardware configuration element as specified in

Table 12   Properties

Field	Value
Friendly name	Write ProLiant System Configuration
Version	1.00
Class	_REGULAR_
Type	NVRCONFIG
Instance Number	1

Field	Value
Service OS Needed to Run Method	_SVC_LINUX_
Apply Method	conrep.sh –l *.xml
Apply Behavior	_NEVER_
Post Behavior	_REBOOT_
Failure	_CRITICAL_
Post-requisite HWCE	N/A
Conditional Pre-requisite HWCE	N/A

6 – Write ProLiant Array Configuration

This HWCE deploys the Array (RAID) configuration of the system using the acu utility.

Prerequisite

You must publish an .ini file containing the required array configuration (e.g., array.ini) as a package (e.g., WRITE_ARR_CONFIG) and connect it to this HWCE.

> The file name in the Apply Method of the HWCE must match the name of the .ini file that you publish. Modify the Apply Method as follows:
>
> acu.sh –i *RequiredFileName.ini* –internal -reset

Packages connected to this HWCE

- SSSTLIB contains ssstlibs.tar.gz
- ARR_CONFIG_COMMON contains acu.tar.gz (utility .acuxebin ver 7.85.18.0) and the script acu.sh

Properties for the Write ProLiant Array Configuration HWCE

You can use the CM Portal to review the hardware configuration element as specified in Table 8 on page 61.

Table 13   Properties

Field	Value
Friendly name	Write ProLiant Array Configuration
Version	1.00
Class	_REGULAR_
Type	ACRCONFIG
Instance Number	4
Service OS Needed to Run Method	_SVC_LINUX_
Apply Method	acu.sh –i *.ini –internal -reset
Apply Behavior	_NEVER_
Post Behavior	_REBOOT_
Failure	_CRITICAL_
Post-requisite HWCE	N/A
Conditional Pre-requisite HWCE	N/A

7 – Write ProLiant iLO Configuration

This HWCE deploys the iLO configuration to the target system using the hponcfg utility.

Prerequisite

You must publish an `.xml` file containing the required iLO configuration as a package and connect it to this HWCE.

> The file name in the Apply Method of the HWCE must match the name of the `.xml` file that you publish. Modify the Apply Method as follows:

```
hponcfg.sh -f RequiredFileName.xml
```

Packages connected to this HWCE

- iLO_CONFIG_COMMON contains hponcfg (ver 1.60), `hponcfglibs.tar.gz` and `hponcfg.sh`.

Properties for the Write ProLiant iLO Configuration HWCE

You can use the CM Portal to review the hardware configuration element as specified in Table 8 on page 61.

Table 14   Properties

Field	Value
Friendly name	Write ProLiant iLO Configuration
Version	1.00
Class	_REGULAR_
Type	ILOCONFIG
Instance Number	1
Service OS Needed to Run Method	_SVC_LINUX_
Apply Method	`hponcfg.sh -f *.xml`
Apply Behavior	_NEVER_
Post Behavior	_CONTINUE_
Failure	_CRITICAL_
Post-requisite HWCE	N/A
Conditional Pre-requisite HWCE	N/A

8 - Deploy Array Configuration RAID [Default]

This HWCE deploys the default array (RAID Default) configuration for the target device using the acu utility. Under this mode the acu utility determines the type of RAID to be configured based on the number of hard drives available (e.g., if two hard drives of same capacity are found, it automatically creates RAID 1).

Packages connected to this HWCE

- SSSTLIB contains `ssstlibs.tar.gz`
- ARR_CONFIG_COMMON contains `acu.tar.gz` (`.acuxebin` ver 7.85.18.0) and `acu.sh`
- DEP_RAID_DEFAULT_FILE contains `pl-acu-rd.ini` (this contains the default RAID configuration)

Properties for the Deploy Array Configuration RAID Default HWCE

You can use the CM Portal to review the hardware configuration element as specified in Table 8 on page 61.

