

HP Client Automation Enterprise

Agents

for Linux operating systems

Software Version: 7.20

Migration Guide

Document Release Date: July 2008

Software Release Date: July 2008



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

© Copyright 2008 Hewlett-Packard Development Company, L.P.

Trademark Notices

Linux is a registered trademark of Linus Torvalds.

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

OpenLDAP is a registered trademark of the OpenLDAP Foundation.

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

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

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

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

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

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

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 a support contract.

To find more information about access levels, go to:

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

To register for an HP Passport ID, go to:

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

Contents

1	Upgrading Your HPCA Agents	6
	HPCA agent Upgrade Methods	6
	Upgrading the Agent for Linux as a HPCA Service.....	6
	Importing the Classes, Instances, and Resources.....	6
	The UPGRADE Class	8
	ZSTOP Expressions	8
	Always Connection	9
	Create Method (ZCREATE)	9
	Verify Method (ZVERIFY)	9
	Update Method (ZUPDATE).....	9
	Media Directory (MEDIADIR).....	9
	Retry Attributes (ATTEMPTS and INTERVAL).....	9
	Command Line (CMDLINE).....	10
	Configuring the Agent Upgrade.....	11
	To configure the agent Upgrade	11
	Customizing the Agent Upgrade Services	13
	Adding the UPGRADE Class to the Master Connection Rules	13
	Mechanics of the agent Upgrade for UNIX when applied as an HPCA Service	16
	Troubleshooting the agent Upgrade for UNIX.....	16
	Upgrading the agent to Version 7.20 – from DVDROM.....	17
	Index	19

1 Upgrading Your HPCA Agents

Use the information in this document to upgrade your Linux HPCA agents to version 7.20. Refer to any Linux HPCA agent guide for additional details.

HPCA agent Upgrade Methods

HP offers several ways to upgrade your agents:

- **Use the HPCA Agent Upgrade service (recommended)**
Use the Agent Upgrade service to distribute the agent upgrade as an HPCA service.
- **Use the standard installation media**
Use the agent upgrade installation media if you want to upgrade a machine in your lab environment. The HPCA agent upgrade contains the standard installation files, as well as an upgrade file. When you run the upgrade, a standard installation is performed; or, if it detects the existence of a previous agent version, it launches the upgrade. Existing agent object attributes are maintained and the agent version information is updated to reflect the latest version.
 - For information about the standard installation, refer to the Linux versions of the *HP Client Automation Application Manager Guide and Application Self-service Manager Guide*, or the *HP Client Automation Inventory Manager Guide*.

Upgrading the Agent for Linux as a HPCA Service

This section describes how to perform an agent upgrade for Linux using the following export decks. These are included on the HPCA DVD in the Agents\migration directory.

- `UPGRADE.XPC`
This is a class export of a new class needed for the agent upgrade. It contains the definition of the `PRIMARY.SOFTWARE.UPGRADE` Class
- Agent operating system-specific instance and resource agent upgrade materials

For RedHat and SuSE Linux:

- `HPCA_AGENT_72_UPGRADE_LINUX.XPI`
- `HPCA_AGENT_72_UPGRADE_LINUX.XPR`

Importing the Classes, Instances, and Resources

In order to distribute the agent upgrade as an HPCA service, you must import the agent **UPGRADE** Class definition as well as upgrade service data into your HP Client Automation Configuration Server Database (HPCA-CSDB).

To import the required classes, instances, and resources

- 1 Stop the HPCA Configuration Server.
- 2 Transfer the appropriate export decks (as listed above) into the Configuration Server binary/executable directory.

SystemDrive:\Program Files\Hewlett-Packard\Configuration Server\bin, if your Configuration Server is running on a Microsoft Windows platform.

or

/opt/HP/CM/ConfigurationServer/exe, if your Configuration Server is running on a Linux platform



If you need to ftp the XPI and XPR files from one machine to another, XPR files are binary and should be transferred in binary mode, while XPI files are ASCII and should be transferred in ASCII mode.

- 3 Change your current working directory to the Configuration Server binary/executable directory where you just copied the export decks.

- 2 Import the UPGRADE class by running the following command:

```
ZEDMAMS VERB=IMPORT_CLASS,FILE=UPGRADE.XPC,PREVIEW=NO
```

- 4 Review the `zedmams.log` to ensure that there were no errors, and that your import completed with RC 0.

- 5 Next, import the required instances by running the following command:

```
ZEDMAMS VERB=IMPORT_INSTANCE,FILE=HPCA_AGENT_72_UPGRADE_LINUX.XPI,PREVIEW=NO
```

- 6 Review the `zedmams.log` to ensure that there were no errors, and that your import completed with RC 0.

- 3 Next, import the required resources by running the following command:

```
ZEDMAMS VERB=IMPORT_RESOURCE,FILE=HPCA_AGENT_72_UPGRADE_LINUX.XPR,PREVIEW=NO
```

- 7 Review the `zedmams.log` to ensure that there were no errors, and that your import completed with RC 0.

