

Radia Client Automation Enterprise OS Manager Hardware Configuration Management

For the Windows® operating system

Software Version: 9.00

User Guide

Document Release Date: April 2013

Software Release Date: April 2013



Legal Notices

Warranty

The only warranties for products and services are set forth in the express license or service agreements accompanying such products and services. Nothing herein should be construed as constituting an additional warranty. Persistent Systems 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 Persistent Systems or its licensors required for possession, use or copying. No part of this manual may be reproduced in any form or by any means (including electronic storage and retrieval or translation into a foreign language) without prior agreement and written consent from Persistent Systems.

Copyright Notice

© Copyright 2013 Persistent Systems, its licensors, and Hewlett-Packard Development Company, LP.

Trademark Notices

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

UNIX® is a registered trademark of The Open Group.

Acknowledgements

This product includes software developed by the Apache Software Foundation (<http://www.apache.org/>).

This product includes cryptographic software written by Eric Young (eay@cryptsoft.com).

This product includes software developed by the OpenSSL Project for use in the OpenSSL Toolkit (<http://www.openssl.org/>).

This product includes software written by Tim Hudson (tjh@cryptsoft.com).

This product includes software written by Daniel Stenberg (daniel@haxx.se).

This product includes OVAL language maintained by The MITRE Corporation (oval@mitre.org).

Documentation Updates

The title page of this document 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 of a document, go to:

<http://support.persistentsys.com/>

This site requires that you register for a Persistent Passport and sign in. Register online at the above address.

For more details, contact your Persistent sales representative.

Support

Persistent 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 being able to:

- Search for knowledge documents of interest
- Submit and track support cases and enhancement requests
- Submit enhancement requests online
- Download software patches
- Look up Persistent support contacts
- Enter into discussions with other software customers
- Research and register for software training

To access the Self-solve knowledge base, visit the [Persistent Support](#) home page.

Note: Most of the support areas require that you register as a Persistent Support user and sign in. Many also require an active support contract. More information about support access levels can be found on the [Persistent Support](#) site.

To register for a Persistent Support ID, go to: [Persistent Support Registration](#).

Contents

User Guide	1
Contents	5
Hardware Configuration Management	8
About the Process Flow	8
Audience	9
Prerequisites	9
Terminology	9
Abbreviations and Variables	10
Using Hardware Configuration Management	12
Step 1: Create One or More HWCEs (OS.LME)	12
Step 2: Creating the Hardware Configuration Object (OS.LDS)	18
Step 3: Set Policy	19
Reviewing the Assigned Hardware Configuration Objects	20
Modifying the Hardware Configuration on a Device	20
About the Hardware Configuration Element Class	22
About the Instance Number and Apply Method Attributes	27
About the Hardware Configuration Class	30
Best Practices and Maintenance	32
Best Practices	32
Maintenance	32
Pre-configured Hardware Configuration Elements for Use with HP ProLiant Hardware	34
Importing the HWCEs to the RCA Configuration Server DB	34
About the Pre-configured HW Configuration Management Elements	34
1 – Read ProLiant Hardware Inventory [Server Specific]	35
Packages Connected to this HWCE	35
Properties for the Read ProLiant Hardware Inventory [Server Specific] HWCE	35

2 – Read ProLiant System Configuration	36
Packages Connected to this HWCE	36
Properties for the Read ProLiant System Configuration HWCE	36
3 – Read ProLiant Array Configuration	37
Packages Connected to this HWCE	37
Properties for the Read ProLiant Array Configuration HWCE	37
4 – Read ProLiant iLO Configuration	38
Packages Connected to this HWCE	38
Properties for the Read ProLiant iLO Configuration HWCE	38
5 – Write ProLiant System Configuration	39
Prerequisite	39
Packages Connected to this HWCE	39
Properties for the Write ProLiant System Configuration HWCE	39
6 – Write ProLiant Array Configuration	40
Prerequisite	40
Packages Connected to this HWCE	40
Properties for the Write ProLiant Array Configuration HWCE	40
7 – Write ProLiant iLO Configuration	41
Prerequisite	41
Packages Connected to this HWCE	41
Properties for the Write ProLiant iLO Configuration HWCE	41
8 – Deploy Array Configuration RAID [Default]	42
Packages Connected to this HWCE	42
Properties for the Deploy Array Configuration RAID Default HWCE	42
9 – Deploy Array Configuration RAID 0	43
Packages Connected to this HWCE	43
Properties for the Deploy Array Configuration RAID 0 HWCE	43
10 – Deploy Array Configuration RAID 1	44
Packages Connected to this HWCE	44
Properties for the Deploy Array Configuration RAID 1 HWCE	44
11 – Deploy Array Configuration RAID 5	45
Packages Connected to this HWCE	45