Table 15   Properties

Field	Value
Friendly name	Deploy ProLiant Array Configuration [Default]
Version	1.00
Class	_REGULAR_
Type	ACRCONFIG
Instance Number	0
Service OS Needed to Run Method	_SVC_LINUX_
Apply Method	`acu.sh –i pl-acu-rd.ini –internal -reset`
Apply Behavior	_NEVER_
Post Behavior	_REBOOT_
Failure	_CRITICAL_
Post-requisite HWCE	DRIVEMAP
Conditional Pre-requisite HWCE	N/A

9 - Deploy Array Configuration RAID 0

This HWCE deploys the RAID 0 configuration for the target device using the acu utility.

Packages connected to this HWCE

- SSSTLIB contains `ssstlibs.tar.gz`

- ARR_CONFIG_COMMON contains `acu.tar.gz` (`.acuxebin` ver 7.85.18.0) and `acu.sh`

- DEP_RAID_0_FILE contains pl-acu-r0.ini (this contains the RAID 0 configuration)

Properties for the Deploy Array Configuration RAID 0 HWCE

You can use the CM Portal to review the hardware configuration element as specified in Table 8 on page 61.

Table 16   Properties

Field	Value
Friendly name	Deploy ProLiant Array Configuration [RAID 0]
Version	1.00
Class	_REGULAR_
Type	ACRCONFIG
Instance Number	1
Service OS Needed to Run Method	_SVC_LINUX_
Apply Method	`acu.sh –i pl-acu-r0.ini –internal -reset`
Apply Behavior	_NEVER_
Post Behavior	_REBOOT_
Failure	_CRITICAL_
Post-requisite HWCE	DRIVEMAP
Conditional Pre-requisite HWCE	N/A

10 - Deploy Array Configuration RAID 1

This HWCE deploys the RAID 1 configuration for the target device using the acu utility.

Packages connected to this HWCE

- SSSTLIB contains `ssstlibs.tar.gz`
- ARR_CONFIG_COMMON contains `acu.tar.gz` (`.acuxebin` ver 7.85.18.0) and `acu.sh`
- DEP_RAID_1_FILE contains pl-acu-r1.ini (this contains the RAID 1 configuration)

Properties for the Deploy Array Configuration RAID 1 HWCE

You can use the CM Portal to review the hardware configuration element as specified in Table 8 on page 61.

Table 17   Properties

Field	Value
Friendly name	Deploy ProLiant Array Configuration [RAID 1]
Version	1.00
Class	_REGULAR_
Type	ACRCONFIG
Instance Number	2
Service OS Needed to Run Method	_SVC_LINUX_
Apply Method	acu.sh –i pl-acu-r1.ini –internal -reset
Apply Behavior	_NEVER_
Post Behavior	_REBOOT_
Failure	_CRITICAL_
Post-requisite HWCE	DRIVEMAP
Conditional Pre-requisite HWCE	N/A

11 - Deploy Array Configuration RAID 5

This HWCE deploys the RAID 5 configuration for the target device using the acu utility.

Packages connected to this HWCE

- SSSTLIB contains `ssstlibs.tar.gz`
- ARR_CONFIG_COMMON contains `acu.tar.gz` (`.acuxebin` ver 7.85.18.0) and `acu.sh`
- DEP_RAID_5_FILE contains pl-acu-r5.ini (this contains the RAID 5 configuration)

Properties for the Deploy Array Configuration RAID 5 HWCE

You can use the CM Portal to review the hardware configuration element as specified in Table 8 on page 61.

Table 18   Properties

Field	Value
Friendly name	Deploy ProLiant Array Configuration [RAID 5]
Version	1.00
Class	_REGULAR_
Type	ACRCONFIG
Instance Number	3
Service OS Needed to Run Method	_SVC_LINUX_
Apply Method	`acu.sh -i pl-acu-r5.ini -internal -reset`
Apply Behavior	_NEVER_
Post Behavior	_REBOOT_
Failure	_CRITICAL_
Post-requisite HWCE	DRIVEMAP
Conditional Pre-requisite HWCE	N/A

12 – Reset ProLiant System Configuration [Reset to Factory Defaults] (Shadow HWCE)

This HWCE resets the BIOS configuration of the device to the factory defaults.

