HP OpenView Multicast Server Using Radia

for the HP-UX, Solaris, and Windows operating systems

Software Version: 3.1

Installation and Configuration Guide

Document Release Date: December 2005

Legal Notices

Warranty

Hewlett-Packard makes no warranty of any kind with regard to this document, including, but not limited to, the implied warranties of merchantability and fitness for a particular purpose. Hewlett-Packard shall not be held liable for errors contained herein or direct, indirect, special, incidental or consequential damages in connection with the furnishing, performance, or use of this material.

A copy of the specific warranty terms applicable to your Hewlett-Packard product can be obtained from your local Sales and Service Office.

Restricted Rights Legend

Use, duplication, or disclosure by the U.S. Government is subject to restrictions as set forth in subparagraph (c)(1)(ii) of the Rights in Technical Data and Computer Software clause in DFARS 252.227-7013.

Hewlett-Packard Company
United States of America

Rights for non-DOD U.S. Government Departments and Agencies are as set forth in FAR 52.227-19(c)(1,2).

Copyright Notices

Trademark Notices

Linux is a registered trademark of Linus Torvalds.

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

OpenLDAP is a registered trademark of the OpenLDAP Foundation.

Acknowledgements

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

2

Support

Visit the HP OpenView support web site at:

http://www.hp.com/managementsoftware/support

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

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

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

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

To find more information about access levels, go to:

http://www.hp.com/managementsoftware/access_level

To register for an HP Passport ID, go to:

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

Revisions

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

- Version number, which indicates the software version.
- Print date, which changes each time the document is updated.

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

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

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

The following table indicates the changes that were made to this document since the previous release.

Table 1   Changes in This Document

Chapter	Version	Changes
All	3.1	Removed superfluous graphics and any references to them.
Chapter 1	3.1	Page 14, HP OpenView Proxy Server Using Radia (Proxy Server) is a new section.
Chapter 1	3.1	Page 15, revised section title from Summary to Documentation Map, and added Table 1 which provides a quick-view reference of each chapter.
Chapter 1	3.1	Page 17, the Supporting Documentation list was revised to include the Proxy Server Guide.
Chapter 2	3.1	Page 22, in accordance with corporate style policy, added "Windows" to each Windows operating system that is listed.
Chapter 3	3.1	Page 35, revised the text of the Note.

Contents

1 Introduction to Multicasting

Introduction

Unicasting vs. Multicasting

The standard method of data transmission is **unicasting** (see Figure 1 on page 12). In a unicast scenario, a server communicates with multiple clients individually and at different times, and transmits data to each during its communications session.

Figure 1   Unicasting scenario
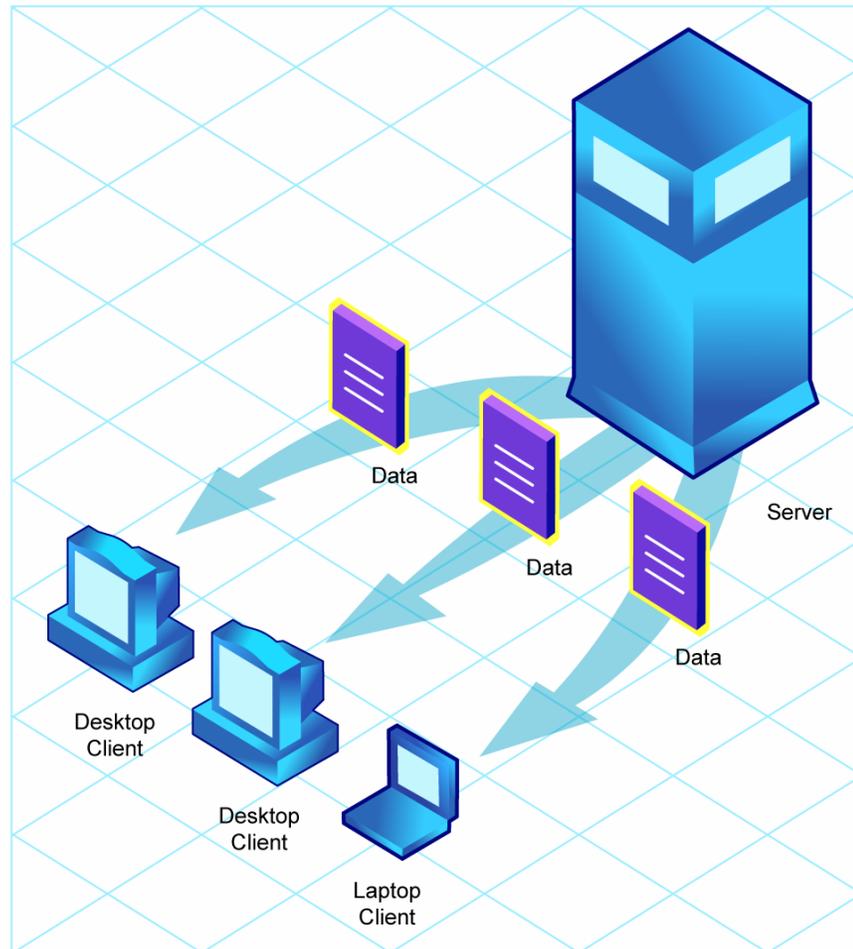
Since the server must repeatedly transmit identical data, the unicast method is very time-consuming.

Multicasting is a technique that allows the simultaneous transmission of a data stream to many receivers (see Figure 2 on page 13). The receivers identify themselves as "interested parties" by joining a logical group, using the Internet Group Membership Protocol (**IGMP**).

Figure 2   Multicasting scenario

The figure shows a Server (large blue tower), a Data packet, and three clients labeled Desktop, Desktop, and Laptop connected via a network.
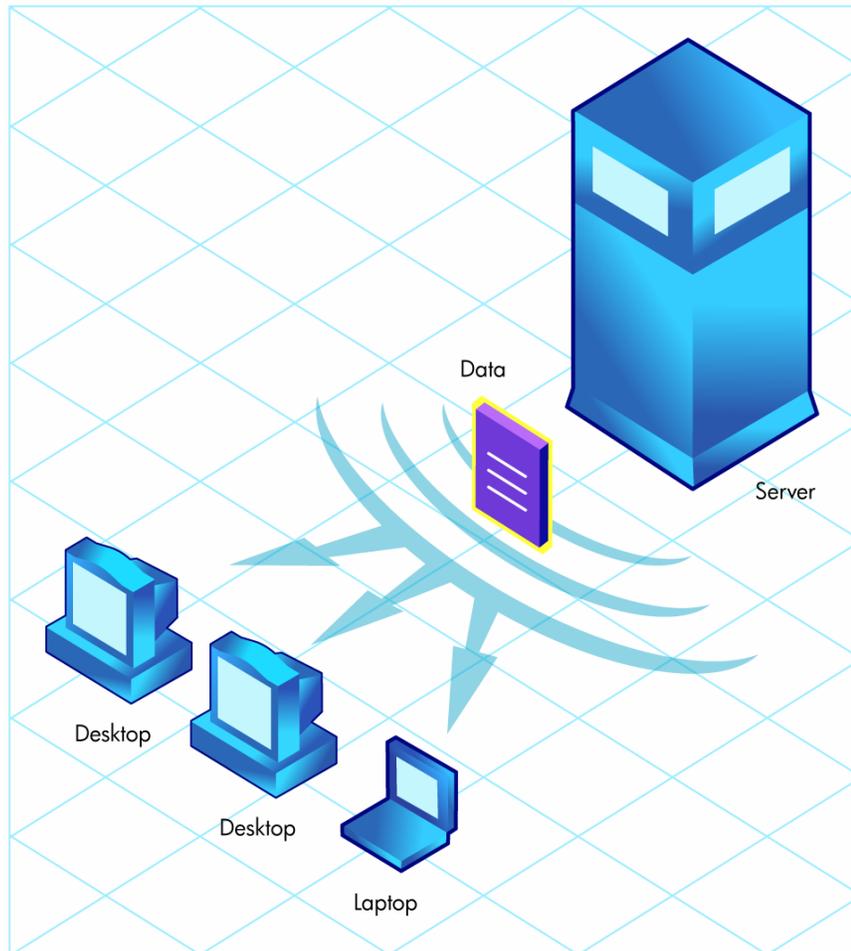
Benefits of Multicasting

Multicasting provides the following benefits:

- It maximizes the use of network bandwidth by transmitting a single data-stream to multiple clients, simultaneously.

- It saves resources on the server by not having to set up separate client sessions, and then repetitively transmit data individually to each client.

Multicast Server and other HP OpenView Using Radia Components

HP OpenView Inventory Manager Using Radia (Inventory Manager)

The Multicast Server accumulates session performance statistics that are sent to the HP OpenView Configuration Server Using Radia (Configuration Server) and HP OpenView Inventory Manager Using Radia (Inventory Manager). For more information on mining this data, see Appendix B, Multicast Results for Reporting, in this guide, and the *Installation and Configuration Guide for the HP OpenView Inventory Manager Using Radia* (*Inventory Manager Guide*).

During the installation, the IP address and port of the Inventory Manager are requested. Be sure to have this information before beginning the Multicast Server installation.

If the Inventory Manager has not been installed, see the section, Multicast Statistics in the Absence of an Inventory Manager, on page 63, for information on configuring database connections that will enable access to these results.

HP OpenView Operating System Manager Using Radia (OS Manager)

> The OS Manager will work with the Multicast Server in a Windows environment only.
>
> The information in this section is not applicable to other operating-system environments.

For more information on using these components together, see the section Multicast and the HP OpenView Operating System Manager Using Radia, starting on page 52 of this guide, and the *Installation and Configuration Guide for the HP OpenView Operating System Manager Using Radia* (*OS Manager Guide*).

HP OpenView Proxy Server Using Radia (Proxy Server)

The Multicast Server's dynamic-windows feature (see page 45) can be used to pre-load a Proxy Server. For more information, including an example, refer to the *Installation and Configuration Guide for the HP OpenView Proxy Server Using Radia* (*Proxy Server Guide*).

Audience

This guide is for systems administrators of HP OpenView Using Radia who want to use the HP OpenView Multicast Server Using Radia (Multicast Server) in their enterprise environments.

Documentation Map

This guide describes the Multicast Server; the following table provides an overview of this book, which will aid in locating specific information about the Multicast Server.

Table 1   Document Map

Chapter	Contents
Chapter 1 Introduction to Multicasting	The theory behind multicasting, and a comparison of the HP OpenView Using Radia multicasting facility to traditional data transmission.
Chapter 2 The Multicast Server	The implementation and benefits of the HP OpenView Using Radia multicasting facility, including a description of the four primary phases of the HP OpenView Using Radia multicast process.
Chapter 3 Installing the Multicast Server	Instructions on installing the Multicast Server.
Chapter 4 Configuring a Multicast Server Environment	Information on the MULTCAST class, the Radia database class in which a multicast session is configured; multicast's dynamic-windows feature; and advanced configuration options.
Chapter 5 Using the Multicast Server	The three methods by which the Multicast Server can be tested.
Appendix A Time Zone Adjustments	Instructions on calculating time-zone differences, which is necessary for configuring a multicast session.

Chapter	Contents
Appendix B Multicast Results for Reporting	Instructions on how to use the Multicast Server's post-session performance statistics for reporting. These statistics can be used with the Inventory Manager and Configuration Server.
Network Test Modules	Instructions on how the Multicast Server's test modules can assist in tuning the multicast and broadcast parameters for a network configuration.
Appendix D Implementation and Diagnostics	This appendix contains information that should be reviewed prior to starting a multicast session (Preliminary Parameter Calculation, Pre-Multicast Session Checklist, and Multicast Program-Call Schematic), as well as diagnostic (Querying the Various Logs) and troubleshooting information (Troubleshooting Tips).

Terminology

The following table lists the terms that might be used interchangeably in this book, as well as in other HP OpenView Using Radia publications.

Substitution is dependent on the context and, therefore, is not always possible.

Table 2   Terminology

Term	Alternate
Application	software, service
Client	Radia client, Application Manager, Software Manager
Computer	workstation, server, machine, host, desktop
edmprof file	Configuration Server Settings File; Profile File; Profile Editor; `edmprof.dat` (Windows); `.edmprof` (UNIX) **Note**: This is the text file wherein a Configuration Server's operational parameters are specified. This manual uses this non-platform specific, generic reference.