Properties for the Deploy Array Configuration RAID 5 HWCE	45
12 – Reset ProLiant System Configuration [Reset to Factory Defaults] (Shadow HWCE)	46
Packages Connected to this HWCE	46
Properties for the Reset ProLiant System Configuration [Reset to Factory Defaults] HWCE	46
13 – ProLiant Firmware Upgrade [All Firmwares]	46
Prerequisites	47
Packages Connected to this HWCE	47
Properties for the ProLiant Firmware Upgrade [All Firmwares] HWCE	47
14 – ProLiant Firmware Upgrade [Individual]	48
Prerequisite	48
Packages Connected to this HWCE	48
Properties for the ProLiant Firmware Upgrade [Individual] HWCE	48
15 – Additional Sample Deploy Array Configuration HWCE	49
We appreciate your feedback!	50

Chapter 1

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 Radia Client Automation Administrator Publisher to publish the HWCEs to the Radia Client Automation Configuration Server Database (CSDB). The HWCEs are published to the PRIMARY.OS.PACKAGE class.
 - c. Use the Radia Client Automation Administrator CSDB Editor to create and configure the HWCE instances in the PRIMARY.OS.LME class.
 - d. Use the CSDB Editor to connect the PACKAGE instance to the HWCE instance.
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 CSDB Editor 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.
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 Radia Client Automation 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 Client Automation and system administrators who are very familiar with manipulating hardware and system components should use Hardware Configuration Management.

Prerequisites

Radia Client Automation Enterprise 9.00

Terminology

Current Hardware Configuration Elements

See "[Last Remembered State \(LRS\)](#)" below.

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 on 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, and 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 named as *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).

Abbreviations and Variables

Abbreviations Used in this Guide

Abbreviation	Definition
RCA	Radia Client Automation
Core and Satellite	RCA Enterprise environment consisting of one Core server and one or more Satellite servers.
CSDB	Configuration Server Database

Variables Used in this Guide

Variable	Description	Default Values
<i>InstallDir</i>	Location where the RCA server is installed	For a 32-bit OS: C:\Program Files\Hewlett-Packard\HPCA For a 64-bit OS: C:\Program Files (x86)\Hewlett-Packard\HPCA
<i>SystemDrive</i>	Drive label for the drive where the RCA server is installed	C:

Chapter 2

Using Hardware Configuration Management

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).

Caution: If you are using Microsoft's BitLocker, see "Using Microsoft BitLocker" in the *Radia Client Automation Enterprise OS Management Reference Guide*.

Step 1: Create One or More HWCEs (OS.LME)

Step a: Gather the Resources for your HWCEs

The table [Hardware Configuration Element Example Definitions](#) 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 descriptive instance name and friendly name for the element in the CSDB Editor.
- **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.

Note: 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 an HWCE is applied to a target device, some HWCEs might have to be reapplied after the original is applied. For example, if you use an HWCE to change the BIOS firmware, the settings will most likely be reset to their default values. Therefore, you would need to use an 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.

Hardware Configuration Element Example Definitions

HWCE	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 command line listed in the Apply Method column initiates the specified resource files. 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 ["About the Hardware Configuration Element Class"](#) on page 22 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 Files for HWCEs

In this section, you will use the Radia Client Automation Administrator Publisher to publish the HWCEs to the Radia Client Automation Configuration Server Database.

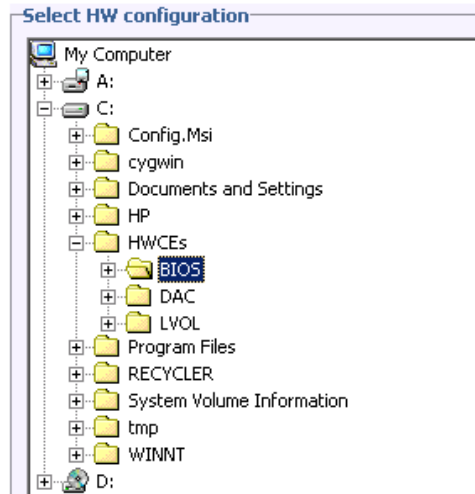
Note: Before you publish your HWCEs, gather your resource files into a single folder.