Repeat the steps above for each of the operating system specific XPI and XPR files relevant to your enterprise have been imported.

- 8 When you are finished with all required imports, start the Configuration Server.

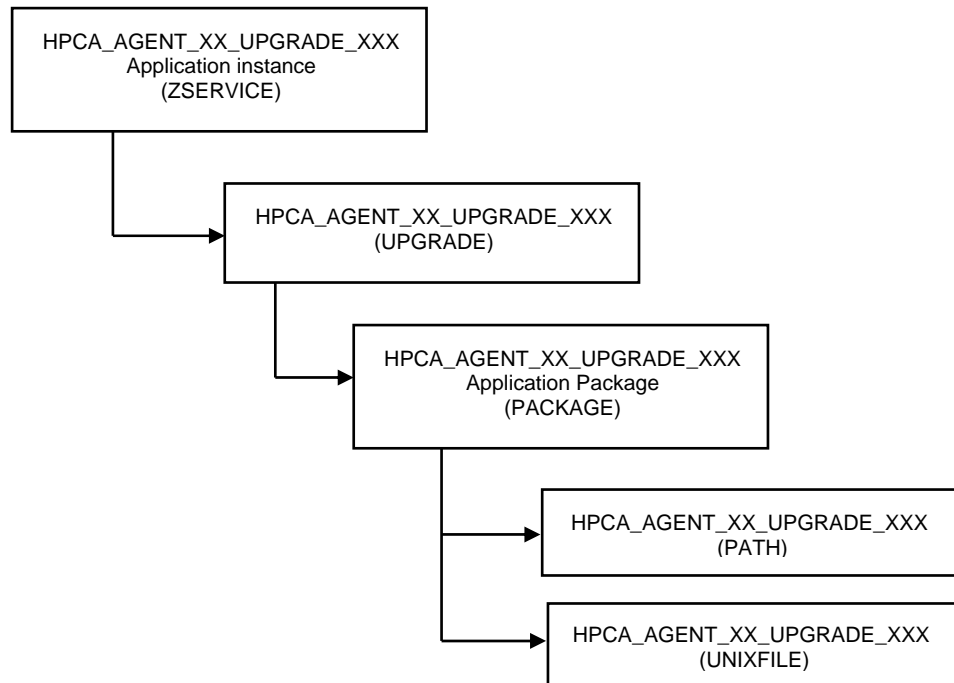
Use the HPCA Admin CSDB Editor to view the new UPGRADE Class and service specific data. Upon the successful completion of the above steps, you will find:

- A new class called **UPGRADE** within the PRIMARY.SOFTWARE Domain.
- Sample agent upgrade Application Services within your SOFTWARE.ZSERVICE Class.
- Sample UPGRADE instance data within the PRIMARY.SOFTWARE.UPGRADE Class.
- Sample PACKAGEs within the PRIMARY.SOFTWARE Domain.

These imported services are mandatory services (distributed to end agents via the Application Manager).

The supplied agent Upgrade Application Services are modeled per [Figure 1](#) on page 8.

Figure 1 Model for the agent upgrade service



HP provides the following sample service that can be used to perform an upgrade of agents version 3.x, 4.x, or 5.x to the current HPCA Version:

- **HPCA_AGENT_72_UPGRADE_LINUX**
Installation of this service would upgrade a pre-existing Red Hat Linux (Intel architecture) or SuSE agent to the current version.

All of the configurable parameters pertaining to the agent upgrade are specified in the UPGRADE instance in the Configuration Server entitlementment model.

The UPGRADE Class

The UPGRADE Class instance is connected between the upgrade SERVICE and PACKAGE instances. It is responsible for the resolution of the agent upgrade materials as well as for launching the agent upgrade process. All upgrade configuration options are defined within the UPGRADE Class instance.

ZSTOP Expressions

The sample UPGRADE instances contain multiple ZSTOP expressions that identify the specific operating system for which the UPGRADE applies and for the Application Manager Agent version. The purpose of these expressions is to gate the resolution for a specific operating system and to prevent the resolution for an agent that has been upgraded already.

For example, in the SOFTWARE.UPGRADE.HPCA_AGENT_72_UPGRADE_LINUX instance, the first ZSTOP expression `WORDPOS (EDMGET (ZMASTER.ZOS) , 'UNIXLINUX') =0`

causes the resolution to continue only if the agent operating system is Red Hat Linux or SuSE Linux.

The other ZSTOP expressions cause the resolution to continue only if the ZPKGRELI variable in the current Application Managers agent ZMASTER object contains a value less than the current version. This expression would stop the deployment of the agent upgrade to agents that are already running the latest version.

You can find these same expressions in the sample SERVICE instances provided.