Term	Alternate
NOVADIGM domain	PRDMAINT domain **Note**: Starting with the 4.0 release of the Radia database, the NOVADIGM domain was renamed to PRDMAINT. Therefore, references to the PRDMAINT domain can be assumed to be referencing the NOVADIGM domain in pre-version 4.0 Radia databases.
Configuration Server	Manager, Active Component Server

Supporting Documentation

The following documents are referenced in this manual. They are shipped as part of the standard HP OpenView Using Radia library, and can be accessed at the HP OpenView web site. We recommend having these documents accessible.

- *User Guide for the HP OpenView Configuration Server Using Radia* (*Configuration Server Guide*)

- *Installation and Configuration Guide for the HP OpenView Application Manager Using Radia* (*Application Manager Guide*)

- *System Explorer Guide for the HP OpenView Administrator Workstation Using Radia* (*System Explorer Guide*)

- *Installation and Configuration Guide for the HP OpenView Inventory Manager Using Radia* (*Inventory Manager Guide*)

- *Installation and Configuration Guide for the HP OpenView Operating System Manager Using Radia* (*OS Manager Guide*)

- *Installation and Configuration Guide for the HP OpenView Proxy Server Using Radia* (*Proxy Server Guide*)

2 The Multicast Server

Most multicast utilities are designed to provide the simplest delivery of a payload that has been statically composed of all files and components for all possible recipients. In this model, every receiver is forced to take all resources bundled into that payload.

The HP multicast option allows the collection of the sets of resources that are needed by only those receivers that are eligible to participate in a specific multicast transmission. This means that only data required by the participants is sent, and the participants retrieve only the data that they have requested.

Benefits of HP OpenView Using Radia Multicasting

Multicast technology is seamlessly integrated with HP OpenView Using Radia products to provide a delivery mechanism that builds on the existing resource-optimization capabilities.

By removing the requirement to repetitively transmit data (to each receiver individually), HP OpenView Using Radia provides an additional dimension to its existing architectural focus—using minimum resources in order to bring the desktop computing environment to its "desired state."

Additionally, an administrator has the ability to perform a "centralized" configuration of the clients, and the ability to group together clients that have similar needs.

Flexible, Centralized Configuration

In many multicast implementations, the logistics of configuring receivers in order to synchronize them with data transmitters becomes very labor-intensive or, for a large number of receivers, extremely restrictive. The administrator must ensure that the receivers are listening at the time the multicast transmission takes place and that the appropriate multicast address has been configured for all listeners. Altering these parameters on a large number of machines, and orchestrating the synchronization of sender and receivers on short notice, can be extremely difficult, if not impossible.

The configuration that is required to setup HP OpenView Using Radia multicast distribution is performed centrally by the administrator. The clients obtain all required multicast configuration parameters in the framework of the standard Radia client connect. The parameters can be changed regularly or as needed, and the client connect process can be started using any of the standard options that allow end user-scheduled, or remotely invoked, initiation.

To further optimize its multicast capabilities, HP OpenView Using Radia provides the ability (using the standard administration tools provided with the product) to associate users with similar requirements into separate multicast groups (for example: file servers, desktops, and HP OpenView Staging Servers Using Radia [Staging Servers]). A scenario in which this "grouping" would be beneficial is Staging Servers that need to be populated overnight with large volumes of data can be segregated from desktops that need smaller application updates as soon as possible.

Increasing Network Efficiency

HP has designed its multicasting utility to enable you to maximize the functionality of your existing network bandwidth. This is accomplished by configuring the Multicast Server to transmit a single data-stream to multiple clients, concurrently.

Each client can distinguish between files and components delivered in the multicast stream, and retrieve only those that it needs. As a result, transfer time is minimized, and the client's storage and processing resources are conserved, because they are not forced to process and store superfluous data.

Guaranteed Delivery

The Multicast Server uses existing HP OpenView Using Radia functionality to determine whether all the required resources have been delivered in the multicast phase. It then proceeds to deliver any unsatisfied requirements using the standard protocols, the client connect process. This protects against potential "thrashing" that can typically occur in broadcast/multicast type protocols when the quality of the communication channel is poor. The Multicast Server can be tuned to the network characteristics using a buffering/transmit scheme, and a programmable time delay between packets.

Results Reporting

Since the Multicast Server is an extension of the HP OpenView Using Radia product family, results of a multicast session are reported to the Configuration Server and Inventory Manager databases. This information can then be mined for reporting and auditing purposes. For more information on how this is accomplished, and the benefits of this feature, see Appendix B, Multicast Results for Reporting.

Operational Requirements

It is important to note that the Multicast Server will work only with services (applications) that are specified as mandatory. For more information on how to specify/ensure that services are mandatory, refer to the *Installation and Configuration Guide for the HP OpenView Application Manager Using Radia* (*Application Manager Guide*).

HP OpenView Using Radia uses the standard IGMP. The Multicast Server requires:

- The availability of HTTP, and

- Routers to be enabled for the IGMP protocol.

> ▶ There are network configurations (primarily satellite or single sub-area networks) that might not support IGMP, so a broadcast option is offered as an alternative.

Additionally, the following requirements must be established in an environment in order to ensure the proper execution of multicasting.

- The Multicast Server must have network connectivity to the Configuration Server.

- The Configuration Server must be, at a minimum, at version 4.5.1.

> ▶ Although the Multicast Server can be installed on a machine other than that which houses the Configuration Server, it is strongly recommended that they be co-resident.

- HP OpenView Using Radia clients must be, at a minimum, version 4.0.1.

- The Proxy Server must be at version 2.1. For more information, refer to the *Proxy Server Guide*.

- The Multicast Server will run on the following platforms:
  - **Windows**: Windows NT 4.0, Windows 2000, and Windows 2003
  - **UNIX**: HP-UX and Solaris

The Multicast Process

During the multicast process, the client interacts with the Configuration Server and the Multicast Server. The multicast process is defined by two time-windows: the **preparation window** and the **distribution window**. (See Figure 3 on page 23.) The time windows are then sub-divided into four **phases**, during which the Radia client interacts with the Configuration Server and the Multicast Server in order to:

- Determine which files are needed.
- Collect the requirements for the necessary files.
- Transmit the files.
- Report the results of the transmission.

Multicast Phases

The multicast phases involve interaction (at various times) between the client, the Configuration Server, and the Multicast Server. The client is involved in all four phases of the process, whereas the Multicast Server and the Configuration Server are active only during those phases in which their participation is necessary. This section details these phases, as well as each component's role in the process.

The four phases of the multicast process are:

- Configuration Phase
- Collection Phase
- Multicast Distribution Phase
- Clean up-and-Reporting Phase

> The parameters of these phases are configured in multicast instances in the Radia database.

Figure 3   HP OpenView Using Radia Multicast time windows and phases
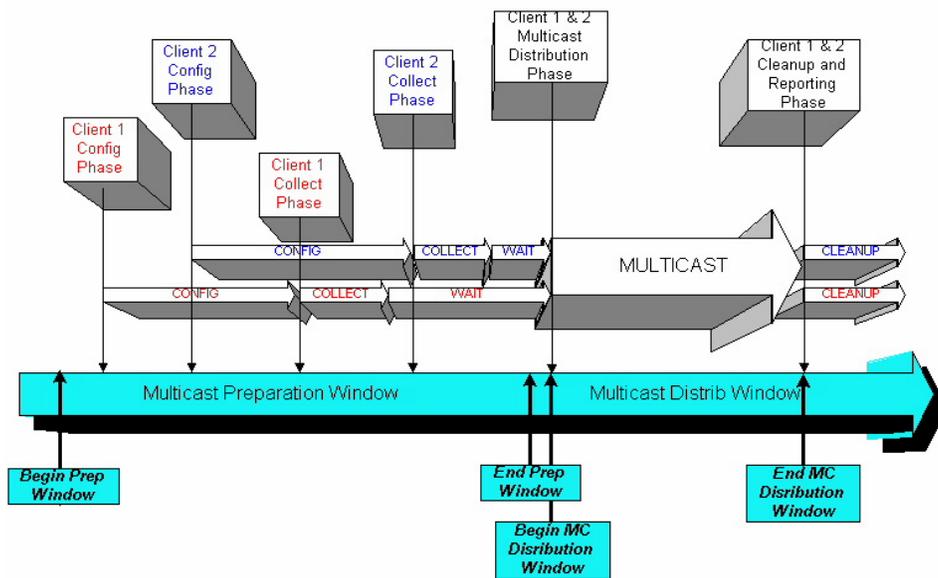
Configuration Phase

The configuration phase (shown in Figure 4 below and Figure 5 on page 25) occurs during a standard client connect with the Configuration Server. A client goes through a normal resolution to determine the resources it requires, and whether it is attached to a multicast group, as established by the system administrator.

Figure 4   Configuration Phase (first part)
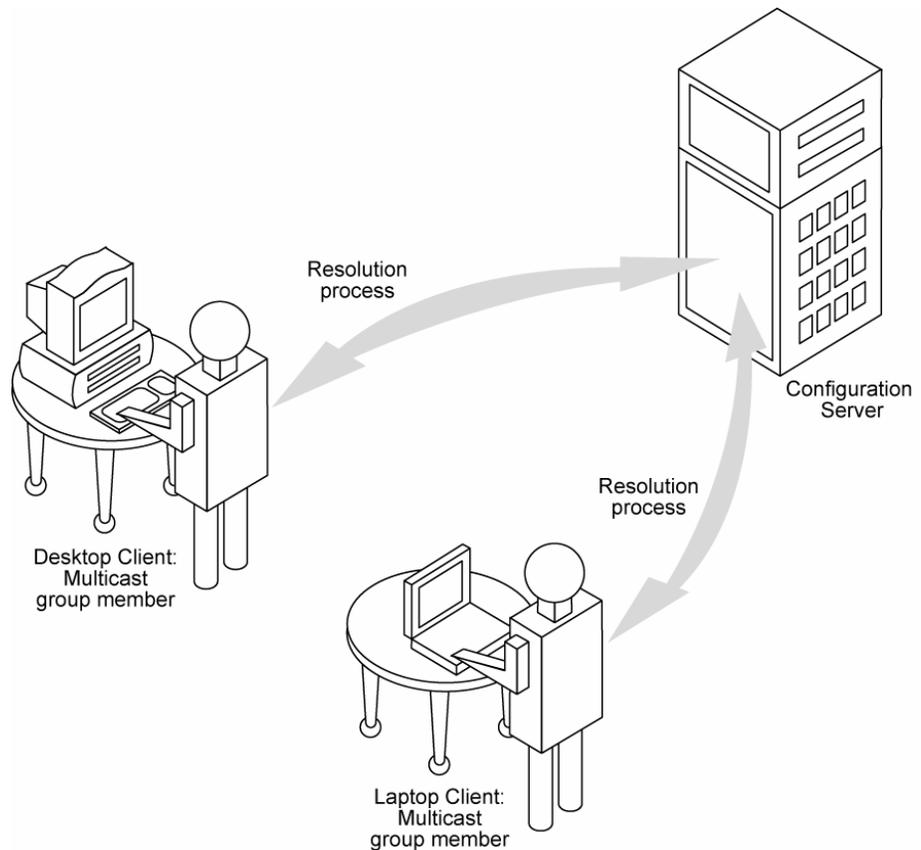
If a client is eligible, it then connects to the Multicast Server and exchanges the MULTCAST object, which contains the multicast parameters. If the

connection fails in this phase, the client will perform normal, point-to-point retrieval (with the Configuration Server, using TCP/IP without multicast) to obtain all resources required to attain the "desired state."

Thus far, all communications have been performed using HP standard TCP/IP communications.