Packages connected to this HWCE

- SSSTLIB contains `ssstlibs.tar.gz`
- RBSU_UTILITY contains rbsureset utility version 1.00 from HP

Properties for the Reset ProLiant System Configuration [Reset to Factory Defaults] HWCE

You can use the CM Portal to review the hardware configuration element as specified in Table 8 on page 61.

Table 19   Properties

Field	Value
Friendly name	Reset ProLiant System Configuration [Reset to Factory Defaults]
Version	1.00
Class	_SHADOW_
Type	NVRCONFIG
Instance Number	2
Service OS Needed to Run Method	_SVC_LINUX_
Apply Method	rbusreset -reset
Apply Behavior	_NEVER_
Post Behavior	_REBOOT_
Failure	_CRITICAL_
Post-requisite HWCE	N/A
Conditional Pre-requisite HWCE	N/A

13 – ProLiant Firmware Upgrade [All Firmwares]

This HWCE upgrades the firmware of the target device using the Firmware Maintenance ISO. The Service OS mounts the Firmware Maintenance ISO over http and runs the required firmware upgrade startup script.

Prerequisites

- The Firmware Maintenance ISO. To get this ISO, go to **http://h18023.www1.hp.com/support/files/server/us/romflash.html** and click the link **Click here to download or purchase the HP ProLiant Firmware Maintenance CD**.

  — Once you have the Firmware Maintenance ISO, store it in the `\upload` directory of the CM OS Manager.

  — Change the Apply Method of the HWCE to the name of the Firmware Maintenance ISO. For example:

    `firmware_upgrade.sh` *FileName*`.iso`

- `firmware_upgrade.sh` and Firmware Maintenance CD as input, containing the firmware upgrades.

Packages connected to this HWCE

- FWUPGRADE contains `firmware_upgrade.sh`

- FW_UPGRADE_LIB contains the library `libstdc++-libc6.2-2.so.3` in `fw_upgrade_lib.tar.gz`

Properties for the ProLiant Firmware Upgrade [All Firmwares] HWCE

You can use the CM Portal to review the hardware configuration element as specified in Table 8 on page 61.

Table 20   Properties

Field	Value
Friendly name	ProLiant Firmware Upgrade All
Version	1.00
Class	_REGULAR_

Field	Value
Type	BIOSFLASH
Instance Number	0
Service OS Needed to Run Method	_SVC_LINUX_
Apply Method	`firmware_upgrade.sh *.iso`
Apply Behavior	_NEVER_
Post Behavior	_REBOOT_
Failure	_CRITICAL_
Post-requisite HWCE	N/A
Conditional Pre-requisite HWCE	N/A

14 – ProLiant Firmware Upgrade [Individual]

This HWCE upgrades firmwares of individual peripheral devices (e.g., NIC, Array) using the firmware upgrade .scexe files that are published and connected to this HWCE.

Prerequisite

Publish the required firmware upgrade files (.scexe) and connect them to this HWCE.

Packages connected to this HWCE

- FWUPGRADE_IND contains `fw_upgrade_individual.sh`.

- FW_UPGRADE_LIB contains the library `libstdc++-libc6.2-2.so.3` in `fw_upgrade_lib.tar.gz`.

Properties for the ProLiant Firmware Upgrade [Individual] HWCE

You can use the CM Portal to review the hardware configuration element as specified in Table 8 on page 61.