Always Connection

The ALWAYS Connection in the upgrade instance creates an association with the operating system-specific package that contains the agent upgrade materials. (For example, the SOFTWARE.UPGRADE.HPCA_AGENT_72_UPGRADE_LINUX instance has an Always Connection of: SOFTWARE.PACKAGE.HPCA_AGENT_72_UPGRADE_LINUX, the package that contains the Linux Agent upgrade materials). The value of this attribute in any UPGRADE instance should contain the name of the PACKAGE instance that is prefixed with the operating system-specific designator that matches the intended target operating system.

Create Method (ZCREATE)

The sample UPGRADE instance contains a create method that causes the upgrade script to be launched with the values specified in the UPGRADE Class instance attributes. This method is called only when the agent upgrade materials are initially installed on disk.

Verify Method (ZVERIFY)

The sample UPGRADE instance contains a verify method that causes the upgrade script to be launched with the values specified in the UPGRADE Class instance attributes. This method is only called after a successful verify. If the agent upgrade was not able to complete successfully when the upgrade materials were installed due to disk space, agent in use, or for any reason, this would initiate the upgrade process again.

Update Method (ZUPDATE)

The sample UPGRADE instance contains an update method value that causes the upgrade script to be launched with the values specified in the UPGRADE Class instance attributes. This method is called after a successful update. If the agent upgrade was not able to complete successfully when the upgrade materials were first installed, and updated agent upgrade materials are available in the connecting Configuration Server, the update method would be launched after the successful installation of those updated materials, initiating the upgrade process again.

Media Directory (MEDIADIR)

The media directory attribute MEDIADIR contains the value of `&(ZMASTER.ZSYSDIR)/ClientUpgrade`. This is the default location where the upgrade process expects to find the agent upgrade materials, when they are downloaded to the agent workstation.

Retry Attributes (ATTEMPTS and INTERVAL)

When the agent upgrade service finds the agent is in use, it will retry the upgrade a number of times at a specified interval. The ATTEMPTS attribute sets the number of times to retry the

upgrade, while the `INTERVAL` attribute specifies the number of seconds the upgrade waits between retry attempts. The default values for both the `ATTEMPTS` and `INTERVAL` attributes is 60.

A value of zero in the `ATTEMPTS` attribute will cause the upgrade process to retry indefinitely.

Command Line (CMDLINE)

The command line attribute, `CMDLINE`, is used to specify the command line arguments to be passed to the agent upgrade process. Because all arguments specified after the `-c` option are passed directly to the agent installer, any arguments that can be specified on the agent install command line can be specified here except for the `-mode` argument which is automatically set by the upgrade script to perform a silent install/upgrade.

The leading text `-c` in the command line attribute called `CMDLINE` is required.

► The command line arguments following the `-c` option must be placed inside double quotes.

The `-installcomp` command line option is used to specify the specific products to be upgraded/installed on the agent workstation.

Valid agent components include:

- ram - Application Manager
- rim - Inventory Manager
- rsm - Application Self-service Manager
- svr - Server Management
- patch - Patch Manager
- rom - OS Manager

For example, to upgrade your Application Manager and also install the Inventory Manager and Application Self-service Manager, you would specify `"-installcomp ram,rsm,rsm"`.

► When multiple products are specified the HPCA product acronyms must be comma separated.

The `-mgrip` and `-mgrport` command line options specify the Configuration Server IP address and port that the upgraded agent will use for its `ZMASTER` object and sample shell scripts. Should you upgrade or install the Application Self-service Manager the configuration file called `args.xml` would be populated with Configuration Server IP address and port number specified with the `-mgrip` and `-mgrport` command line options. These parameter values do not have to be the same value as the Configuration Server IP and port, from which the agent is receiving the upgrade service. The default value for `-mgrport` is 3464.

► Sample UPGRADE instances specify a `CMDLINE` attribute value of: `-c "-instcomp ram,rsm -mgrip rcs001 -mgrport 3464"` which would upgrade Application Manager and Inventory Manager agent components and pre-configure the upgraded agent to receive new applications from the Configuration Server called `rcs001` running on port 3464.

Therefore, please be sure to update the value of the UPGRADE instance `CMDLINE` attribute to reflect your preferred Configuration Server ip and port.

You may use symbolic substitution for the values passed via the `-mgrip` and `-mgrport` command line arguments, providing the Configuration Server from which the agent receives the agent upgrade Service is the same as Configuration Server from which you would like the upgraded agent receive future Application Services. For example, the `UPGRADE.CMDLINE` attribute may be configured to read:

```
-c "-instcomp ram,rim,rsm -mgrip &(ZMASTER.ZIPADDR) -mgrport  
&(ZMASTER.ZDSTSOCK)
```

This would upgrade Application Manager, Inventory Manager, and Application Self-service Manager on the agent receiving the service and pre-configure the upgraded agent to receive new applications from the Configuration Server from which it installed the Agent Upgrade Service.