To publish an HWCE:

1. Go to **Start > All Programs > Radia Client Automation Administrator > Client Automation Administrator Publisher**.
For details on how to use the Publisher, see the *Radia Client Automation Enterprise Administrator User Guide*.
2. Type your User ID and Password.
3. From the Publishing Options drop-down list, select **HW Configuration**.
4. Click **OK**.
5. Select the folder that contains the resources needed to create your HWCE. In our example, we selected `C:\HWCEs\BIOS`.

Caution: 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. Also, all the files you want to publish need to be located under `C:\HWCEs\BIOS`. The Publisher does not look into subdirectories of `C:\HWCEs\BIOS`.

HW Configuration



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**.
9. Review the information and then click **Publish**. The package resources will be published in a non-compressed format.
10. When the Publisher is done, click **Finish**.
11. Click **Yes** to confirm that you want to close the Publisher.

Use the CSDB Editor to view the package that has been created in `PRIMARY.OS.PACKAGE`.

Repeat the above steps to create packages for the Disk Array Controller (DAC) and the Logical Volumes (LVOL). Make 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 the previous section, we have *only* created the resource packages that will be downloaded as part of the hardware configuration management actions.

We still need to create LMEs and/or LDS instances.

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 "[About the Hardware Configuration Element Class](#)" on page 22 for detailed information about the fields and the information you provide into the fields.

To Configure the Hardware Configuration Element:

1. Go to **Start > All Programs > Radia Client Automation Administrator > Client Automation Administrator CSDB Editor**.
2. Log on as the administrator (by default, the user ID is admin and the password is secret).
3. Go to PRIMARY.OS.LME.
4. Right-click the class name and select **New Instance**.
5. Type the display name and the instance name.
When creating the HWCE instances, you may want to use the following Instance and Friendly Names.

For the BIOS, set the:

- Display Name to **Pro32 WS Bios Rev 1.00**
- Instance Name to **P32_BIOS_100**

For the Disk Array Controller, set the:

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

For the Logical Volume, set the:

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

6. Click **OK**.
7. Double-click the instance that you want to configure.

8. Enter the information needed to configure the BIOS element as described in the table [Configuring the BIOS HWCE](#).

See "[About the Hardware Configuration Element Class](#)" on page 22 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

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

9. Click **OK**.
10. Connect the PACKAGE instance that you created in step b to the PRIMARY.OS.LME instance that you are creating in this step.

Caution: Never connect a PACKAGE instance other than the ones created using the HWCE Publisher. You cannot connect an OS package, a SYSPREP package, a DRIVEMAP or BEHAVIOR instance or any other package instance that deals with OS

deployment in itself.

- Repeat the steps above to configure the HWCEs for the Disk Array Controller (DAC) and the Logical Volumes (LVOL). Use tables [Configuring the DAC HWCE](#) and [Configuring the LVOL HWCE](#) as a guide for the information to be entered.

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

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_
Post Behavior	_CONTINUE_
Failure	_CRITICAL_

Field	Value
Post-requisite HWCE	DRIVEMAP
Conditional Pre-requisite HWCE	N/A

Step 2: Creating the Hardware Configuration Object (OS.LDS)

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

To create the hardware configuration:

1. Go to **Start > All Programs > Radia Client Automation Administrator > Client Automation Administrator CSDB Editor**.
2. Log on as the administrator (by default, the user ID is admin and the password is secret).
3. Go to PRIMARY.OS.LDS.
4. Right-click HW Config (LDS) and select **New Instance**.
5. Type the display name, such as `Pro32 WS CAD Setup Rev 1.00` and the Instance name, such as `P32_CAD_100`.
6. Click **OK** and then double-click on the instance.
7. Use the MANUFLTR (Reg Expr to Filter Manufacturer) field *and* the MODLFLTR (Reg Expr to Filter Model) field to restrict the hardware that the hardware configuration is applied to.

Caution: You *must* type 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 or newer).

