Radia Client Automation Enterprise Multicast Server

For the Windows® operating system

Software Version: 9.00

Reference Guide

PERSISTENT

Legal Notices

Warranty

The only warranties for products and services are set forth in the express license or service agreements accompanying such products and services. Nothing herein should be construed as constituting an additional warranty. Persistent Systems shall not be liable for technical or editorial errors or omissions contained herein. The information contained herein is subject to change without notice.

Restricted Rights Legend

Confidential computer software. Valid license from Persistent Systems or its licensors required for possession, use or copying. No part of this manual may be reproduced in any form or by any means (including electronic storage and retrieval or translation into a foreign language) without prior agreement and written consent from Persistent Systems.

Copyright Notice

Trademark Notices

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

UNIX® is a registered trademark of The Open Group.

Acknowledgements

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

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

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

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

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

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

Documentation Updates

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

- Software Version number, which indicates the software version.

- Document Release Date, which changes each time the document is updated.

- Software Release Date, which indicates the release date of this version of the software.

To check for recent updates or to verify that you are using the most recent edition of a document, go to:

http://support.persistentsys.com/

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

For more details, contact your Persistent sales representative.

Support

Persistent Software support provides customer self-solve capabilities. It provides a fast and efficient way to access interactive technical support tools needed to manage your business. As a valued support customer, you can benefit by being able to:

- Search for knowledge documents of interest

- Submit and track support cases and enhancement requests

- Submit enhancement requests online

- Download software patches

- Look up Persistent support contacts

- Enter into discussions with other software customers

- Research and register for software training

To access the Self-solve knowledge base, visit the Persistent Support home page.

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

To register for a Persistent Support ID, go to: Persistent Support Registration.

Contents

Chapter 1

Introduction

Multicasting

This chapter provides an overview of the multicast method of transmitting files, and how multicasting compares to the standard method, **unicasting**. It also details the benefits of multicasting, and how the Multicast Server works with other Client Automation products.

Unicasting vs. Multicasting

The standard method of data transmission is **unicasting** (see figure Unicasting Scenario). In a unicast scenario, a server communicates with multiple agents individually and at different times, and transmits data to each during its communications session.

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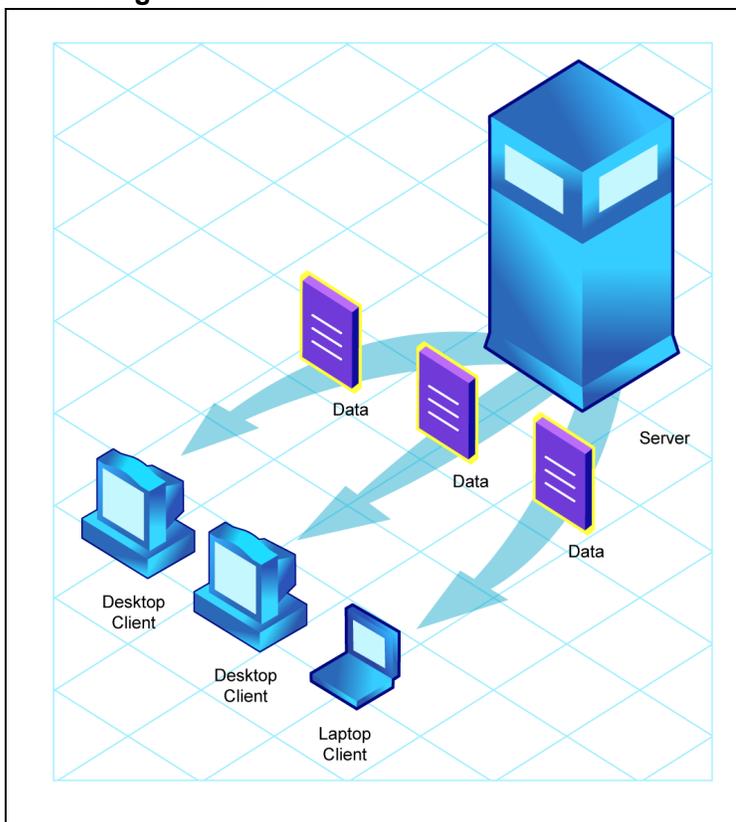
Since the server must repeatedly transmit identical data, the unicast method is very time-consuming.