Minimally, we recommend the use of the following upgrade command line options `-instcomp`, `-mgrip`, and `-mgrport`.

You can append the text `-loglevel 9` in the command line to set maximum logging for the agent upgrade/install. This command line option is typically used for diagnostic purposes.

Instead, you may specify a command line to use an agent install configuration file.

For information about installer configuration files (created by the Remote Installation Setup process of the agent installer), please consult the agent installation chapter of an HPCA agent guide for Linux.

Should you have an installation configuration file already on the agent workstations, (created by the Remote Installation Setup process of the agent installer), you can have the agent upgrade use this configuration file as a basis to perform a agent upgrade by specifying `-cfg configfilename.cfg` in the `CMDLINE` attribute, where `configfilename.cfg`, specifies a fully qualified path name to a configuration file.

 We do not recommend that you specify both a configuration file and command line options `-installcomp`, `-mgrport`, or `-mgrip` because the command line options will override the values specified in the configuration file.

For example, the command line specification of `-c "-loglevel 9 -cfg /tmp/myown.cfg"` would rely exclusively on the variables set in the `/tmp/myown.cfg` configuration file to determine what HPCA components to upgrade/install and what Configuration Server IP and port to use as well as other upgrade/install options.

Configuring the Agent Upgrade

To configure the agent Upgrade

- 1 In the HPCA Admin CSDB Editor, navigate to the HP supplied Application Service pertinent to the target operating system for which you intend to apply the agent upgrade.
For example, `SOFTWARE.ZSERVICE.HPCA_AGENT_72_UPGRADE_LINUX`. The Service instance is connected to an `UPGRADE` Class instance that contains the configuration options relevant to the agent upgrade.
- 2 Go to the `UPGRADE` instance (`PRIMARY.SOFTWARE.UPGRADE`) for example, `HPCA_AGENT_72_UPGRADE_LINUX`.
- 3 In the right-hand pane double-click **CMDLINE**. This allows you to define specific agent upgrade options and behaviors. This attribute is used to specify the agent components you wish to upgrade/install as well as your preferred Configuration Server IP address and port number.

- 4 In the Editing Instance dialog box, edit the value to include all the agent components you would like to upgrade as well as your preferred Configuration Server IP address and port number.

Valid agent components include:

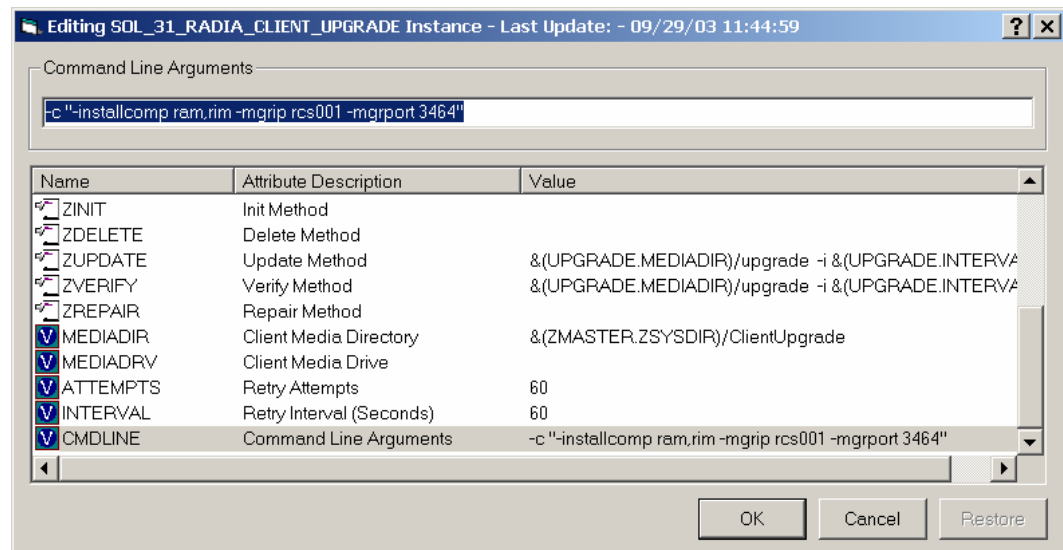
- ram - Application Manager
- rim - Inventory Manager
- rsm - Application Self-service Manager
- svr - Server Management
- patch - Patch Manager
- rom - OS Manager

► The command line arguments following the `-c` option must be placed inside double quotes.

When multiple agent components are specified the HPCA product acronyms must be comma separated.