- a. Double-click the MANUFLTR (Reg Expr to Filter Manufacturer) attribute and type a regular Tcl expression to indicate the manufacturer that you want to apply this hardware configuration to. If you do not provide 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. *It does 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 MODLFLTR (Reg Expr to Filter Model) attribute, 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. *It does 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.* D1360.* PROLIANT.*DL360.*
```

or

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

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

Caution: 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.

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

8. If you want to track the versions of your configuration elements use the VERSION attribute.
9. Click **OK**.

To connect the HWCEs to the Hardware Configuration Package:

1. Go to **Start > All Programs > Radia Client Automation Administrator > Client Automation Administrator CSDB Editor**.
2. Log on as the administrator (by default, the user ID is admin and the password is secret).
3. Go to PRIMARY.OS.LME and drag and drop the instances to the HW Config (LDS) instance. In this example, use the three elements created above.

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 OS Manager, see “About Policy” in the *Radia Client Automation Enterprise OS Management Reference Guide*.

Note: Policy can be assigned using the OS Deployment Wizard in the Enterprise console or through the use of the System Explorer directly in the CSDB.

To set policy using the CSDB:

1. Go to **Start > All Programs > Radia Client Automation Administrator > Client Automation Administrator CSDB Editor**.
2. Log on as the administrator (by default, the user ID is admin and the password is secret).
3. Go to the LDS instance and drag and drop it on the appropriate policy class instance, typically a Machine Manufacturer or Machine Model instance.

To set policy using the Enterprise console:

Use the OS Deployment Wizard to assign a Hardware Configuration Object to the target device. See “About Policy” in the *Radia Client Automation Enterprise OS Management Reference Guide*.

Reviewing the Assigned Hardware Configuration Objects

After an RCA OS connect has occurred and the hardware configuration has been deployed to a target device, you can review the Assigned Hardware Configuration Objects that have been successfully applied. To do this, go to the Device Properties sheet in the Enterprise console.

Device Properties

The screenshot displays the 'Device Properties' window in the Enterprise console. On the left is a navigation pane with 'Properties' selected. The main area is divided into three sections:

- Information:** A header section with the text 'All properties for this directory object are listed below.'
- Device Summary:** A section containing a monitor icon with a question mark and a list of properties:

DNS Hostname:	15.2.113.129
Operating System:	
Service Pack:	
System Manufacturer:	
System Product Name:	
System Serial Number:	
IP Address:	
MAC Address:	
- OS Management:** A section showing:

OS State:	Unknown
Assigned Operating System:	WINVISTA
Assigned Hardware Configuration Objects:	P32_CAD_100

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. Use the CSDB Editor to make any necessary modifications to the HWCE for the updated firmware. See *To configure the Hardware Configuration Element* in "[Step 1: Create One or More HWCEs \(OS.LME\)](#)" on page 12 for details.
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 an RCA OS connect, it detects the hardware configuration and applies the tasks connected to this hardware configuration.
6. In the Enterprise console, go to the Device Properties sheet for the target device to see the Current Hardware Configuration again. Notice that the new Hardware Configuration object, P32_CAD_110, is listed. Access the ROM object to see 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.

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.

Appendix A

About the Hardware Configuration Element Class

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

Hardware Configuration Element Class

Name	Attribute Description	Default Value	Potential Values
Name (required)	Friendly name	N/A	N/A
Version	A user-defined version number. For reference only.	1.00	N/A
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.