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

Multicasting Scenario
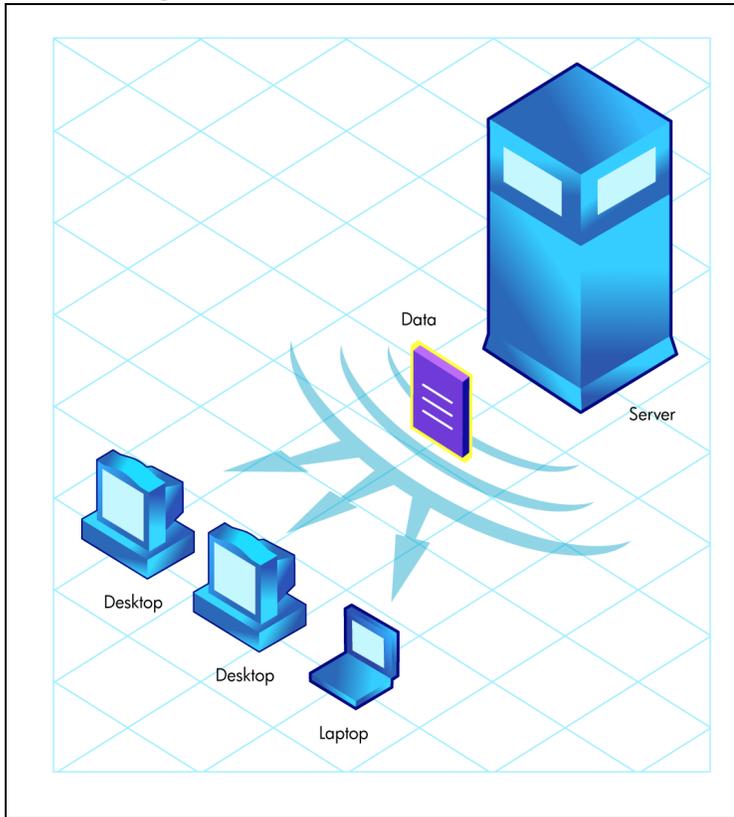
Benefits of Multicasting

Multicasting provides the following benefits:

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

Abbreviations and Variables

Abbreviations Used in this Guide

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

Variables Used in this Guide

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

Chapter 2

The RCA Multicast Server

Multicasting

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 that are bundled into that payload.

The RCA 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 that is by the participants is sent, and the participants retrieve only the data that they have requested.

Benefits of Radia Client Automation Multicasting

Multicast technology is seamlessly integrated with Radia Client Automation 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), the Multicast Server provides an additional dimension to its existing architectural focus—using minimum resources in order to bring the desktop computing environment to its desired state.

An administrator also has the ability to perform a "centralized" configuration of the RCA agents, and the ability to group RCA agents 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 in order to setup RCA multicast distribution is performed centrally by the RCA administrator. The RCA agents obtain all required multicast configuration parameters in the framework of the standard RCA agent connect. The parameters can be changed regularly or as needed, and the RCA agent 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, RCA provides the ability (using the standard administration tools that are provided with the product) to associate users that have similar requirements into separate multicast groups (for example, file servers and desktops).

Increasing Network Efficiency

The Radia multicasting utility enables you to maximize the functionality of your existing network bandwidth. This is done by configuring the Multicast Server to concurrently transmit a single data-stream to multiple RCA agents.

Each RCA agent can distinguish between the files and components that are being delivered in the multicast stream, and retrieve only those that it needs. As a result, transfer time is minimized, and the RCA agent'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 RCA functionality to determine whether all of the required resources have been delivered in the multicast phase. It then uses the standard protocol—the RCA agent connect—to deliver any unsatisfied requirements. This protects against the "thrashing" that can occur in broadcast/multicast type protocols when the quality of the communications 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.

Note: The guaranteed delivery feature is specific to Windows operating systems.

Results Reporting

Since the Multicast Server is an extension of the RCA product family, results of a multicast session are reported to the RCA Configuration Server Database (CSDB). This information can then be mined for reporting and auditing. For more information on how this is accomplished and the benefits of this feature, see Appendix B, .

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 that services are mandatory, see the *Radia Client Automation Enterprise Application Manager and Application Self-Service Manager Reference Guide*.