For example, to specify upgrades for the Application Manager, and Inventory Manager and have these upgraded agent components configured to connect to a Configuration Server called **rcs001** running on port **3464**, your CMDLINE (command line) attribute would look like the following (and shown in the figure below):

```
-c "installcomp ram,rim -mgrip rcs001 -mgrport 3464"
```



- 5 Click **OK**, and then click **Yes** to confirm your changes.
- 6 If you would like to alter the values of the number of times the agent will attempt to install the agent upgrade as well as the time interval between attempts, change the values of the **ATTEMPTS** and **INTERVAL** attributes in the same manner. The remaining attributes should retain their default values for best performance.
- 7 Connect the OS specific sample service (e.g., `HP_51x_RADIA_CLIENT_UPGRADE`) to the appropriate users/machines for distribution. See the *Application Manager and Application Self-service Manager Guide for Linux* for more information.

- ▶ Before connecting an upgrade service to any agent instances, make sure any users and machines that are configured to receive the agent upgrade service are not connected to self-maintenance for version prior to the 5.1x agents. If self-maintenance is not disconnected, and the upgrade is installed and completed, you could run the risk of back-leveling the upgraded agent by the application of agent maintenance for earlier versions of the agents on the agent's next connect to the Configuration Server.

To save disk space after a successful upgrade, remove the ZSERVICE.UPGRADE instance to un-install the install media deployed at \$IDMSYS/ClientUpgrade.

- ▶ The ZSTOP expressions in the sample Upgrade Application Service (ZSERVICE) instances as well as the supplied UPGRADE instances are OS-specific, these expressions are there to ensure the appropriate agent upgrade materials are distributed to agents that match the OS identifier contained in the expression.

Customizing the Agent Upgrade Services

Should you decide to create your own agent upgrade Application Service (ZSERVICE instance) to meet your enterprise needs, HP strongly recommends that you do so by copying one of the supplied sample instances to a new name, ensuring that the instance prefix (used to denote the target operating system) that is used as the source of the copy, matches your intended target operating system. This is to ensure that your new instance has the necessary ZSTOP expression pertinent to the operating system to which the upgrade applies.

If you decide to customize UPGRADE instance data, to meet your enterprise needs, HP strongly recommends that you do so by copying one of the supplied sample instances to a new name ensuring that the instance prefix (used to denote the target operating system) that is used as the source of the copy matches your intended target operating system. This is to ensure that your new instance has the necessary ZSTOP expression pertinent to the operating system to which the upgrade applies. By performing this copy instance function within the UPGRADE Class your custom UPGRADE instance should already have a connection to the appropriate SOFTWARE.PACKAGE instance for your intended target operating system. You should then, connect your custom UPGRADE instance to your custom ZSERVICE instance replacing the pre-existing UPGRADE instance connection in the Application Service instance.

Always be certain your UNIX agent upgrade service has connections like that depicted in [Figure 1](#) on page 8 of this document.

Adding the UPGRADE Class to the Master Connection Rules

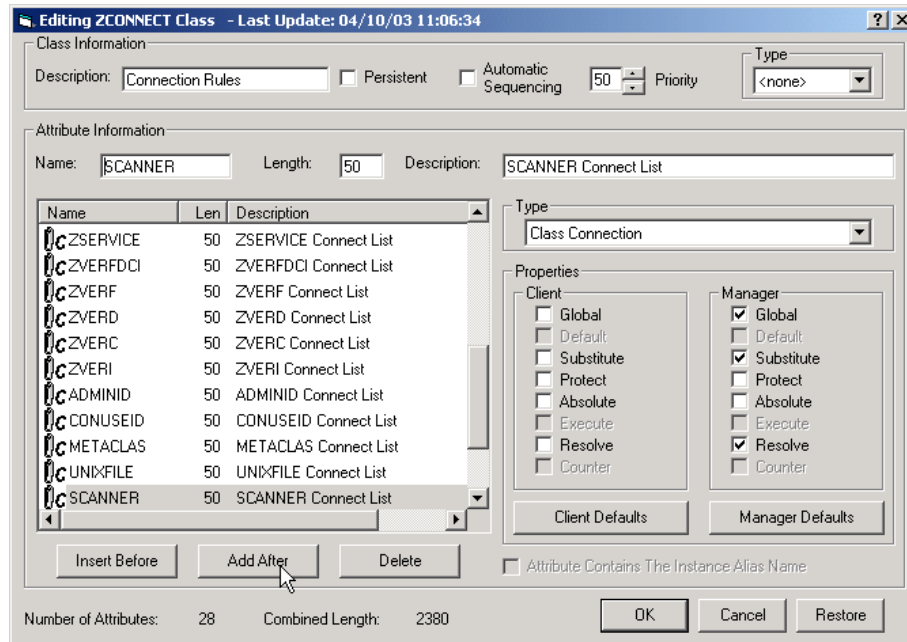
Should it be necessary to add new ZSERVICE/UPGRADE instances for the agent upgrade to suit your specific enterprise needs. You may use the HPCA Admin CSDB Editor to enable the connections lists for the new UPGRADE Class prior to adding the new instances.

Follow the steps below to have the UPGRADE class show up as a connectable class from the ZSERVICE Class, and the PACKAGE Class to be shown as a connectable class from the UPGRADE Class through the Show Connections feature of the HPCA Admin CSDB Editor.