Figure 5   The Configuration Phase (second part)
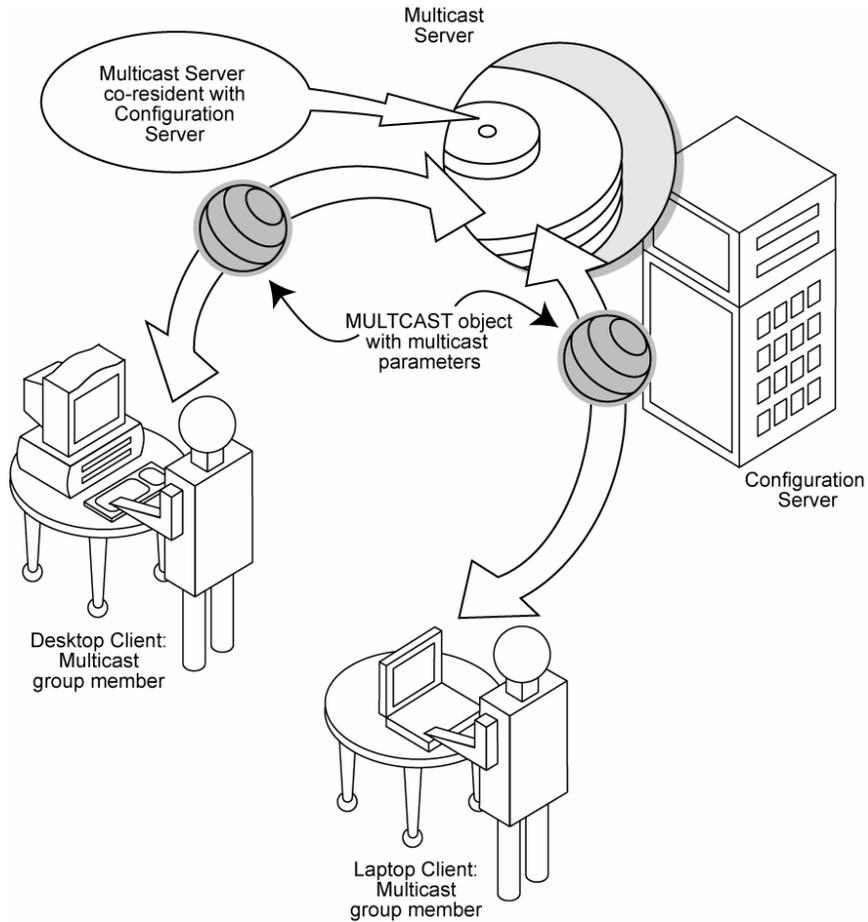
Collection Phase

In this phase (assuming the criteria to be eligible for multicast distribution, as described in the first phase, have been met), the client will send, to the Multicast Server, the MMCLIST object (the list of requested resources that was compiled by the Configuration Server during the Configuration phase). The Multicast Server stockpiles all client requests and prepares to transmit the files at the scheduled time. The client adopts a wait mode, in anticipation of the multicast transmission (Figure 6).

Figure 6   The Collection Phase
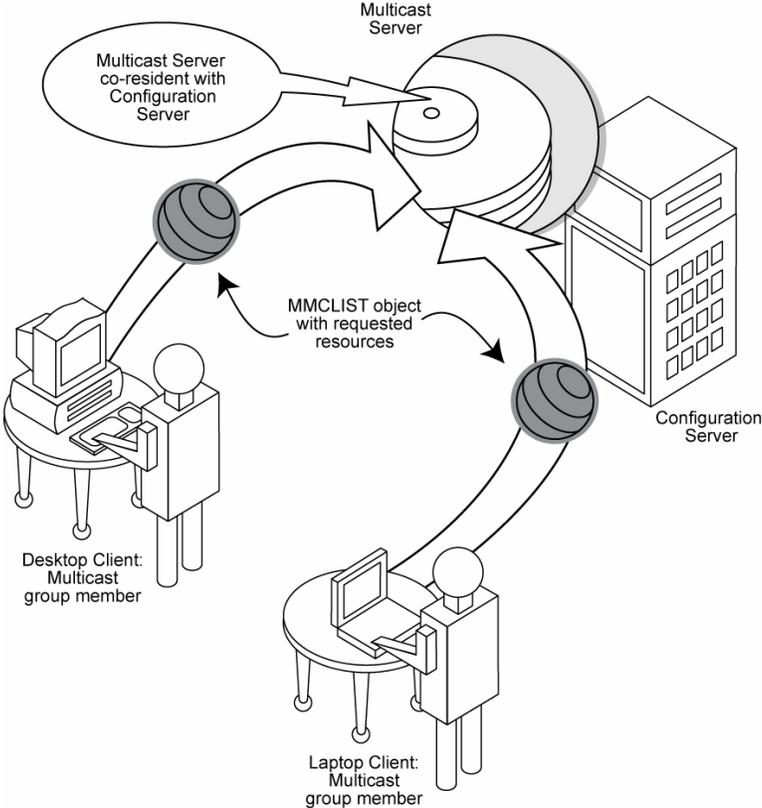
Distribution Phase

The third phase is the actual multicast transmission of the files (see Figure 7 below). The Multicast Server retrieves required files from the Configuration Server, and then transmits a single data-stream that contains only the files and components that have been defined by the clients involved. As files arrive, they are checked (in the MMCLIST object) by the client to determine if they are required. If required, it is read from the multicast stream and marked to indicate it has been received.

Figure 7   The Distribution Phase
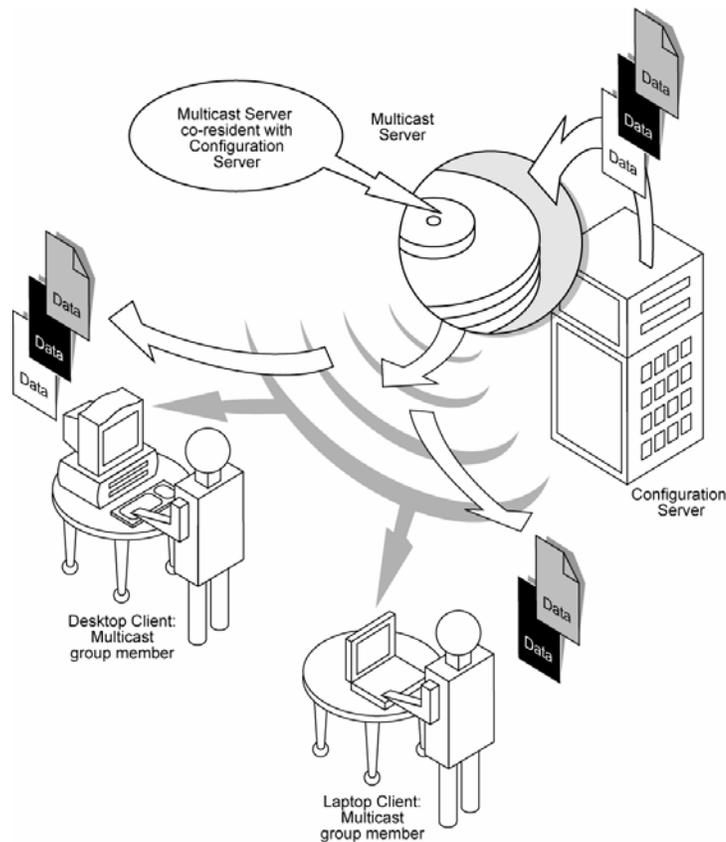
Clean up-and-Reporting Phase

The final phase of the process runs when all required files have been received by the client, or the distribution time window has expired. At this time, the client checks for the presence of all required resources, and uses standard HP OpenView Using Radia communications techniques to obtain, from the Radia database, any that are missing. Installation activities are then run for the resources; the completion status (including error information and multicast file transfer statistics) is reported to the Radia database (Figure 8).

Figure 8   The Clean up-and-Reporting Phase
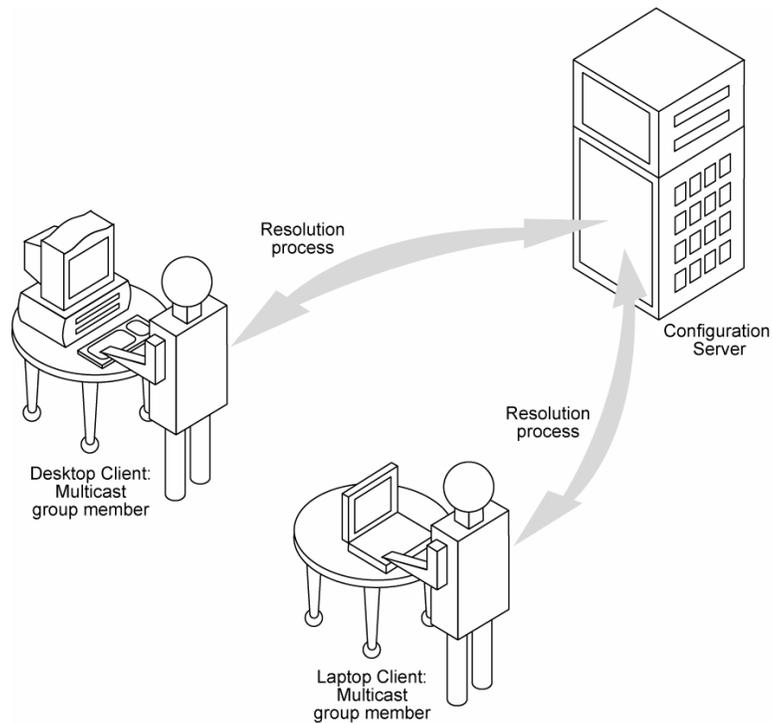
Completion Status Information

Unlike most multicast implementations, which are fundamentally file-transfer utilities, the Multicast Server provides full status information on the completion and installation of the distribution for each connecting client.

See Appendix B, Multicast Results for Reporting for more information on how multicast session results can be used for auditing and reporting.

3 Installing the Multicast Server

Multicasting Components

The Multicast Server is implemented in the standard HP OpenView Using Radia product architecture, which contains, at a minimum, a Configuration Server, and multiple clients.

> Radia database interaction in the multicast process is two-fold—its role in the client connect, and as the repository of the database files. There are no Configuration Server changes required in order to enable multicasting.

The Radia Database

The Radia database is the repository from which the Multicast Server will obtain the files that will be transmitted to the clients during the multicast.

> Although the Multicast Server does not have to be on a machine that houses a Configuration Server, HP strongly recommends it.

Additionally, the Radia database contains the information required to:

- Determine whether a client is eligible for multicast.
- Determine whether a required service is eligible for multicast.
- Assign the client to a multicast group.
- Inform the client of the files it needs in order to achieve its desired state.

> All installation and testing of HP OpenView Using Radia multicasting should be performed on a **copy of the production Radia database**, prior to introducing it to a production environment.

Multicast Server Directories

Table 3 lists the directories of the Multicast Server, and the files contained therein.

Table 3   Multicast Server Directories and Contents

Directory	Contents
`Multicast_Server`	Platform-specific folders housing the Multicast Server code and execution engine.
`Multicast_Test_Modules`	Two sub-directories, `multicast_test_modules\receive` and `multicast_test_modules\send`. For information on these test modules, see Appendix C, Network Test Modules, which describes the test programs for broadcast and multicast file-transfers. The `receive` directory contains: • the broadcast and multicast receive files (`BRECV.CMD` and `MRECV.CMD`) • the receive executable (`RADCRECV.EXE`) • two application `.dll` files The `send` directory contains: • the broadcast and multicast receive files (`BSEND.CMD` and `MSEND.CMD`) • the send executable (`RADCSEND.EXE`) • four test files of varying sizes

Installation

> Before beginning this installation consult the section, Operational Requirements (on page 21).

This section describes the Multicast Server installation. Although this exercise is performed in a Windows environment, the UNIX steps are similar, but with the expected platform differences. Additionally, there are pre-installation steps for a UNIX environment, which are described in the next section.

UNIX Pre-Installation Notes

- Make sure the user performing the installation has adequate UNIX operating system rights in order to create, and update, the target installation directory.

- Make sure the user performing the installation has a `home` directory on the UNIX workstation, and is not logged in as root.

- The logon user ID should be the same user ID as that which was used to install the Configuration Server. (This will ensure that the correct Configuration Server profile is queried in order to locate the Radia database.)

- To start the UNIX installation of the Multicast Server, type **./install**, on the command line. (Depending on how the UNIX operating system mounts the CD, it might require the user to specify the installation program (in uppercase) as **./INSTALL**.)

To install the Multicast Server in a UNIX environment

1 Open a UNIX shell window at the console, or through an X-Windows emulator, logged on as the UNIX user ID who will be running the HP OpenView Using Radia software.