RCA uses the standard IGMP. The Multicast Server requires:

- The availability of HTTP, and

- Routers to be enabled for the IGMP protocol.

Note: 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.

The Multicast Process

During the multicast process, the RCA agent 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 the figure Multicast Time Windows and Phases.) The time windows are then subdivided into four **phases**, during which the RCA agent 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 RCA agent, the Configuration Server, and the Multicast Server. The RCA agent is involved in all four phases of the process, whereas the Multicast Server and the Configuration Server are active only during some phases. 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

> **Note:** The parameters of these phases are configured in multicast instances in the Configuration Server Database.

Multicast Time Windows and Phases
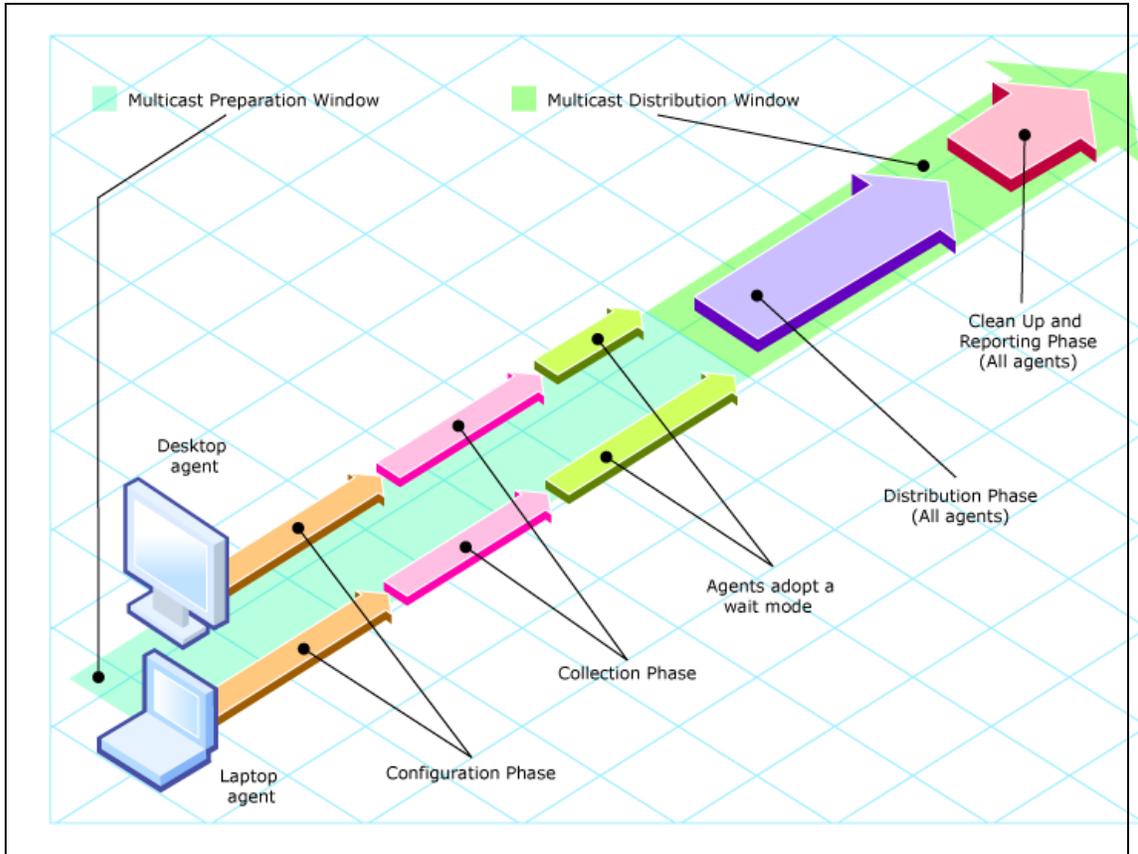
Configuration Phase

(RCA agents –Configuration Server) and (RCA agents –Multicast Server)

The configuration phase (shown in figure Configuration Phase (first part) and figure Configuration Phase (second part)) occurs during a standard RCA agent connect with the Configuration Server. An RCA agent 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.