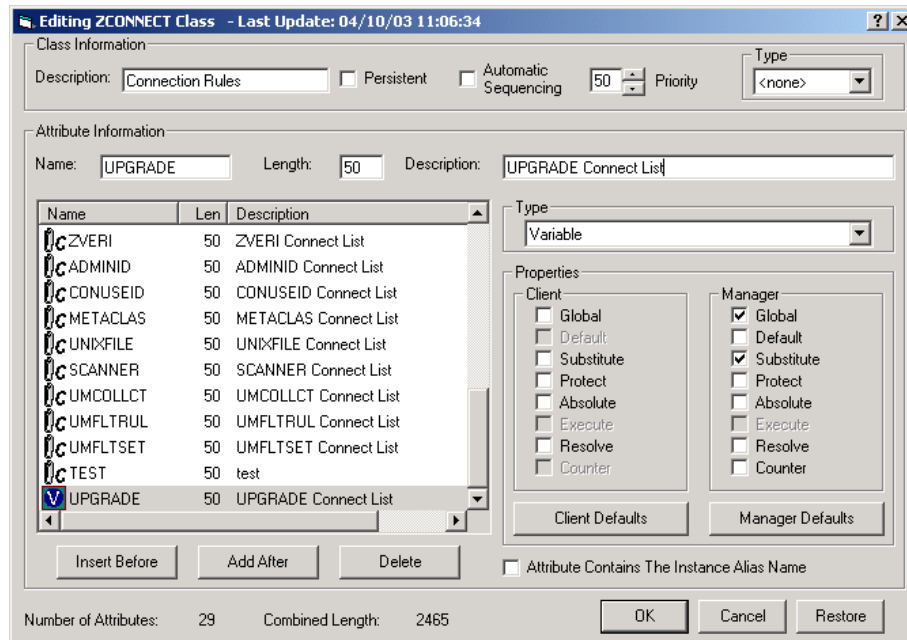
To add the Upgrade class to the Master Connection Rules

- 1 Use the HPCA Admin CSDB Editor and go to the PRIMARY.ADMIN.ZCONNECT Class.
- 2 Right-click **Connection Rules (ZCONNECT)** and select **Edit class**.

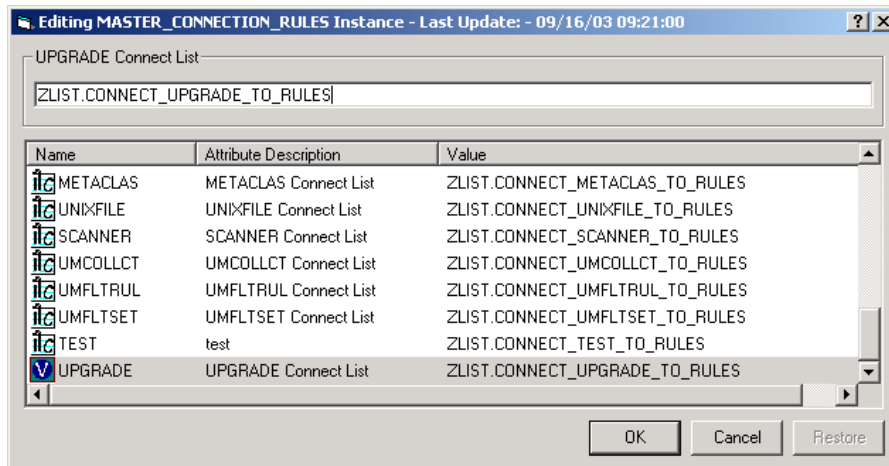
- In the Editing Class window, scroll to the bottom of the class list, highlight the last template entry and click **Add After**.



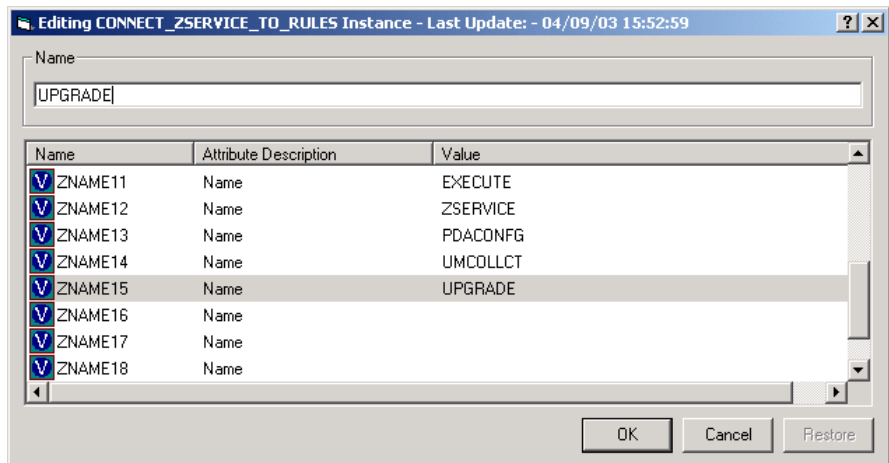
- Add a new class connection field similar to the existing connection fields.
- Name the new connection field **UPGRADE** and assign it a length of 50 and a description of **UPGRADE Connect List**.