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 an HWCE can be	N/A	Examples of user-defined types BIOSFLASH Flashes the BIOS of a device. NVRCONFIG Configures the non-volatile RAM of a system. The

Name	Attribute Description	Default Value	Potential Values
	<p>applied to, then you can set more than one HWCE to the same type and set the INSTANCE attribute to 0.</p> <p>See " Instance Number (0 -> apply to all) (required)" on next page and " About the Instance Number and Apply Method Attributes" on page 27.</p>		<p>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.</p> <p>ACRFLASH</p> <p>Flashes the firmware of an array controller.</p> <p>ACRCONFIG</p> <p>Configures the array controller with respect to RAID volumes that are built from raw local disks or raw fiber channel LUNs (==FC disk drives).</p> <p>LDRVCONFIG</p> <p>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.</p> <p>NICFLASH</p> <p>Flashes the firmware of network interface controllers (NIC).</p> <p>NICCONFIG</p> <p>Configures any programmable elements of NICs.</p> <p>HBAFLASH</p> <p>Flashes the firmware of a SCSI or fiber channel host bus adapter (HBA) (disk controller).</p>

Name	Attribute Description	Default Value	Potential Values
			<p>HBACONFIG Configures SCSI or fiber channel host bus adapter with regard to getting raw disks or FC LUNs (disks) assigned.</p> <p>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 uploaded to the OS Manager.</p> <p>Reserved types <i>Do not use the OS or DRIVEMAP types for your HWCEs because they are reserved for internal use only.</i></p> <p>OS Causes an OS installation.</p> <p>DRIVEMAP Causes an OS installation.</p>
<p>Instance Number (0 - > apply to all) (required)</p>	<p>A unique number used to establish a convention between the apply methods and the application of the apply methods to a set of HWCEs.</p> <p>Information in this field does not do anything until the person writing the apply method scripts uses it to establish conventions.</p> <p>See " About the Instance Number and Apply Method Attributes" on page 27 for more information.</p>	<p>0</p>	<p>0 or greater depending on the apply method scripts that you write.</p> <p>Typically, the value of 0 is used to indicate that you want an HWCE to be applied to <i>all identical components</i> 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.</p> <p>If you set this to 1, the apply method may be</p>

Name	Attribute Description	Default Value	Potential Values
			<p>applied to only one hardware configuration element. Again, the apply method script must support this.</p> <p>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.</p> <p>See "About the Instance Number and Apply Method Attributes" on page 27 for more information.</p>
Service OS Needed to Run Method	System defined service OS identifier.	_SVC_LINUX_	<p>_SVC_LINUX_ to run the Linux Service Operating System.</p> <p>_SVC_PEX86 to run the WinPE 2.0 Service Operating System.</p> <div style="background-color: #f0f0f0; padding: 5px;"> <p>Note: Be sure that you set the Service OS for each HWCE to ensure it will run in the required Service Operating System.</p> </div>
Apply Method (required)	<p>A command line that initiates the HWCE.</p> <p>Note that text files must be in the UNIX text file format.</p> <p>Also, 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,</p>	N/A	N/A

Name	Attribute Description	Default Value	Potential Values
	<p>redirecting the output would ensure that the user sees the message before the reboot.</p> <p>See " About the Instance Number and Apply Method Attributes" on next page for more information.</p>		
Apply Behavior (required)	<p>Defines the reboot behavior on completion of the HWCE's apply method.</p> <p>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 <code>_NEVER_</code>.</p>	<code>_NEVER_</code>	<p><code>_ALWAYS_</code> The apply method always triggers a reboot after the operation without returning control to the RCA OS Manager User Agent.</p> <p><code>_OK_</code> After success, the apply method triggers a reboot without returning control to the RCA OS Manager User Agent only.</p> <p><code>_NEVER_</code> The apply method never reboots and always returns control to the RCA OS Manager User Agent.</p> <p><code>_ERROR_</code> <i>For future use.</i></p>
Post Behavior (required)	<p>After an HWCE has been processed successfully, this instructs the machine being configured what to do next with regards to its boot state.</p>	<code>_CONTINUE_</code>	<p><code>_REBOOT_</code> On successful completion of an apply behavior operation, this triggers a reboot of system.</p> <p><code>_CONTINUE_</code> On 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.</p>