Configuration Phase (first part)
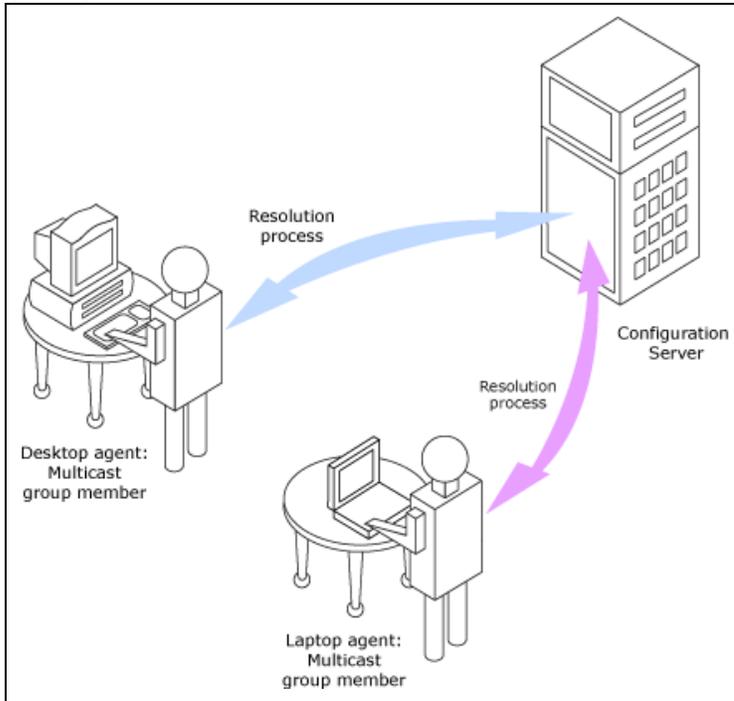
If a RCA agent is eligible, it then connects to the Multicast Server and exchanges the MULTCAST object, which contains the multicast parameters. If the connection fails in this phase, the RCA agent will perform normal, point-to-point retrieval (with the Configuration Server, using TCP/IP without multicast) to obtain all resources that are required in order to achieve the desired state.

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

Configuration Phase (second part)
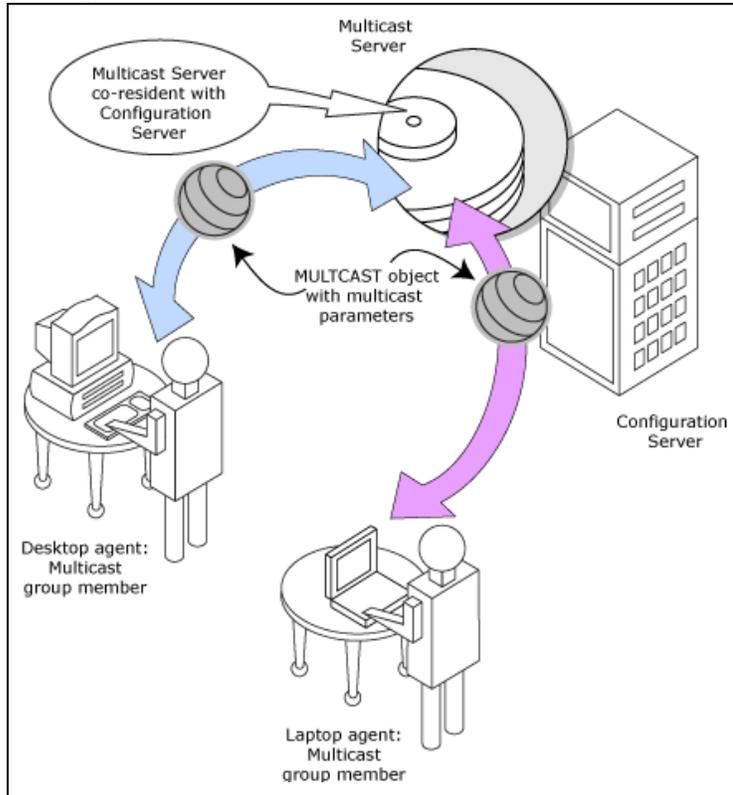
Collection Phase

(RCA agents –Multicast Server)

In this phase (assuming the criteria to be eligible for multicast distribution, as described in the first phase, have been met), the RCA agent 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 RCA agent requests and prepares to transmit the files at the scheduled time. The RCA agent adopts a wait mode in anticipation of the multicast transmission (See figure Collection Phase).