2 Insert the CD-ROM into the CD-ROM drive.

3 Change the current directory to the directory on the CD-ROM drive containing the installation program (`setup.sh`), as in:

> **cd /**_cd-drive_**/Multicast_Server/unix-type**

For example, if the Multicast Server is being installed on an HP-UX platform, and the CD-ROM drive is mounted as `cdrom`, type,

```
cd /cdrom/Multicast_Server/hpux
```

4   Press **Enter**.

5   At the prompt, type,

```
./install
```

6   Press **Enter**.

The installation program will start. Continue with the steps outlined in the following section, Multicast Server Installation.

To install the Multicast Server in a Windows environment

1   Insert the CD-ROM into the CD-ROM drive.

2   Click **Start**, and **Run**.

The contents of the Multicast Server CD-ROM are displayed.

3   Double-click the Multicast_Server folder.

4   Double-click the Win32 folder.

5   Double-click setup.exe.

The installation program will start. Continue with the steps outlined in the next section, Multicast Server Installation.

Multicast Server Installation

The Radia Multicast Server Install Welcome window opens.

1   Click **Next**.

(At any time during the installation, click **Cancel** to exit the installation.)

The HP Software License Agreement window for Multicast Server opens.

2   Click **Accept**.

> If **Accept** is not selected, the Multicast Server installation program will terminate.

The Radia Multicast Server Files Location window opens.

This window displays the directory into which the Multicast Server files will be installed.

— Accept the default directory that is specified in this window. (Recommended)

— Or specify a different directory for the Multicast Server files.

3 Click **Next**.

The Radia Multicast Server Resource Location window opens.

In this window, select one of the locations in which the Multicast Server files will be stored.

— Select **Radia Configuration Server database** or,

— Select **Radia Proxy Server static cache**.

> For this exercise, **Radia Configuration Server database** was selected.
> If **Radia Proxy Server static cache** is selected, a similar series of windows will be seen.

4 Click **Next**.

The Radia Multicast Server RCS Resource Location window opens.
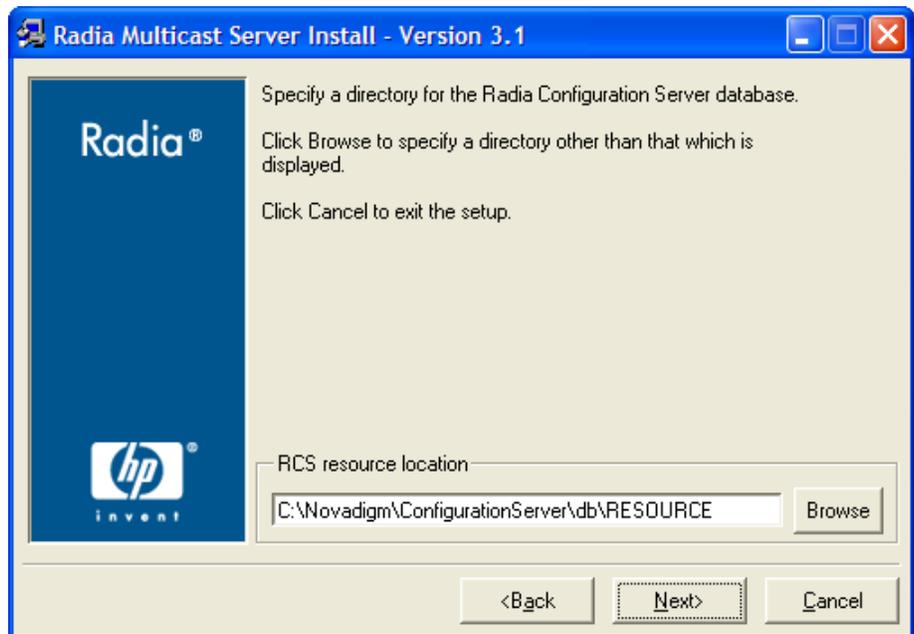
This window displays the location of the Configuration Server's database.

— Accept the default directory that is specified in this window. (*Recommended*)

— Or specify a different location.

5   Click **Next**.

The Radia Multicast Server Install Port window opens.

This window displays the communications port that the Multicast Server will use.

— Accept the default port (**3463**) that is specified in this window. (*Recommended*)

— Or specify a different port for Multicast Server communications.

6   Click **Next**.

The Radia Multicast Server Install Summary window opens.

This window presents all the information that was specified during the Multicast Server installation. This is the final opportunity to review and modify the specified settings.

— If you discover any errors, or wish to modify any of the entries, click **Back** until you reach the appropriate windows, and make the necessary changes.

The information that was entered in the other windows will not be affected. After making the changes, click **Next** repeatedly, until you arrive back at the Summary window.

7 To continue, click **Install**.

When the installation is complete, the Radia Multicast Server Install Finish window opens.

8 Click **Finish**.

The Multicast Server has been installed successfully.

Additionally, in a Windows environment, the Multicast Server should now be present as a service (**mcast**) in the **Services** list.

UNIX Post-Installation Note

Unlike the Windows installation, on UNIX systems the Multicast Server is not automatically started by the installation program. The start and stop shell scripts (`startmcast.sh` and `killmcast`) are provided in the `multicast` directory. These scripts can be used as-is, or tailored to the host system.

Windows Services

The Multicast Server is automatically installed as the service **mcast** on Windows platforms. Its startup configuration (**Automatic**, **Manual**, **Disable**) can be specified in the **Startup Type** area on these platforms.

For more information, consult the documentation specific to the operating system on which the Multicast Server has been installed.

The next chapter, Configuring a Multicast Server Environment, discusses the multicast-specific features of the Radia database and describes the changes that are required using the System Explorer. Chapter 5, Using the Multicast Server, starting on page 55, contains an example of how to set up a multicast session on a client machine.

4 Configuring a Multicast Server Environment

The Multicast Server is a processing engine that requires access to the resources in the Radia database. Once the Multicast Server has been installed, it is necessary to make a few modifications to the Radia database.

When the client connects to the Multicast Server, it sends up the **MULTCAST** object, from which the Multicast Server obtains control information for the multicast session. The client also sends up the list of required files in an object called **MMCLIST**. The Multicast Server collects the file lists, in groups, based on the multicast address.

The parameters needed for the multicast session are stored in the MULTCAST object (see the section, The MULTCAST Class, below) in the Radia database. They are delivered to the client during normal resolution, and are stored in the client's sub-directory for each service.

Configuration Server Changes

In order for multicasting to execute properly, a few modifications must be made to the Radia database. These changes are described in this section, starting with the **MULTCAST** class—the database class that controls the multicast session. Following that are the details of variables that deal with:

- multicast eligibility (**MCELIGBL**),
- the delivery and installation of mandatory services (**MCORDER**),
- the delivery of multicast-specific client objects (**BYPASCON**), and
- the duration of a multicast session (**MWINDOW**).

The MULTCAST Class

The MULTCAST class of the POLICY domain is where a Radia multicasting session is configured and scheduled. In order to perform these tasks, access the System Explorer by clicking **Start**, **Programs**, **Radia Administrator**, and **Radia**

System Explorer. The MULTCAST class should not be modified during a multicast session.

> Before starting the editing process, do the following.
>
> On the System Explorer tool bar, click **View**, **List View**, and select **Details**.
>
> On the System Explorer tool bar, click **View**, and select **Options**. Then:
>
> 1 On the **General** tab, select the check box for **Show Class Names Next to Descriptions**.
>
> 2 On the **Instance Options** tab, under **When Displaying Instance Attributes**, **Show Attribute**, select **Both**.

Figure 11 shows a list of the instance attributes of the _BASE_INSTANCE_ of the MULTCAST class.

Figure 9   **MULTCAST class _BASE_INSTANCE_ instance attributes**

Multicast Class "_BASE_INSTANCE_" Instance Attributes:		
Name	**Attribute Description**	**Value**
V DOMAIN	Domain Name	&(ZOBJDOMN)
V CLASS	Class Name	&(ZOBJCLAS)
V INSTANCE	Instance Name	&(ZOBJNAME)
V MCAST	Enable Multicast [Y/N]	Y
V MODE	Broadcast or Multicast [M/B]	M
V MCORDER	Service Installation Order [B/A/S]	S
V ADDRESS	Broadcast or Multicast Address	229.0.0.0
V PORT	Broadcast or Multicast UDP port	9512
V DELAYFP	Delay After First Packet (mil sec)	20
V DELAYBP	Delay Between Packets (mil sec)	5
V RESENDS	Number of Resends	1
V STORE	Backup Packets to Store for Resend	20
V CGMTDATE	Collection Start Date (YYYYMMDD)	
V CGMTTIME	Collection Start Time GMT(HH:MM:SS)	
V CWINDOW	Duration of Collection Window (min)	40
V MDELAY	Delay Before Multicast Start (min)	40
V MWINDOW	Duration of Multicast Window (min)	0
V TTL	Number of Router Hops	3
V BYPASCON	User to Prime Multicast Server	
V ALTADDRM	Multicast Server IP	
V ALTPORTM	Multicast Server Port	3463
V MINREF	Minimum Num Clients Requesting File	1
V MINSIZE	Minimum File Size for Multicast	1024
ALWAYS	Manager REXX Method	SYSTEM.ZMETHOD.MULTCAST

Table 4 contains the instance attributes (with a description and a sample value) of the MULTCAST class. Configure your environment-specific multicast scenario by specifying these attributes with the appropriate values.

Table 4   **MULTCAST Class Instance Attributes**

Attribute	Value	Description
DOMAIN	&(ZOBJDOMN)	Specifies the domain name.

Attribute	Value	Description
		Note: Do not change this value.
CLASS	&(ZOBJCLAS)	Specifies the class name. **Note**: Do not change this value.
INSTANCE	&(ZOBJNAME)	Specifies the instance name. **Note**: Do not change this value.
MCAST	Y or N	A flag to indicate if multicast is enabled. The default is **Y**.
MODE	B or M	Broadcast or Multicast. The default is **M**. **Notes**: If MODE = M, the ADDRESS variable should be specified in the standard Internet "dotted-decimal" format, and should be between 225.0.0.0 and 239.255.255.255. (Multicast addresses are defined as IP class-D addresses in this range. Avoid using the extreme low end of the range—224.0.0.1 – 224.255.255.255—as most of these are reserved for specific purposes.) The B option is for those network configurations (primarily satellite and single sub-area networks) that don't support IGMP.
MCORDER	B, A, or S	For information on this attribute, refer to the section, MCORDER, starting on page 47. The default is **S**.
ADDRESS	229.0.0.0	Specifies a valid broadcast/multicast address. Additionally, this attribute is important to the dynamic multicast feature. For more information, refer to the section, ADDRESS starting on page 45.
PORT	9512	Broadcast or Multicast UDP port.
DELAYFP	20	Delay (milliseconds) after the first packet is sent.
DELAYBP	5	Delay (milliseconds) between packets. For information on calculating this value, see Calculate DELAYBP and MWINDOW, on page 71.
RESENDS	1	Number of re-sends. **Note**: Multicast is designed to re-send all packets if RESENDS > 0. It will send *STORE* number of packets and then re-send that group for *RESENDS* times.
STORE	20	Number of packets to buffer for resends.

Attribute	Value	Description
		Note: Multicast is designed to re-send all packets if RESENDS > 0. It will send *STORE* number of packets and then re-send that group for *RESENDS* times.
CGMTDATE	19991213	The start date of the collection period. The format is **YYYYMMDD**.
		Additionally, this attribute is important to the dynamic multicast feature. For more information, refer to the section, CGMTDATE starting on page 46.
CGMTTIME	14:00:00	The start time of the collection period. The format is **HH:MM:SS**.
		Additionally, this attribute is important to the dynamic multicast feature. For more information, refer to the section, CGMTTIME starting on page 46.
		Note: This variable is specified in GMT. Refer to Time Zone Adjustments on page 57 for important time zone information.
CWINDOW	60	Duration (in minutes) of collection phase for all clients to register their list of required files.
		Additionally, this attribute is discussed as it relates to the dynamic multicast feature. For more information, refer to the section, Multicast Dynamic Windows starting on page 45.
		Note: When specifying the duration of your collection window, be sure it is sufficient to collect all the files in the request list.
MDELAY	1	Delay between close of collection and start of multicast (minutes).
MWINDOW	60	For information on this attribute, refer to the section, MWINDOW starting on page 48.
		Note: For information on calculating this value, see Calculate DELAYBP and MWINDOW, on page 71.
TTL	3	Number of router "hops."
BYPASCON	UserJoe	For information on this attribute, refer to the section, BYPASCON starting on page 48.
ALTADDRM	208.244.225.46	The IP address of Multicast Server for object exchange.