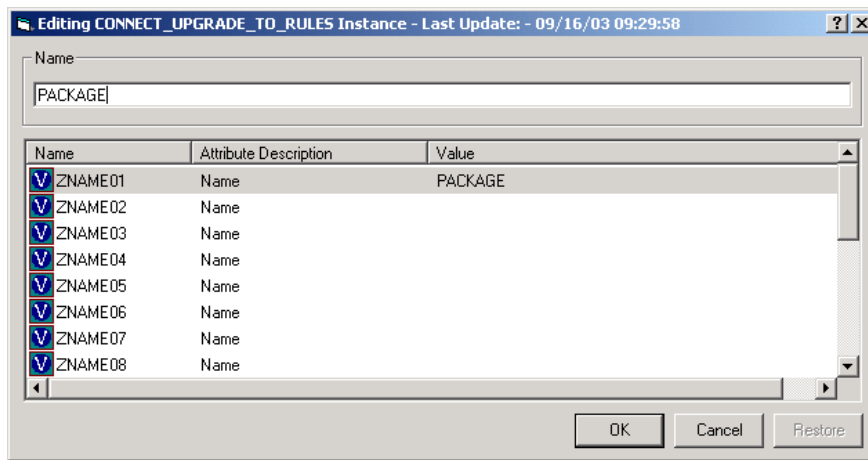
- Click **OK** then **Yes**.
- Right-click the **PRIMARY.ADMIN.ZCONNECT.MASTER_CONNECTION_RULES** instance and select **Edit Instance**.
- For the new **UPGRADE** attribute, add the value **ZLIST.CONNECT_UPGRADE_TO_RULES**.



- 9 Right-click the **PRIMARY.ADMIN.ZLIST.CONNECT_ZSERVICE_TO_RULES** instance and select **Edit Instance**.
- 10 Add the value **UPGRADE** in the first empty variable slot.



- 11 Click **OK**.
- 12 To add a new instance name to the **ZLIST** Class, right-click **Name Lists (8) (ZLIST)** and select **New Instance**.
- 13 Name the new instance **CONNECT_UPGRADE_TO_RULES**.
- 14 Right-click the new instance and select **Edit Instance**.
- 15 Add the value **PACKAGE** to the first empty variable entry.



16 Click **OK**.

Completion of these steps shall enable the HPCA Admin CSDB Editor to display the UPGRADE Class as a connectable class from the ZSERVICE Class and the PACKAGE Class to be shown as a connectable class from the UPGRADE Class.

Mechanics of the agent Upgrade for UNIX when applied as an HPCA Service

After a user or machine's policy entitlement model is configured to include the agent upgrade materials, and the agent connects and receives one of the mandatory application services (for example, HPCA_AGENT_72_UPGRADE_LINUX) containing the agent upgrade materials, HPCA will download all of the necessary materials to perform a agent upgrade.

The default location of the upgrade media, when downloaded via a agent connect, is `&(ZMASTER.ZSYSDIR)/ClientUpgrade` (the location, as defined in UPGRADE.MEDIADIR attribute) in the machine's entitlement model. The installation of the Upgrade Service runs a method called "install_upgrade" (with additional command line parameters) which waits for agent processes to finish before upgrading the agent binaries. The install_upgrade method will exit with a return code of 1, which results in a 709 radconct return code. This is the correct behavior as this tells the agent that the upgrade is not yet complete. For reporting purposes, an APPEVENT is sent that properly identifies the ZSERVICE with an ERRORNUM of 324 and STATUS of "Install/Upgrade is not complete." Depending on the level of radconct, install_upgrade may have to do a second connect to verify the upgrade status. Install_upgrade and upgrade logs can be found at `$IDMLOG` folder for troubleshooting purposes

When the agent processes have finished their execution, the actual agent upgrade begins. The upgrade process stops the agent daemons (radsched and radexecd) if they are found to be running, upgrades the agent components specified by the `-installcomp` parameter of the UPGRADE Class instance or the SelectComponent section of the specified configuration file, and restarts the agent daemons that were running at the time the upgrade was initiated.

To save disk space after a successful upgrade, remove the ZSERVICE.UPGRADE instance to un-install the install media deployed at `$IDMSYS/ClientUpgrade`.

Troubleshooting the agent Upgrade for UNIX

The agent upgrade creates an `upgrade.log` file in your log directory (`IDMLOG`). This file contains information regarding the upgrade and whether or not a successful upgrade has

taken place. Review this file when the upgrade is complete, if your upgrade is not successful and more information is required, you can append `-loglevel 9` to the upgrade command line. This will set maximum logging for the agent installer. Then run the agent upgrade again.