Collection Phase
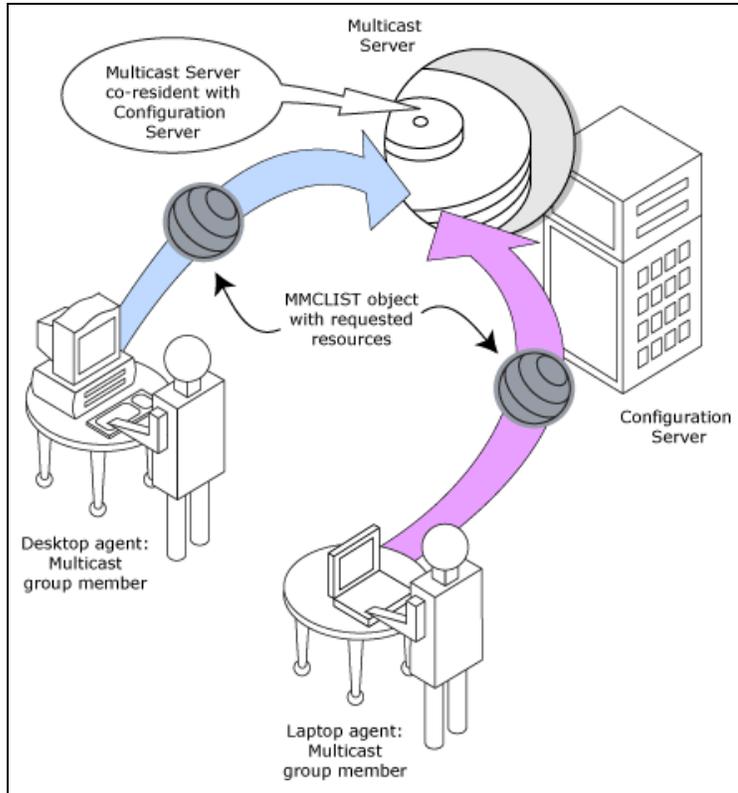
Distribution Phase

(RCA agents –Multicast Server –Configuration Server)

The third phase is the actual multicast transmission of the files (see the figure Distribution Phase). 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 RCA agents involved. As files arrive, they are checked (in the MMCLIST object) by the RCA agent to determine if they are required. If required, they are read from the multicast stream and marked to indicate that they have been received.

Distribution Phase
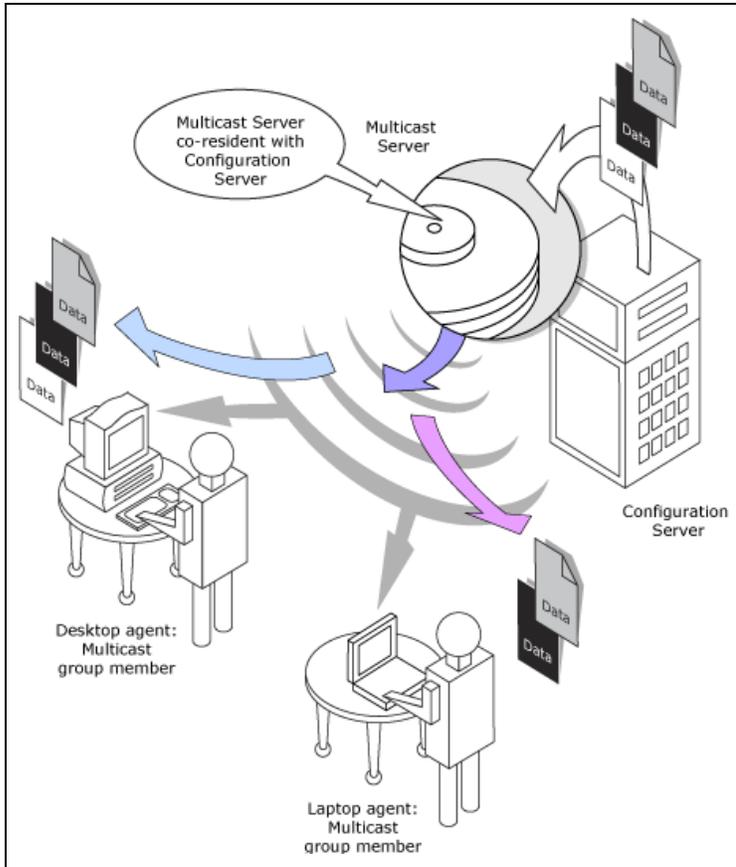
Clean up-and-Reporting Phase