Attribute	Value	Description
ALTPORTM	3463	The port of Multicast Server for object exchange. **Notes**: This is the TCP/IP port that the Multicast Server listens on, not the multicast transmission port. The default is **3463**. This port was chosen to avoid conflicts with known HP OpenView Using Radia ports (such as the Configuration Server and Staging Server ports). Therefore, be prepared to adjust accordingly, as other network software might also conflict.
MINREF	1	The minimum number of clients that must request a file in order for it to be considered for multicast.
MINSIZE	1024	The minimum size a file must be in order to be considered for multicast.
ALWAYS	SYSTEM. ZMETHOD. MULTICAST	A Configuration Server REXX method.

Example

In the following example, the five primary multicast variables are specified, followed by the result of these specifications.

```
CGMTDATE  = 20011016
CGMTTIME  = 14:00:00
CWINDOW   = 45
MDELAY    = 2
ALTADDRM  = 208.244.225.46
```

Result

With the parameters above specified, a multicast session will be initiated on October 16, 2001 at 2 p.m. Greenwich Mean Time (**GMT**). The collection phase (**CWINDOW**) will last 45 minutes. When the end of the collection window has been reached, the Multicast Server stops the collection process.

There will then be a 2-minute delay (**MDELAY**) before the transmission begins. At the designated multicast start-time (*CGMTTIME* + *CWINDOW* + *MDELAY* = **14:47:00 GMT**), the Multicast Server (specified by **ALTADDRM**) starts multicasting the files that are on the list it compiled from the various eligible clients.

Multicast Dynamic Windows

To further increase the effectiveness of the Multicast Server, it offers the ability to configure dynamic (collection and transmission) windows. This means that for a logical distribution group, multiple multicast distributions can be active concurrently, allowing the delivery of data to members of the group falling into different time windows.

This is accomplished by specifying a range of valid IP addresses in the ADDRESS field and leaving blank either the CGMTDATE and CGMTTIME field.

Assume that:

- Most connections occur in a concentrated time-period (such as morning logons, between 8:00 and 10:00 A.M.) and

- There is a need to optimize the distribution by allocating collection windows for a relatively small duration (such as 30 minutes).

With the dynamic-windows feature, the overlap will be avoided because a different address will be used.

ADDRESS

A range of addresses can be specified in order to avoid multiple distributions overlapping on the same multicast address. The range must consist of an IP address, followed by dash, and a valid decimal value for the last octet (as shown in the following example). All addresses in the range must be valid within the IGMP class-D IP address range (225.0.0.0 – 239.255.255.255).

```
ADDRESS = 225.0.0.000-034
```

Note that the first three octets (**225.0.0**) are fixed and the range is only in the last octet (**000-034**). The addresses in this range will be used sequentially to accommodate overlapping windows for a group.

The Multicast Server will keep a list of addresses that are currently in use for each MULTCAST instance. When a new multicast session is created, the Multicast Server will look for a valid, available address (within the range specified in ADDRESS), assign it to the new session, and add the address and the associated session to the in-use list. When a session completes its transmission, the address is removed from the in-use list and is available once again. If all of the addresses in the range are in use, the client will be informed that no IGMP address is available.

Although the range is limited to the last octet of the IP address, this should not be a significant limitation because there are 255 possible entries. Even if the collection windows were as small as 15 minutes, for a full 24 hours, only 96 (4 * 24) addresses would be needed. The 255 possible entries allow more than two-and-a-half days of continuous collection windows.

CGMTDATE

If the CGMTDATE field is empty and CGMTTIME has a valid time, a multicast session will be run once a day. Its collection window will start at the same time (as specified by CGMTTIME) each day.

The duration specified in the CWINDOW field, as well as all other duration-type fields, will be used as described in Table 3 on page 41.

CGMTTIME

Conversely, if the CGMTTIME field is empty and the CGMTDATE field has a valid date, multiple multicast sessions can run throughout the date specified. In this situation, the collection window will begin when the first client connects and transmits the MULTCAST object to the Multicast Server. This collection window will remain open for the duration specified by CWINDOW, and multicast-eligible clients that connect subsequent to this will use this collection window until it closes.

The first client to connect after this window closes will cause the Multicast Server to create a new collection window, subject to the availability of a valid IGMP address different from that of the first session. This is determined when a range of addresses is specified in the ADDRESS field.

CGMTDATE and CGMTTIME Not Specified

If CGMTDATE and CGMTTIME are blank, the MULTCAST instance can be used continually, over any range of dates and times, subject to the availability of an IGMP address.

Time Zone Offsets

In order for a multicast session to execute when you want, time-zone offsets must be taken into consideration. Since the Multicast Server/Configuration Server and client might be in different time zones, all multicast times are specified in GMT. As an administrator, you must determine the GMT-offsets of your servers and clients.

▶ All HP OpenView Using Radia multicast software converts the local system clock to GMT prior to doing any comparisons or calculations.

For an overview of GMT and time zone calculations, refer to Appendix A, Time Zone Adjustments starting on page 57.

Advanced Multicast Configuration Options

Three multicast-specific variables (MCORDER, MWINDOW, and BYPASCON) are in the MULTCAST class of the Radia database. A fourth, MCELIGBL, is in the ZSERVICE class of the Radia database. This section describes these variables, and how they can be customized, using the Radia System Explorer.

MCELIGBL

In order to participate in a multicast session, services must be individually defined as multicast-eligible. This is done with the MCELIGBL variable, found in PRIMARY.SOFTWARE.ZSERVICE. All services that are instances of the ZSERVICE class, will, by default, be multicast-eligible, because the Configuration Server installation sets MCELIGBL=Y.

If a service is not to be distributed via multicast, edit only that service by specifying MCELIGBL=N.

▶ Do not edit the _BASE_INSTANCE_.

MCORDER

If a service is mandatory, and not eligible for multicast, it will be downloaded to and installed on the client directly from the Radia database. This variable

enables an administrator to specify when to download and install mandatory services that are not multicast eligible.

Services can be configured as mandatory with the "on/off switch," **ZSVCMO**, an instance attribute of all services. For more information on the deployment of mandatory services and ZSVCMO, refer to the *Application Manager Guide*.

The MCORDER attribute is located in the PRIMARY.POLICY.MULTCAST class. It has three values, as described in Table 5.

Table 5   MCORDER Values

Value	Result
B	Download the service and install it on the client BEFORE the multicast session.
A	Download the service and install it on the client AFTER the multicast session.
S	SPLIT the process so that it downloads the service to the client before the multicast, and installs the service after the multicast session. This is the default.

MWINDOW

This variable controls the amount of time (in minutes) for a client to run before terminating its multicast "receive" activity. The default is **0**, which allows a client to maintain the receive mode for as long as is necessary to receive all the requested data. This will not affect the duration of the multicast session; it will ensure only that the client process does not run longer than is necessary to receive the data it requested.

> For more information on calculating a value for MWINDOW, see Appendix D, Implementation and Diagnostics.

BYPASCON

When multicast processing occurs, each client connects to the Configuration Server and the Multicast Server. The clients then send a **control object** and the list of required resource files (a *needs list*) to the Multicast Server.

However, if all the clients' data payloads are identical, HP OpenView Using Radia multicasting can be configured to save on network bandwidth usage, by limiting which clients send which data to the Multicast Server. This is

accomplished with the BYPASCON variable (of the PRIMARY.POLICY.MULTCAST class). BYPASCON has three values.

BYPASCON=
All clients will exchange a control object and a needs list object with the Multicast Server.

BYPASCON=UserJoe
The client with this user ID will send both objects to the Multicast Server, and the other clients will send only the control object.

BYPASCON=UserJoe 2 (a user ID, followed by a space and the numeral 2)
The client with this user ID will send both objects to the Multicast Server; no other clients will be part of the object exchange.

▶ This option requires that the local time on the clock of the machine that houses the Multicast Server be in synchronization with all multicast-eligible clients.

Creating a Multicast Instance

Multicast instances must be created, and then connected to users, workgroups, and departments in order for them to be included in a multicast session. This section describes how to create a multicast instance. Associating Groups with a Multicast Instance on page 50 describes how to associate workgroup and department instances with a multicast instance.

To create a multicast instance

1   From the **Start** menu, select **Programs**, **Radia Administrator**, and **Radia System Explorer**.

2   In the tree-view, double-click the following icons to open them:

    PRIMARY, **POLICY**, and **MULTCAST**.

3   Right-click **MULTCAST** and select **New Instance** from the pop-up menu.

    The **Create Instance** dialog box opens.

4   Type an instance name (for example, **MCAST1**), and click **OK**.

    The MCAST1 instance is now displayed in the tree view and the list view of the System Explorer as an instance of the MULTCAST class.

5   Double-click **MCAST1** in the list view and verify that it has inherited the attributes of the _BASE_INSTANCE_.

A multicast instance has been successfully created.

> To create additional multicast instances, repeat steps 3 and 4.

By creating several instances in the MULTCAST class (for example MCAST1, MCAST2, etc.), multiple multicast sessions can easily be customized and scheduled for various client computers.

Associating Groups with a Multicast Instance

In order to be included in a multicast session, a client must be part of a group (such as a workgroup or a department) that is "associated" with a multicast instance. Once a user is connected to a group, it will automatically be included in any multicast session for which the group is configured. Since, in a typical scenario, there will be more than one client, and each client will likely be affiliated with more than one group, it is far more efficient to associate a multicast instance with a group.

Typically, users will be assigned to groups based on department or geography. Bandwidth capacity might be another consideration when assigning users and groups. Although a client can be associated with several multicast sessions, at any given time, only one session can be active for a client.

This section describes how to create this association using the System Explorer. For detailed instructions, refer to the *System Explorer Guide*.)

> Refer to the *Application Manager Guide* for comprehensive information on how to include users in workgroups and departments.

To associate a group with a multicast instance

1   Open the System Explorer, and navigate to and open PRIMARY.POLICY.WORKGRP.

2   Right-click on the workgroup that is to be associated with multicast, and from the pop-up menu that appears, select **Show Connections**.

3   From the drop-down list in the resulting dialog box, select **POLICY**, and then double-click, **MULTCAST**.

Now, MCAST1 should be in the list view of the System Explorer, and the PRIMARY.POLICY.WORKGRP class should be open in the tree view.

4 Left-click (and hold) the **MCAST1** icon.

5 Drag the **MCAST1** icon (still holding down the left button of the mouse) to the selected instance of PRIMARY.POLICY.WORKGRP.

As you drag the MCAST1 icon, a circle with a slash will appear. This will change to a paper-clip icon when you place it on the selected instance. This indicates that the connection is allowed.

6 Complete the connection by dropping (releasing the left button of the mouse) the MCAST1 icon on the instance.

The Select Connection Attribute dialog box opens.

7 Click **Copy** (MCAST1 instance to PRIMARY.POLICY.WORKGRP._BASE_INSTANCE_).

In the tree view, MCAST1 will be listed under the associated instance, and POLICY.MULTCAST.MCAST1 will be listed as an _ALWAYS_ connection in the list view.

The MCAST1 instance has been successfully connected to the WORKGRP class.

Multicast and the HP OpenView Operating System Manager Using Radia

The HP OpenView Operating System Manager Using Radia (OS Manager) supports guaranteed-delivery multicast so that large numbers of operating system images can be concurrently rolled out.

> ▶ This section explains how to configure the Multicast Server for use with the OS Manager.
>
> For information on the OS Manager, refer to the *Installation and Configuration Guide for the HP OpenView Operating System Manager Using Radia* (*OS Manager Guide*).

Requirements

In order for the Multicast Server to work with the OS Manager the requirements that are listed in this section must be met.

- The Multicast Server must be installed on a Windows machine.