Table 21   Properties

Field	Value
Friendly name	ProLiant Firmware Upgrade [Individual]
Version	1.00
Class	_REGULAR_
Type	BIOSFLASH
Instance Number	1
Service OS Needed to Run Method	_SVC_LINUX_
Apply Method	`fw_upgrade_individual.sh` `*.`*scexe*
Apply Behavior	_NEVER_
Post Behavior	_REBOOT_
Failure	_CRITICAL_
Post-requisite HWCE	N/A
Conditional Pre-requisite HWCE	N/A

Glossary

Current Hardware Configuration Elements

See Last Remembered State (LRS).

Hardware Configuration object

The Hardware Configuration object contains the information about how a target device's hardware must be configured in order for it to be ready for operating system installation. It contains one or more Hardware Configuration elements.

Hardware Configuration element (HWCE)

A Hardware Configuration Element is an object that identifies the resources and methods to be used to bring a target device the state where it is ready to have an operating system installed.

A Hardware Configuration Element must define:

— The system device or components it applies to.

— The resource files required to manage the element.

— The methods for the operations.

— Sequencing information with respect to dependencies upon other Hardware Configuration Elements.

— Additional Hardware Configuration Elements that must be applied if this Hardware Configuration Elements is applied.

— Information for handling failures, reboot sequencing, nd post-Hardware Configuration Elements completion steps

Last Remembered State (LRS)

A list of the hardware configuration elements previously configured on the device. When operating system management resolution occurs, the Last Remembered State and the new resolved state of hardware configuration elements are compared to determine what hardware configuration elements must be configured.

Also called *Current Hardware Configuration Elements*.

Shadow Hardware Configuration Elements

A Shadow Hardware Configuration Element is an exception to how a device's state is typically managed. A Shadow Hardware Configuration Element is applied to your target device one time and is then deleted. In other words, the operation is run one time on a device, but does not become part of the Current Hardware Configuration Elements (also known as the Last Remembered State).

Index

A

B

C

D

F

H

hardware configuration
    configuring the object, 26
    creating, 26
    modifying, 31
    repairing, 34

Hardware Configuration Class
    about, 52

Hardware Configuration Element Class
    about, 52

Hardware Configuration Elements, 11, 16
    configuring, 20, 62, 63
    connecting to Hardware Configuration, 28
    current, 30, 34, 37
    publishing, 18
    shadow, 37, 38
    successfully applied, 30

Hardware Configuration Management
    about, 11
    sample operations, 11

Hardware Configuration object, 12

HBA. *See* host bus adapter

HBACONFIG field, 44

HBAFLASH field, 44

host bus adapter, 44

HWCE. *See* Hardware Configuration Element

I

IGNORE value, 49, 62, 63, 64, 66, 67, 68, 69, 70, 71, 72, 73, 75, 76

Image Capture and Upload, 63, 64, 69, 70, 71, 72, 73, 74

imgcapt value, 63, 64, 65, 67, 68, 69, 70, 71, 72, 73, 75, 76

Instance Number field, 24, 25, 46, 61, 62, 63, 64, 65, 67, 68, 69, 70, 71, 72, 73, 75, 76

Instance Number, used with apply method, 50

L

last remembered state, 34, 37

LDRIVCFG value, 25

LDRVCONFIG value, 44

legal notices, 2
    copyright, 2
    restricted rights, 2
    trademark, 2
    warranty, 2

logical volumes, 16

LVOL Hardware Configuration Element, 25

LVOLS. *See* logical volumes

M

manufacturing filter, examples, 27

model filter, examples, 27

N

network interface controllers, 44, 51

NEVER value, 24, 25, 48, 63, 64, 66, 67, 68, 69, 70, 71, 72, 73, 75, 76

NIC. *See* network interface controllers

NICCONFIG value, 44

NICFLASH value, 44

NVRAMFLASH_1 value, 49

NVRCONFIG value, 44

O

object, Hardware Configuration, 26

OK value, 48

OS image capture, 62, 64, 65, 66, 67, 68, 75

OS value for RMP field, 44

P

policy, setting, 29

Post Behavior field, 24, 25, 26, 48, 61, 62, 63, 64, 66, 67, 68, 69, 70, 71, 72, 73, 75, 76