(RCA agents –Configuration Server)

The final phase of the process runs when all required files have been received by the RCA agent, or the distribution time window has expired. At this time, the RCA agent checks for the presence of all required resources, and uses standard RCA communications techniques to obtain, from the CSDB, 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 CSDB (see figure Clean up-and-Reporting Phase).

Clean up-and-Reporting Phase
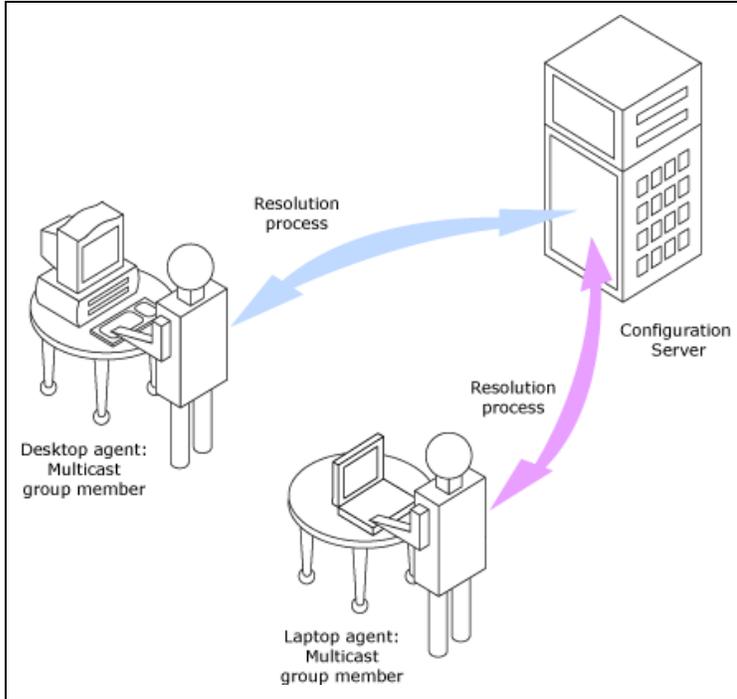
Completion Status Information

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 RCA agent.

See Appendix B, "Multicast Results for Reporting" on page 42 for more information on how multicast session results can be used for auditing and reporting.

Chapter 3

Configuring an RCA Multicast Server Environment

RCA Multicast Server

The Multicast Server is a processing engine that requires access to the resources in the Configuration Server Database. After the Multicast Server has been installed, it is necessary to make a few modifications to the CSDB.

When the RCA agent connects to the Multicast Server, it sends up the **MULTCAST** object, from which the Multicast Server obtains control information for the multicast session. The RCA agent also sends up the list of required files in an object named as **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 MULTCAST Class" below) in the CSDB. They are delivered to the RCA agent during normal resolution, and are stored in the RCA agent's subdirectory for each service.

RCA Configuration Server Database Changes

For Multicast to work, disable all the data SAPs and create a new data SAP with ROLE=B. Also set the Core SETTINGS class attribute, RCSDATA to Y.

In order for multicasting to execute properly, a few modifications must be made to the CSDB. 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 RCA agent objects (**BYPASCON**), and
- the duration of a multicast session (**MWINDOW**).

The MULTCAST Class

The MULTCAST Class of the POLICY Domain is where an RCA multicasting session is configured and scheduled. In order to perform these tasks, access the Admin CSDB Editor by clicking Start, Programs, Radia Client Automation Administrator, and Client Automation Administrator CSDB Editor.

The MULTCAST Class should not be modified during a multicast session.

Before starting the editing process, do the following:

On the Admin CSDB Editor tool bar, click **View**, **List View**, and select **Details**.

On the Admin CSDB Editor 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**.

The table "MULTCAST Class Instance Attributes" 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.