Failure (required)	<p>The command or shell script returned an error and this indicates how to handle that error.</p>	<code>_CRITICAL_</code>	<p><code>_CRITICAL_</code> Any failure of this HWCE operation is considered</p>

Name	Attribute Description	Default Value	Potential Values
			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 an 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.
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 an 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 RCA 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 started 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.

Appendix B

About the Hardware Configuration Class

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

The Hardware Configuration Class

Name	Attribute Description		
Stop Expressions	For internal use only.	N/A	N/A
Connections	Allows you to manually create connections to hardware configuration elements.	N/A	N/A
Name	The name of the Hardware Configuration.	N/A	N/A
ZSVCPRI	Service Create Ordering For internal use only.	N/A	N/A
Version	A user-defined version number. For reference only.	1.00	N/A
MANUFLTR	Reg Expr to Filter Manufacturer Restricts the hardware that the hardware configuration is applied to based on manufacturer. You must provide information in the filter fields as a regular Tcl expression otherwise the hardware configuration will never be applied to your target device. It recommends that you create a thorough plan for the proper filters. See " Step 2: Creating the Hardware Configuration Object (OS.LDS) " on page 18 for detailed information.	""	Any regular Tcl expression
MODLFLTR	Reg Expr to Filter Model Restricts the hardware that the hardware configuration is applied to based on model. You must provide information in the filter fields as a regular Tcl expression otherwise the hardware configuration will never be applied to your target device. It recommends that you create a thorough plan for the proper filters. See " Step 2: Creating the Hardware Configuration Object (OS.LDS) " on page 18 for detailed information.	""	Any regular Tcl expression

Appendix C

Best Practices and Maintenance

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.

Maintenance

If you receive a fix from CPE, place it in

`\IntegrationServer\OSM\SOS\linux\testing\work` and then reboot the machine.

Appendix D

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 RCA Configuration Server DB.

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 RCA Configuration Server DB

To import the resources for the HWCEs:

1. From the Radia Client Automation 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 RCA Core server installed. There is no need to stop the Core server; the import uses RADDBUTIL.
2. Run `ImportLME.cmd`. This script populates the RCA Configuration Server DB with the new HWCEs. All of the required resources and `.sh` files are provided.
3. Use the Configuration Server DB Editor (CSDB Editor) 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 `<InstallDir>/OSManagerServer/upload` directory on the RCA Core server.

Note: 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 CSDB Editor to review the hardware configuration element as specified in the following table:

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

Field	Value
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 `<InstallDir>/OSManagerServer/upload` directory on the RCA Core server.

Note: 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 CSDB Editor to review the system configuration element as specified in the following table:

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	<code>conrep.sh -s system.xml</code>
Apply Behavior	_NEVER_
Post Behavior	_CONTINUE_
Failure	_CRITICAL_

Field	Value
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 RCA OS Manager Server.

Note: 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

- SSSTLIB contains `sstlibs.tar.gz`
- ARR_CONFIG_COMMON contains `acu.tar.gz` (having `.acuxebin` utility, version 8.50-6.0) and `acu.sh`

Properties for the Read ProLiant Array Configuration HWCE

You can use the Radia Client Automation Portal to review the Hardware Configuration Element as specified in "Properties for the Read ProLiant Hardware Inventory [Server Specific] HWCE" on page 35.

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	<code>acu.sh -c array.ini</code>

Field	Value
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 RCA OS Manager Server.

Note: 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:

```
hponcfg.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 CSDB Editor to review the hardware configuration element as specified in "Properties for the Read ProLiant Hardware Inventory [Server Specific] HWCE" on page 35.

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	<code>hponcfg.sh -w iLO.xml</code>

Field	Value
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.

Note: 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 -lRequiredFileName.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 CSDB Editor to review the hardware configuration element as specified in "Properties for the Read ProLiant Hardware Inventory [Server Specific] HWCE" on page 35.

Properties

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

Field	Value
Instance Number	1
Service OS Needed to Run Method	_SVC_LINUX_
Apply Method	<code>conrep.sh -l *.xml</code>
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.

Note: 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 -iRequiredFileName.ini -internal -reset
```

Packages Connected to this HWCE

- SSSTLIB contains `ssstlibs.tar.gz`
- `ARR_CONFIG_COMMON` contains `acu.tar.gz` (utility `.acuxebin ver 8.50-6.0`) and the script `acu.sh`

Properties for the Write ProLiant Array Configuration HWCE

You can use the CSDB Editor to review the hardware configuration element as specified in "Properties for the Read ProLiant Hardware Inventory [Server Specific] HWCE" on page 35.

Properties

Field	Value
Friendly name	Write ProLiant Array Configuration
Version	1.00

Field	Value
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.

Note: 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 CSDB Editor to review the hardware configuration element as specified in "Properties for the Read ProLiant Hardware Inventory [Server Specific] HWCE" on page 35.

Properties

Field	Value
Friendly name	Write ProLiant iLO Configuration

Field	Value
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 8.50-6.0) 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 CSDB Editor to review the hardware configuration element as specified in "Properties for the Read ProLiant Hardware Inventory [Server Specific] HWCE" on page 35.

Properties

Field	Value
Friendly name	Deploy ProLiant Array Configuration [Default]
Version	1.00

Field	Value
Class	_REGULAR_
Type	ACRCONFIG
Instance Number	0
Service OS Needed to Run Method	_SVC_LINUX_
Apply Method	<code>acu.sh -i pl-acu-rd.ini -internal -reset</code>
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 8.50-6.0) 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 CSDB Editor to review the hardware configuration element as specified in "Properties for the Read ProLiant Hardware Inventory [Server Specific] HWCE" on page 35.

Properties

Field	Value
Friendly name	Deploy ProLiant Array Configuration [RAID 0]
Version	1.00
Class	_REGULAR_
Type	ACRCONFIG
Instance Number	1

Field	Value
Service OS Needed to Run Method	<code>_SVC_LINUX_</code>
Apply Method	<code>acu.sh -i pl-acu-r0.ini -internal -reset</code>
Apply Behavior	<code>_NEVER_</code>
Post Behavior	<code>_REBOOT_</code>
Failure	<code>_CRITICAL_</code>
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 8.50-6.0) 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 CSDB Editor to review the hardware configuration element as specified in "Properties for the Read ProLiant Hardware Inventory [Server Specific] HWCE" on page 35.

Properties

Field	Value
Friendly name	Deploy ProLiant Array Configuration [RAID 1]
Version	1.00
Class	<code>_REGULAR_</code>
Type	ACRCONFIG
Instance Number	2
Service OS Needed to Run Method	<code>_SVC_LINUX_</code>
Apply Method	<code>acu.sh -i pl-acu-r1.ini -internal -reset</code>
Apply Behavior	<code>_NEVER_</code>

Field	Value
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 8.50-6.0) 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 CSDB Editor to review the hardware configuration element as specified in "Properties for the Read ProLiant Hardware Inventory [Server Specific] HWCE" on page 35.

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	<code>acu.sh -i pl-acu-r5.ini -internal -reset</code>
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_UTILTY contains `rbsureset` utility version 1.00 from HP

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

You can use the CSDB Editor to review the hardware configuration element as specified in "Properties for the Read ProLiant Hardware Inventory [Server Specific] HWCE" on page 35.

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	<code>rbusreset -reset</code>
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](#).
- After you have the Firmware Maintenance ISO, store it in the `\upload` directory of the 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 CSDB Editor to review the hardware configuration element as specified in "Properties for the Read ProLiant Hardware Inventory [Server Specific] HWCE" on page 35.

Properties

Field	Value
Friendly name	ProLiant Firmware Upgrade All
Version	1.00
Class	_REGULAR_
Type	BIOSFLASH
Instance Number	0
Service OS Needed to Run Method	_SVC_LINUX_
Apply Method	<code>firmware_upgrade.sh *.iso</code>
Apply Behavior	_NEVER_
Post Behavior	_REBOOT_
Failure	_CRITICAL_
Post-requisite HWCE	N/A

Field	Value
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 CSDB Editor to review the hardware configuration element as specified in "Properties for the Read ProLiant Hardware Inventory [Server Specific] HWCE" on page 35.

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	<code>fw_upgrade_individual.sh *.scexe</code>
Apply Behavior	_NEVER_
Post Behavior	_REBOOT_
Failure	_CRITICAL_
Post-requisite HWCE	N/A
Conditional Pre-requisite HWCE	N/A

15 – Additional Sample Deploy Array Configuration HWCE

OS.LDS.PRO380_CAD_HW_REV1_RAID5

OS.LDS.PRO380_CAD_HW_REV2_RAID5

OS.LDS.PRO380_CAD_HW_REV3_RAID5

These are sample LDS+LME+PACKAGE combinations that were run against the following hardware;

DL380 G6

- Controller HP Smart Array P410i, Firmware Version 3.00, License Keys Supported
- Port 0, with Array A and Array B available

These are provided as-is.

We appreciate your feedback!

If an email client is configured on this system, by default an email window opens when you click [here](#).

If no email client is available, copy the information below to a new message in a web mail client, and then send this message to radiadocfeedback@persistent.co.in.

Product name and version: Radia Client Automation Enterprise OS Manager Hardware Configuration Management, 9.00

Document title: User Guide

Feedback:

User Guide

We appreciate your feedback!