- The Multicast Server must be version 3.1.

- The OS Manager must be version 2.0 or later (support of a Multicast-aware version of ROMA).

  > ▶ Although version 3.0 of the Multicast Server is compatible with version 3.1, only version 3.1 supports the OS Manager.

- The operating-system images will be downloaded only if the **Service Multicast Eligible** option is selected for the operating-system service.

  To do this:

  a Open the HP OpenView Management Portal Using Radia administrative interface and go to the appropriate operating-system service.

  b Click **Modify**.

  c Click **Advanced**.

  d Scroll to near the bottom of the screen and make sure that **Service Multicast Eligible** is selected.

Configuring Multicast for use with the OS Manager

The following steps detail how to configure the Multicast Server for use with the OS Manager.

To configure guaranteed-delivery multicast

1 Go to the appropriate Behavior instance.

2 Click **Advanced** to see the advanced options.

3 Modify the **ROMA Parameters** field as follows:

```
-multicast multicastIPAddress:3463 -mcastretrycount 1 -
mcastretrywait 240
```

The following table describes the ROMA parameters.

Table 6   ROMA Parameters Described

Parameter	Description
multicastIPAddress	Refers to the Multicast Server host; *host name* can be used also; 3463 is equal to the Multicast Server port.
mcastretrycount	Specifies the number of times that multicast will be retried if there is a failure. The default is **1**.
mcastretrywait	Specifies how long to wait before starting the retry. The default is **240** (seconds).

4 If necessary, modify the Multicast Server's configuration file, `mcast.cfg`, which is located in *SystemDrive*`:\Novadigm\MulticastServer\etc.`.

The following table describes the parameters of `mcast.cfg`.

Table 7   MCAST.cfg Parameters Described

Parameter	Description
root	The root directory from which the Multicast Server will retrieve resources.
address	A range of IP addresses for use with *dynamic windows*. See Multicast Dynamic Windows on page 45.

Parameter	Description
minref	The minimum number of clients that must request a file in order for it to be considered for multicast.
cwindow	The duration (in minutes) of the collection phase during which all clients must register their list of required files.

5   If changes were made to `mcast.cfg`, restart the Multicast Server service (**mcast**) to implement the changes.

5 Using the Multicast Server

Now that the Multicast Server has been installed and configured, it is important to test it before putting it into production to ensure that it will execute as expected. There are three ways to perform a multicast test: one from a client perspective, and two from an administrator perspective.

The client-based method is not efficient in production, because it is specific only for that client. However, it is a good test to check the installation and configuration of the Multicast Server. The two remaining methods are more suited to a production environment because they offer the ability to include multiple clients (as part of a group, as discussed in the previous chapter).

In order to conduct either of these tests, do the following.

- Delete the service that is going to be used for the test from the client machine.

- Connect the multicast instance to the client via a group.

- Set up a multicast instance with an active collection window.

Testing the Multicast Server

Client Testing

To test the operation of the Multicast Server and the associated client software

> Make sure that the service that is being used for the test has been deleted from the Application Manager client machine on which the test is being conducted.

1 On the client machine, open a command window.

2 Change the directory to the database location where the Radia client is stored.

3 Type:

```
Radskman.exe mname=serverid,dname=SOFTWARE
,ipaddr=serverid,port=3464,cat=Y
```

where *serverid* is the network name of the Configuration Server.

This method of multicasting files is not suitable in production, because these steps would have to be executed on each client machine. The methods that are covered in the next section are more appropriate for production because the Radia database can be accessed and then used to include clients in a group, which can then be associated with a multicast instance.

This means that at a central location, multiple clients can be configured for a multicast session. Additionally, the eligibility of a service (MCELIGBL), the parameters for delivering the service (MCORDER), and the transmission of the request files (BYPASCON) can be determined.

Administrator Testing

Notify and TIMER

In addition to the client test presented above, the Application Manager's **Notify** and **Scheduler** (**TIMER**) features are convenient ways to transmit data to Application Manager client computers.

> For comprehensive information on these features, refer to the *Application Manager Guide*.

To determine if the installation (multicast session) was successful, look at `connect.log` and `radrecv.log` in the client's `Novadigm/log` directory.

A Time Zone Adjustments

In order for a multicast session to execute when you want it to, the CGMTTIME instance of the MULTCAST class must be configured correctly. Since the Configuration Server uses the operating system's clock, it is important that the CGMTTIME instance be properly set—using Greenwich Mean Time (GMT). Additionally, **base-24** (a.k.a. military) time must be used when configuring this setting.

> GMT is a constant and does not adjust for Daylight Saving Time (**DST**).

Therefore, a Radia database in NY, USA, which is 5 hours (300 minutes) behind GMT during local *standard* time, would need the proper number of adjustment minutes added, in order to be synchronized with GMT. For various sample GMT settings, see the examples that follow.

Example A (DST in effect):

To schedule a multicast session to begin collection on Wednesday, July 09, 2001 at 2:35:15 (P.M.) local time, on a Multicast Server in New York, USA, specify:

    CGMTDATE=20010709

    CGMTTIME=18:35:15

> The value of the MULTCAST class instance attribute, CGMTTIME (2:35:15 PM), can be calculated as follows:
> - Convert the time to base-24 time (2:35:15 becomes 14:35:15).
> - Add to this, the difference between EDT and GMT (4 hours).
>
> The result is the GMT time (18:35:15) that must be specified for the CGMTTIME attribute in order to schedule the specified multicast session.

Example B (DST *not* in effect):

To schedule a multicast session to begin collection on Wednesday, July 09, 2001 at 2:35:15 (P.M.) local time, on a Multicast Server in New York, USA, specify:

```
CGMTDATE=20011109

CGMTTIME=19:35:15
```

➤ The value of the MULTCAST class instance attribute, CGMTTIME (2:35:15 PM), can be calculated as follows:
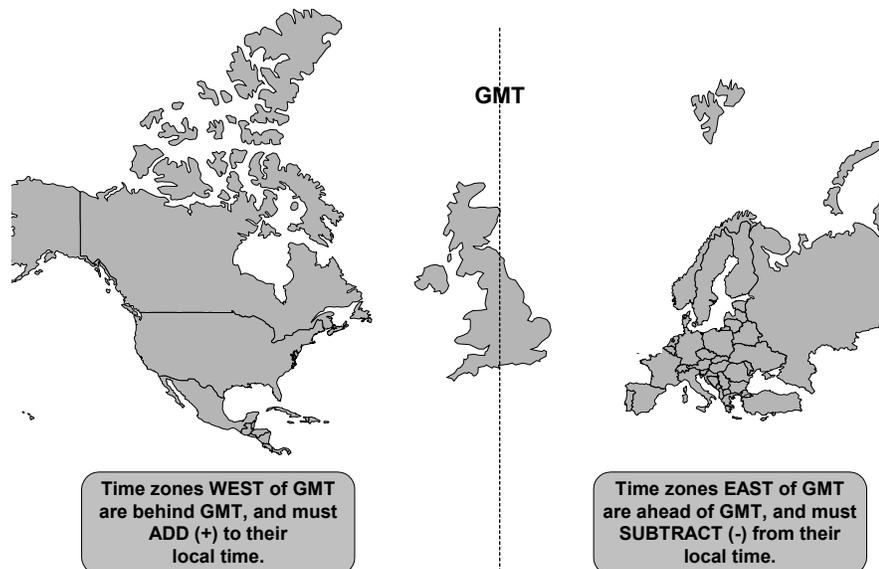
- Convert the time to base-24 time (2:35:15 becomes 14:35:15).
- Add to this, the difference between EDT and GMT (5 hours).

The result is the GMT time (18:35:15) that must be specified for the CGMTTIME attribute in order to schedule the specified multicast session.

Time Zone Overview

Error! Reference source not found. offers a map with a view of the approximate location of GMT, and will help you remember whether to adjust forward or back for various time zones.

Figure 10   The line in the center of the graphic represents Greenwich Mean Time (GMT)

GMT

Time zones WEST of GMT are behind GMT, and must ADD (+) to their local time.

Time zones EAST of GMT are ahead of GMT, and must SUBTRACT (-) from their local time.

Automatic Adjustments for Daylight Saving Time

If the machine that houses the Configuration Server offers the ability to have its clock automatically adjust for the Daylight Saving Time change, we recommend that this feature be activated.

- On a Windows machine, this is accomplished in the **Control Panel** area.

- On a UNIX machine, this is configured during installation. If you need further information, consult the operating system's documentation.

B   Multicast Results for Reporting

At the end of a multicast session, the Multicast Server accumulates session performance statistics and sends them, via http, to the Inventory Manager database—if one is installed. These statistics are sent in a Web-based Enterprise Management (**WbEM**) object, **RMSSTATS**, which contains information such as the:

- Number of files requested

- Number of files rejected

- Number of files and bytes transmitted

- Date and time the transmission started

The full list of session-performance statistics is detailed in Table 5, on page 62.

RMSSTATS Object

For more information on the RMSSTATS object in the Inventory Manager, see the section, RMSSTATS and Inventory Manager, starting on page 63.

> If there is no Inventory Manager in the environment, the database connections can be configured so that the multicast statistics can be examined. See the section, Multicast Statistics in the Absence of an Inventory Manager, on page 63.

Besides being sent to the Inventory Manager, the RMSSTATS object can be sent to the Configuration Server (in addition to or instead of). For more information, see the section, RMSSTATS and the Configuration Server, starting on page 64.

RMSSTATS

The following table contains a list of the variables of the RMSSTATS object.

Table 8   Multicast RMSSTATS Object Variables

Variable	Value
CLASS	NVD_MulticastStatistics
KEY	ServiceID,SourceType,SourceID
NAMSPACE	NVD
PROP000	ServiceID:S=*manager.domain.class.instance*
PROP001	Nclients:I=*number of clients connected*
PROP002	Ndevices:I=*number of devices connected*
PROP003	SourceType:S=MCS
PROP004	SourceID:S=*IPaddr:port.MCinstance* • *IPaddr:port* are the address and port that the client uses to connect to the Multicast Server. • *MCinstance* (the multicast instance) distinguishes between multiple sessions on the same machine.
PROP005	NfilesReq:I=*number of files requested*
PROP006	NbytesReq:I=*number of bytes requested*
PROP007	NfilesXmt:I=*number of files transmitted*
PROP008	NbytesXmt:I=*number of bytes transmitted*
PROP009	NfilesRej:I=*number of files rejected* **Note**: The number of files rejected for Nclients < MINREF.
PROP010	NbytesRej:I=*number of bytes rejected* **Note**: The number of bytes rejected for Nclients < MINREF.
PROP011	StartTime:T=*YYYY-MM-DDThh:mm:ssTZD* **Note**: The start time of the multicast transmission.
PROP012	Duration:I=*duration of download* (seconds)
STATUS	N/A
TYPE	WbEM
ZOBJDATE	*YYYYMMDD*
ZOBJTIME	*hh:mm:ss*

Variable	Value
ZOBJNAME	*service_name*
ZUSERID	*Hostname*

Multicast Durations

For the Multicast Server, the reported duration of the multicast session might be greater than that of the same session as reported by the client. Since the Multicast Server transmits files in the order of *most_files_requested* to *least_files_requested*, a client needing only the last file will record a duration that is much longer than the actual file-transmission time. It is likely that 90% of the time a client will discard packets until its requested files are transmitted. Therefore, multicast durations need to be properly computed before being specified.

For more information on how this information can be used for auditing and reporting, refer to the *Inventory Manager Guide*.

RMSSTATS and Inventory Manager

In the Inventory Manager database, the session-performance information is placed into a table (**rNVD_MulticastStatistics**) from which it can be mined (via the HP OpenView Management Portal Using Radia [Management Portal]) for reporting and auditing with SQL92-compliant reporting applications such as Microsoft Access and an SQL database table.

> See the next section, Multicast Statistics in the Absence of an Inventory Manager, for information on configuring database connections in order to enable access to multicast statistics when no Inventory Manager is installed.

Multicast Statistics in the Absence of an Inventory Manager

If there is no Inventory Manager installed in the environment, the multicast statistics can still be mined, and viewed with the System Explorer. To enable this feature, configure the database connections shown in the following table.