The agent installer log file `setup.log` produced by the increased log level can be found in the `~/tmp/setup` directory.

Should you encounter a problem when running the agent upgrade, provide be prepared to provide the files `upgrade.log` as well as `setup.log` files to HP Technical Support.

Upgrading the agent to Version 7.20 – from DVDROM

To upgrade the Agent – Local upgrade method

- 1 Login to the Linux workstation as root.
- 2 From the HPCA DVD, go to `/Agents/platform` (where *platform* is the operating system on which you are performing the agent upgrade).
- 3 Type:

```
./upgrade -c "--installcomp <PARMS> -mgrip<IP> -mgrport <PORT>"
```

where:

- The text *PARMS* is either a single argument or a comma separated list of agent components that you wish to upgrade.

Valid agent components include:

- ram - Application Manager
- rim - Inventory Manager
- rsm - Application Self-service Manager
- svr - Server Management
- patch - Patch Manager
- rom - OS Manager

- The text *IP* would be replaced with the literal IP address of the Configuration Server from which you would like your upgraded agent to receive new applications.
- The text *PORT* would be replaced with the port number on which the Configuration Server is running.

The `upgrade` command line is used to specify the command line arguments to be passed to the agent upgrade process. Because all arguments specified after the `-c` option are passed directly to the agent installer, any arguments that can be specified on the agent install command line can be specified here except for the `-mode` argument which is automatically set by the upgrade script to perform a silent install/upgrade.



The command line arguments following the `-c` option must be placed inside double quotes.

The `-installcomp` command line option is used to specify the specific products to be upgraded or installed on the agent workstation. For example, to upgrade your Application Manager and also install the Inventory Manager and Application Self-service Manager, you would specify `"-installcomp ram,rim,rsm"`.

- ▶ When multiple products are specified in the command line arguments the product acronyms must be comma separated.

The `-mgrip` and `-mgrport` command line options specify the Configuration Server IP address and port that the upgraded agent will use for its ZMASTER object and sample shell scripts. Should you upgrade or install the Application Self-service Manager, its configuration file called `args.xml` would be populated with Configuration Server IP address and port number specified with the `-mgrip` and `-mgrport` command line options. The default value for `-mgrport` is 3464.

You can specify `-loglevel 9` in the command line to set maximum logging for the agent upgrade/install.

Alternatively, you can perform a local agent upgrade using an agent install configuration file.

Should you have an installation configuration file already on the agent workstation, (created by the Remote Installation Setup process of the agent installer), you can have the agent upgrade use this configuration file as a basis to perform an upgrade.

To perform a local agent upgrade using an agent installation configuration file

- 1 Create a configuration file by running the agent installation. See any agent guide for instructions on running a remote installation setup.
- 2 Login to the Linux workstation as root.
- 3 From the HPCA DVD, go to `/Agents/platform` (where *platform* denotes the operating system on which you are performing the agent upgrade).
- 4 Run the following:

```
./upgrade -c "--cfg configfilename.cfg"
```

Where *configfilename.cfg* specifies a fully qualified path name to an install configuration file that contains the components selection and desired configuration options.

For information about installer configuration files (created by the Remote Installation Setup process of the agent installer), please consult the agent installation chapter of the Application Manager, Inventory Manager, and/or the Application Self-service Manager Guides for Linux.

- ▶ We do not recommend that you specify both a configuration file and command line options `-installcomp`, `-mgrport`, or `-mgrip` because the command line options will override the values specified in the configuration file.

For example, the command line of `-c " -loglevel 9 -cfg /tmp/myown.cfg"` would rely exclusively on the variables set in the `/tmp/myown.cfg` configuration file to determine what HPCA components to upgrade/install and what Configuration Server IP and port to use as well as other upgrade/install options.

Index

A

- Agent Upgrade
 - configuring, 11
 - mechanics, 16
 - Methods, 6
 - troubleshooting, 16
- Agent Upgrade Services
 - customizing, 13
- Always Connection, 9
- ATTEMPTS, 9

C

- CMDLINE, 10
- Command Line, 10
- Create Method, 9

I

- importing
 - classes, 6
 - instances, 6
 - resources, 6
- installcomp, 10
- INTERVAL, 9

M

- Master Connection Rules, 13
- Media Directory, 9
- MEDIADIR, 9

- mgrip, 10
- mgrport, 10

P

- passport registration, 4

R

- RedHat, 6
- Retry Attributes, 9

S

- sample services, 8
- SuSE, 6

U

- Update Method, 9
- UPGRADE Class, 8
- UPGRADE.XPC, 6

V

- Verify Method, 9

Z

- ZCREATE, 9
- zedmams.log, 7
- ZSTOP expressions, 8
- ZUPDATE, 9
- ZVERIFY, 9