Table 9   **Database Connections for Multicast Statistics**

Connect...	To...
PROCESS.RMSSTATS	ZMETHOD.RMSSTATS
PROCESS.CLISTATS	ZMETHOD.CLISTATS

RMSSTATS and the Configuration Server

In addition to the Inventory Manager, the Configuration Server can be used to view the reporting statistics of a multicast session. In order to use it, an appropriate `NVDCAST.RC` file must be present and configured on the Multicast Server's host machine.

> The `NVDCAST.RC` file is for reporting purposes only; it is not required for standard multicast operations.

C Network Test Modules

Multicast Server media contains a `multicast_test_modules` directory that has two sub-directories, `receive` and `send`. These modules are provided in order to assist in tuning the multicast and broadcast parameters for a network configuration.

The `receive` directory contains:

- The broadcast and multicast receive files (`BRECV.CMD` and `MRECV.CMD`),
- The receive executable (`RADCRECV.EXE`), and
- Two application `.dll` files.

The `send` directory contains:

- The broadcast and multicast send files (`BSEND.CMD` and `MSEND.CMD`),
- The send executable (`RADCSEND.EXE`), and
- Four test files of varying sizes.

> ▶ The parameters that are specified in earlier sections of this document might not be suited to all network configurations. Specify values in accordance with the needs of the environment.

RADCSEND and RADCRECV

Two programs (RADCSEND and RADCRECV) perform broadcast and multicast file-transfers, based on command-line parameters. On Windows, these can be run from the command line using the `.cmd` files that are provided.

1 Copy the contents of `multicast_test_modules\receive` to a temporary directory on the destination computer.

2 Run `mrecv.cmd`.

3 Copy the contents of `multicast_test_modules\send` to the source computer.

4 Run `msend.cmd`.

TESTDATA0001 – TESTDATA0003 will be sent to the destination computer via multicast.

RADCRECV will create the log and data files in the current directory.

> If HP OpenView Using Radia is not installed, logs are created in the respective, current directories of each program. Otherwise, look for the radcrecv.log in the log sub-directory.
>
> If either IDMSYS or IDMLOG (in win.ini or nvd.ini) is defined, it will make use of directories for the data and log files.

Syntax

The syntax for the RADCSEND and RADCRECV programs is presented below. Following the syntax, Table 7 defines the parameters.

RADCRECV

```
RADCRECV M|B [address] port buffers timeout (min) [Object with
file names]
```

RADCSEND

```
RADCSEND M|B address port filename file number delay after
first packet delay between packets time-to-live buffers resends
last-file-flag[1] packet data size [n-to-drop[0]]
```

Table 10   Syntax Parameters Defined

Parameter	Definition
M|B	M = Multicast, B = Broadcast
address	The target network IP address (*xxx.xxx.xxx.xxx*). **Note**: This parameter must be omitted for RADCRECV in Broadcast mode, and always used for RADCSEND.
port	The IP port that is associated with the (broadcast or multicast) transmission.
filename	The name of the file to be sent.
file number	The relative file number (if more than 1 file is to be sent, increment this by 1 for each successive file). RADCSEND expects to find these files in the directory from which it is executed.

Parameter	Definition
delay after first packet	The delay (in milliseconds) after the first packet before sending the remaining packet. The delay gives the receiver time to open and compare the file.
delay between packets	The delay (in milliseconds) between data packets (packet size approximately 1 KB).
time-to-live	The IP TTL value (maximum number of routers to traverse).
buffers	The number of packets to buffer when packet-retransmits (resends) are specified.
resends	The number of retransmits for each grouping of *buffers* packets.
last-file-flag	Indicates the last file to be transmitted. This will cause the receiver to terminate execution at the end of this file. The default is **1** (true). Set to zero for all but the last file, if multiple files are sent.
packet size	The maximum number of data bytes transmitted per packet.
n-to-drop	Set this to a non-zero value to simulate packets being "dropped" by the transmission network. RADCSEND will randomly drop packets so that (100 / *n-to-drop*) percent of the packets will not be sent, and therefore, not received by RADCRECV. This value is optional and the default is **0**.
Object with file names	Multicast only. This value is optional. RADCRECV will store files in IDMDATA as named in this object. If not specified, the received file is stored in the same directory as RADCRECV. The object must have the same format as MMCLIST. RADCRECV also looks in IDMLIB for the object, MULTCAST, which is also optional. To set the parameters log level (default = **40**) and resends (default = **0**), ZTRACEL and RESENDS are set here.
timeout (min)	Maximum time that RADCRECV will accept data before stopping (failsafe mechanism for live run).

> If MULTCAST.ZTRACEL is set to 55 or greater, RADCRECV will log header information for every packet that gets to the GetControlInfo routine. (RADCRECV bypasses packets that are "malformed" or not intended for the multicast/broadcast session it is looking for.)

BROADCAST Tests

The contents of the BSEND.CMD file are:

```
radcsend B 231.1.222.6 9511 TESTDATA000n 1 50 50 2 20 1 1 1020
```

where:

Table 11   RADCSEND Parameters for BROADCAST

B	Indicates Broadcast.
231.1.222.6	*address* (specified on the command line)
9511	*port*
TESTDATA000*n*	*filename* (specified on the command line)
1 (2, 3, ...)	*file number*
50	*delay after first packet*
50	*delay between packets*
2	*time-to-live*
20	*buffers*
1	*resends*
1	*last-file-flag*
1020	*packet data size*
	n-to-drop (optional, not used in the above example)

To run, type:

BSEND *dest_IP_address filename*

> The receive program must be started first, in order to receive the file.

The contents of the BRECV.CMD file are:

```
radcrecv B 9511 10 45
```

where:

Table 12   RADCRECV Parameters for BROADCAST

B	Indicates Broadcast.
9511	*port*
10	*buffers*
45	*timeout* (minutes)

To run, type:

BRECV

MULTICAST Tests

The contents of the MSEND.CMD file are:

```
radcsend M 231.1.222.6 9511 TESTDATA0001 1 50 50 2 20 1 0 1020
radcsend M 231.1.222.6 9511 TESTDATA0002 2 50 50 2 20 1 0 1020
radcsend M 231.1.222.6 9511 TESTDATA0003 3 50 50 2 20 1 1 1020
```

where:

Table 13   RADCSEND Parameters for MULTICAST

M	Indicates Multicast.
231.1.222.6	*address* (specified on the command line)
9511	*port*
TESTDATA000*n*	*filename* (specified on the command line)
1 (2, 3, ...)	*file number*
50	*delay after first packet*
50	*delay between packets*
2	*time-to-live*
20	*buffers*
1	*resends*
1	*last-file-flag*

1020	*packet data size*
	n-to-drop (optional, not used in the above example)

To run, type:

MSEND

▶ The receive program must be started first, in order to receive the file.

The contents of the MRECV.CMD file are:

```
radcrecv M 231.1.222.6 9511 20 45
```

where:

Table 14   RADCRECV Parameters for MULTICAST

M	Indicates Multicast.
231.1.222.6	*address*
9511	*port*
20	*buffers*
45	*timeout* (minutes)
	Object with file names (optional, not used in the above example)

To run, type:

MRECV

D Implementation and Diagnostics

This appendix offers information on:

- Implementing the multicast process
  - — Preliminary Parameter Calculation, below
  - — Pre-Multicast Session Checklist, starting on page 74
  - — Multicast Program-Call Schematic, on page 75
- Diagnosing the multicast process to assist in problem resolution
  - — Querying the Various Logs, starting on page 77
  - — Troubleshooting Tips, starting on page 79

Preliminary Parameter Calculation

Before attempting to run a multicast session, it is necessary to:

1 Calculate the DELAYBP and MWINDOW values.

2 Test the network (using the multicast test modules, RADCSEND and RADCRECV) to ensure that it is properly configured for multicast.

3 Examine the test logs.

Calculate DELAYBP and MWINDOW

DELAYBP

`DELAYBP = (8*`*P*`/`*N*`)`

- **P** is the *packet size* (typically, **1020** bytes) and
- **N** is the *network bandwidth* (in this example, **16** kilobits/second).

  Therefore, **8** multiplied by **1020** (P) divided by **16384** (16*1024) equals **0.498** seconds, which rounds to **0.5** second.

  So, **DELAYBP = .5 second**.

`MWINDOW =` D **`* (`**T**`/`**P**`) * (`**R**` + 1)`**

- **D** is the DELAYBP (as computed above),
- **T** is the *total bytes* (to be transmitted),
- **P** is *packet size* (from DELAYBP example), and
- **R** is the *number of resends*.

  Assuming values of: R = **1** and T = **10MB** (10,000,000 bytes), the value of MWINDOW computes as follows: **0.5** (D) multiplied by **10,000,000** divided by **1020** (T/P) multiplied by **2** (R + 1) equals **9803.9215** seconds, which rounds to **9804** seconds.

  So, **MWINDOW = 2 hours, 44 minutes**.

Test the Network

Run the network test modules as discussed in Appendix C, Network Test Modules but substitute the value of DELAYBP with that which pertains to your network.

▶ Considering the MWINDOW value previously calculated, it is recommended that if testing on a 16-Kbps network, a smaller test file (such as 10 KB) be used. This should take approximately 10 seconds.

▶ MWINDOW is only a fail-safe to ensure termination of the receive program (RADCRECV). Use it in production—not for testing, where you are observing the test. Typically, it is used as an estimate of the end of the transmission. In practice, add time for a margin of safety.

Examine the Test Logs

▶ In the following examples, (*italicized text within parentheses*) is an explanatory comment; it will not be present in the log.

RADCRECV

The `radcrecv.log` can be found in the same directory as the program. If the Radia client is on the same machine, the log will be found in the `log` sub-directory of the Radia client installation directory, `IDMSYS`. At the end of the log is the summary information (shown below with irrelevant text removed):

```
08:50:57   Number of times Each Packet is Sent [1]
```
(*RESENDS + 1, no bearing on receiver*)
```
08:50:57   Multicast Packet Inactivity Timeout [5]
```
(*minutes*)
```
08:50:57   Multicast IP address [231.1.222.6] mode [M]
08:50:57   Port [9511]
08:50:57   Number of buffers: [20]
08:50:57   MWINDOW value in (sec): [2700]

09:30:20   Packets received: 39216 dropped(est): 0
```
(*The number of unique packets received, not counting resends. The number dropped is an estimate, based on the gaps in the packet number sequence.*)
```
09:30:20   Done
```

RADCSEND

The `radcsend.log` (in earlier versions, this was `edmcsend.log`), is always in the program's directory. The log has this information before the start:

(*Test with: 20ms delay, 2 resends, 40MB file, MWINDOW of 2353 seconds = 39.2 minutes*)

```
EDMCSEND started on [Wed Nov 20 08:44:58 2002]
Mode selected is: [M]
Destination address is: [231.1.222.6]
Destination port is: [9511]
Processing file: [TESTDATA0004]
Relative file number: [1]
Delay after the first packet: [20]
Delay between packets: [20]
Number of packets to re-send: [20]
Number of resends: [2]
Time to live: [3]
Last file flag: [1]
Packet Data Size: [1020]
File Size: [40000000]
```

(*And at the end:*)

```
Total Packets: 117648
Exiting on [Wed Nov 20 09:24:11 2002]: [Done]
```

(*From this, the following can be can computed:*
Effective delay between packets: 2353 seconds divided by 117648 packets = 20 ms/packet(?).
Thruput: 40MB divided by 2353 sec = 0.017 MB/sec [remember, there were 2 resends])

Pre-Multicast Session Checklist

Prior to starting a multicast session, do the following:

1   Be sure that each SERVICE to be multicast is defined as MANDATORY.

2   Specify MCELIGBL = Y for SERVICES to be multicast.

3   Connect the SERVICE instances and the MULTCAST instance to the USER or WORKGROUP instance.

4   Set the MULTCAST instance parameters, as outlined below:

The most common problems are determining the start time and duration of the Collection window (CGMTTIME and CWINDOW). The Collection window should be long enough to allow clients to:

—   Connect to the Configuration Server to determine the needed resources, and

—   Connect to the Multicast Server to exchange the request information.

> Be advised that the internal clocks on the machines (clients and servers) should be synchronized to within a few minutes of each other, *with respect to GMT*. This means that CGMTTIME should be a few minutes *earlier* than the time that the clients are expected to connect, and the CWINDOW time should be a few minutes *later* than the time that the last client is expected to complete the collection/request phase of multicast.

5   Specify identical values for the parameters DELAYBP and DELAYFP.

> The value of DELAYBP is determined by the calculations shown in the section, Calculate DELAYBP and MWINDOW, and modified by the results of the tests shown in the section, Examine the Test Logs.

The value of MWINDOW (calculated in the section, Calculate DELAYBP and MWINDOW) should be padded 20–50% to allow for unforeseen influences, such as network congestion.

▶ Remember to base these calculations on the total number of bytes of all the files that are expected to be transmitted.

Multicast Program-Call Schematic

Figure 11 illustrates the sequence in which the various programs are called during a multicast session.

▶ The execution flows from *left to right* and *top to bottom*.

Figure 11   The call sequence of the multicast session programs
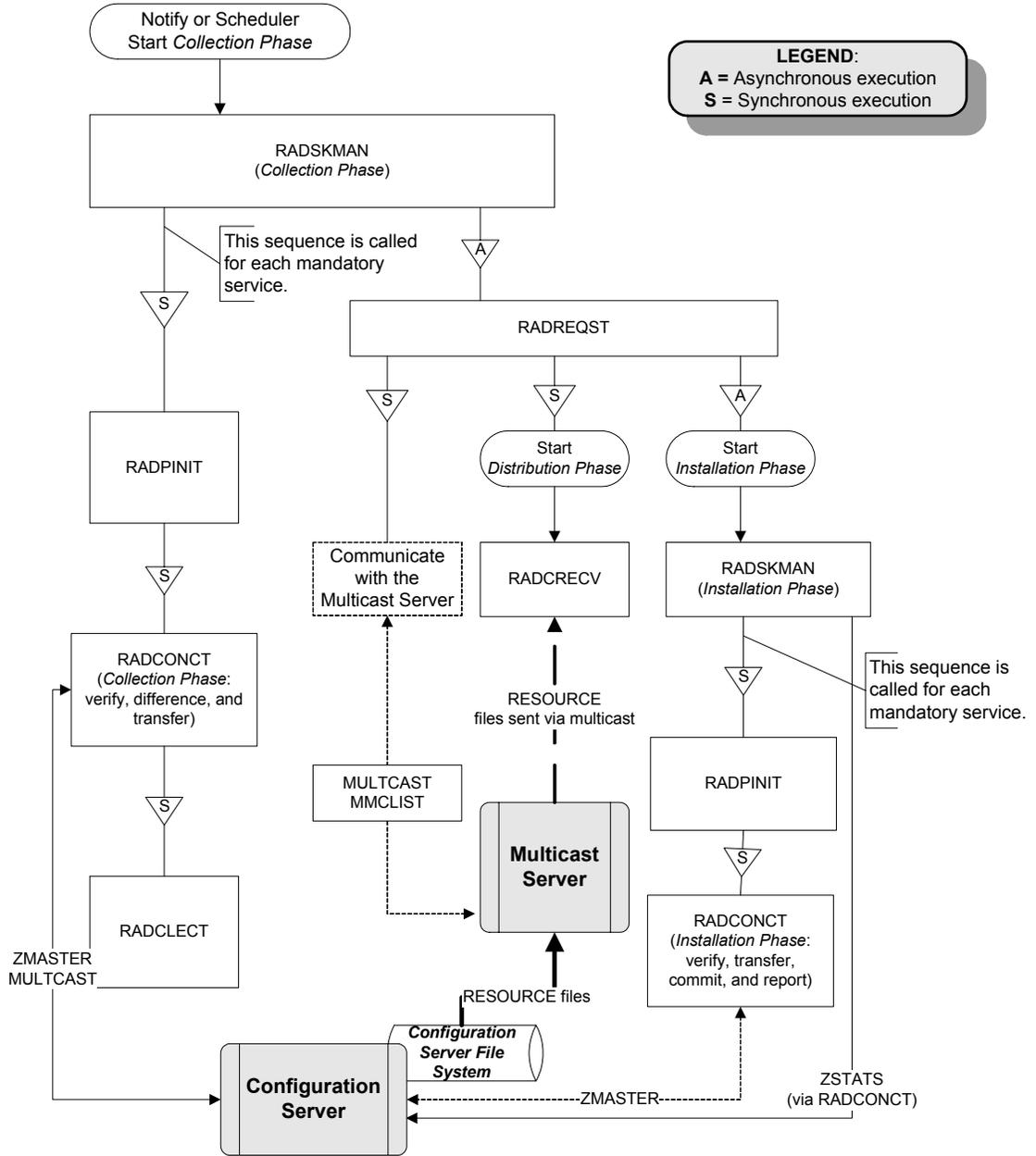
Troubleshooting

Querying the Various Logs

This section provides information on what to look for in the client logs (located in `IDMSYS\log`) when running a multicast session. This process works backwards through the logs in order to determine the point at which a problem occurred.

RADCRECV.log

Does it exist? If NO, check `RADREQST.log` (see **#2**).

- If YES, were all files received? If YES, done.
- If NO:
    — Was there a timeout? If YES, increase MWINDOW (or set to 0, don't use).
    — Were too many packets dropped? If YES, increase DELAYBP or RESENDS.
    — Did a file already exist?
    — Does `RADCLECT.log` show files rejected because of size?
    — Does `NVDCAST.log` show files rejected because of too few requests?
    — If none of the above, check `RADREQST.log`.

RADREQST.log

Does it exist? If NO, check `RADCLECT.log` (see **#3**).

- If YES, look at the last line of the log.

    ```
    Sleeping for n seconds - OK; more than 2 minutes before
    multicast.
    ```

    Ending with code:
    — **240** – OK, normal completion of multicast.
    — **241** – Can't open *object_name*.
    — **244** – Outside collection window (check times and time zone in log.).
    — **246** – No MMCLIST object; nothing to process. (See `RADCLECT.log`.)

— **016** – (Multicast Server error returned; refer to error message text.)

RADCLECT.log

Does it exist? If NO, look at `CONNECT.log` (see **#4**).

```
Added heaps: 1 with STATUS = RMS_REQ, 0 with STATUS = REJ_SIZ
– Normal operation.
```

(If no heaps are added, there will be no multicast.)

Note that if some files are not sent by multicast, this program will be called again, but will end with code **241** – "Multicast done."

Ended with code:

```
241 – Multicast NOT enabled – OK
   Multicast done – OK
   APPINFO.MCELIGBL = N, application not eligible for
   multicast. – OK
   Incorrect NVDLIB – Check the directory structure.
   Can't open [object_name]

243 – OK, normal completion of first pass.
```

CONNECT.log

Does it exist? If NO, it has been deleted or the Radia client never ran. The normal sequence of events is (look for these lines in the log, in this order):

a `Multicast available; starting Phase 1`

b `Adding Branch [MULTCAST] Priority [50]`

c `Requesting [n] files via multicast`

d `[ZGETAFIL] is launching program [RADCLECT.exe]`

e `END RADCLECT.exe ----- rc = [243]`

f `Collector loaded MMCLIST with [n] Files [1048576] Bytes`

g `RADCONCT exit status [859]`

h `RADCONCT [Installation is not complete (phased install
      process)]`

(Steps **c** through **h** will be repeated for each service that requires resources.)

i `Receiver launched [C:\PROGRA~1\Novadigm\RADREQST.exe] rc[0]`

j `RADSKMAN Exit code [859]`

Wait for multicast to finish, unless some condition causes RADREQST to prematurely terminate.

k   `CMD LINE = mname=engsvr1,dname=SOFTWARE,ip=engsvr1
     ,port=3464,cat=n,mcast=2`
     (*This is the start of the second phase, after multicast has finished.*)

Troubleshooting Tips

This section provides information on how HP OpenView Using Radia Multicast determines if it is within the collection window, and how to troubleshoot a failed multicast session.

How does HP OpenView Using Radia Multicast determine if it is within the collection window?

When the Radia client exchanges objects with the Configuration Server, a MULTCAST object is returned to the client. During the resource determination the client puts, into the MULTCAST object, the time that it (the MULTCAST object) was downloaded. The variable is LCLTIME.

When the client has collected all the resource requirements, it opens the MULTCAST object again and performs the following sequence:

- Computes the Configuration Server's GMT time from ZMGRDATE, ZMGRTIME, and ZMGRTMZN.

- Obtains its LCLTIME and computes the "clock difference" as *RCS_GMT_time* minus *client_GMT_time*.

- Computes the current *RCS_time* by adding the "clock difference" to the current *client_time*.

- Compares the current *RCS_time* with the Collection *start* and *end* times (which are computed using CGMTTIME, CGMTDATE, and CWINDOW).

  — If the current *RCS_time* is between the Collection *start* and *end* times, it proceeds to exchange, with the Multicast Server, the MULTCAST and MMCLIST objects.

  — If the current *RCS_time* is not between the Collection *start* and *end* times, it immediately starts phase 2.

- If the Multicast Server has an existing, but un-started, session for the MULTCAST instance and IGMP address that were sent by the Radia client, the client's requests are processed.

— If the transmission has started, the Multicast Server returns an error, indicating that the collection window is closed.

— If there is no session for these parameters and the collection window is open, the Multicast Server will create a session.

- The Radia client re-computes the "clock difference," based on the time returned by the Multicast Server. Two minutes before the scheduled transmission start-time, the client will "awaken" and start the receiver.

How can this sequence fail?

- If the Multicast Server is not on the same machine as the Configuration Server, and the clocks are not synchronized, the check done by the Radia client against the Configuration Server's clock might pass, but the check done by the Multicast Server might not.

- If a MULTCAST object already exists on the Radia client (perhaps left over from an aborted connection), the client will be using old time stamps. Therefore, when the Multicast Server receives the request from the client, the request will not be in the collection window.

- It is possible that the Radia client is very near the end of the collection window, and by the time it communicates with the Multicast Server, the collection window will be closed.

- If the Radia client has an extensive list of resource requirements, the collection window might close before all of them have been submitted to the Multicast Server.

— Those that are submitted after the window closes but before the transmission begins will be accepted.

— Those that are submitted after the transmission begins will be rejected, and the client will receive only those that were submitted prior to transmission.

If the multicast process terminates abnormally during testing and the logs have been examined, perform the following steps to ensure a clean restart.

1 Stop the Configuration Server, delete all of its logs, and restart it.

2 Stop the Multicast Server, delete all NVDCAST logs, and restart it.

3 On the Radia client: delete all logs, and delete any MULTCAST and MMCLIST objects in the Radia client sub-directories.

4 Using HP OpenView Using Radia, uninstall the applications that are to be used for testing.

Ensure that all the files are deleted as well.

5   Set up the MULTCAST object with new collection window.

6   Start the Radia client process.

Index

.

./install, 33

A

ADDRESS field, 46
address parameter, mcast.cfg file, 53
advanced configuration options, 47
    BYPASCON, 47
    MCORDER, 47
    MWINDOW, 47
APPINFO, 78
Application Manager, 21

B

brecv, 68
broadcast
    radcrecv, 68
    radcsend, 68
    tests, 68
bsend, 68
BYPASCON, 39, 48, 56

C

calculating parameters, 71
CGMTDATE, 46, 79
CGMTTIME, 46, 74, 79
checklist, multicast session, 74
Clean up-and-Reporting phase, 28
client time, 79
Collection phase, 26
Collection start and end times, 79
completion status, 29
Configuration phase, 24
Configuration Server, 21, 64

and RMSSTATS, 64
    configuring, 39
    nvdcast.rc, 64
configuring
    Configuration Server, 39
    Multicast Server, 39
    time zone offsets, 47
connect.log, 78
creating a multicast instance, 49
customer support, 4
CWINDOW, 46, 74, 79
cwindow parameter, mcast.cfg file, 54

D

Daylight Saving Time. *See* DST
DELAYBP, 71, 74, 77
DELAYFP, 74
Distribution phase, 27
document map, 15
DST, 57, 59
dynamic multicast windows, 45

E

edmcsend.log, 73

G

GMT, 57, 58, 79
Greenwich Mean Time. *See* GMT

H

HTTP, 21

I

IDMSYS, 73
IDMSYS.log, 77