MULTCAST Class Instance Attributes

Attribute	Value	Description
DOMAIN	&(ZOBJDOMN)	Specifies the domain name. **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 whether multicast is enabled. The default is **Y**.
MODE	B or M	Broadcast or Multicast. The default is **M**. **Note:** 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 do not support IGMP.
MCORDER	B, A, or S	For information about this attribute, see "MCORDER" on page 28. The default is **S**.
ADDRESS	229.0.0.0	Specifies a valid broadcast/multicast address. Also, this attribute can accept up to 19 characters in order to accommodate a range of addresses, which is important to the dynamic multicast feature. For more information, see "ADDRESS" on page 26.

Attribute	Value	Description
PORT	9512	Broadcast or Multicast UDP port.
DELAYFP	20	Delay (in milliseconds) after the first packet is sent.
DELAYBP	5	Delay (in milliseconds) between packets. This gives the receiver time to close the current file and open one for the next resource. For information on calculating this value, see "Calculate DELAYBP and MWINDOW" on page 52.
RESENDS	1	Number of times a packet will be re-sent. **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. **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**. Also, this attribute is important to the dynamic multicast feature. For more information, see "CGMTDATE" on page 27.
CGMTTIME	14:00:00	The start time of the collection period. The format is **HH:MM:SS**, expressed in base-24 time. Also, this attribute is important to the dynamic multicast feature. For more information, see "CGMTTIME" on page 27. **Note:** This variable is specified in GMT. See the section "Time Zone Adjustments" on page 40 for important time zone information.
CWINDOW	60	Duration (in minutes) of the collection phase for all RCA agents to register their list of required files. This attribute, as it links to the dynamic multicast feature, is discussed in the section, "Multicast Dynamic Windows" on page 26. **Note:** When specifying the duration of your collection

Attribute	Value	Description
		window, be sure it is sufficient to collect all the files in the request list.
MDELAY	1	Delay (in minutes) between the close of the collection window and the start of the multicast.
MWINDOW	60	For information on this attribute, see "MWINDOW" on page 29. **Note:** For information on calculating this value, see "Calculate DELAYBP and MWINDOW" on page 52.
TTL	3	Number of router "hops."
BYPASCON	UserJoe	For information on this attribute, see "BYPASCON" on page 29.
ALTADDRM	208.244.225.46	The IP address of Multicast Server for object exchange.
ALTPORTM	3463	The port of Multicast Server for object exchange. **Note:** 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 HPCA ports (such as that of the Configuration Server). Therefore, be prepared to adjust accordingly, as other network software might also conflict.
MINREF	1	The minimum number of RCA agents 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  = 20071016
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 RCA agents.

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 concurrently active, 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 or 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 (inclusive) 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).

> **Note:** 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 RCA agent will be informed that no IGMP address is available.

Note: 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.

Note: The duration specified in the CWINDOW field, as well as all other duration-type fields, will be used as described in the table "MULTCAST Class Instance Attributes".

CGMTTIME

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

The first RCA agent 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.

Note: This variable is specified in base-24 time, in relation to its offset from GMT. For additional GMT-offset information, see Appendix A, "Time Zone Adjustments" on page 40.

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 RCA agent 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 RCA agents.

> **Note:** All RCA multicast software converts the local system clock to GMT before doing any comparisons or calculations.

For an overview of GMT and time zone calculations, see Appendix A, "Time Zone Adjustments" on page 40.

Advanced Multicast Configuration Options

Three multicast-specific variables (MCORDER, MWINDOW, and BYPASCON) are in the MULTCAST Class. A fourth, MCELIGBL, is in the ZSERVICE Class. This section describes these variables, and how they can be customized using the Admin CSDB Editor.

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.

> **Note:** Do not edit the _BASE_INSTANCE_.

MCORDER

If a service is mandatory but not eligible for multicast, it will be downloaded to and installed on the RCA agent directly from the CSDB. MCORDER enables an administrator to specify when to download and when to 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, see the *Radia Client Automation Enterprise Application Manager and Application Self-Service Manager Reference Guide*.

The MCORDER attribute is located in the PRIMARY.POLICY.MULTCAST Class. It has three values, as described in the table "MCORDER Values".

MCORDER Values

Value	Result
B	Download the service and install it on the RCA agent BEFORE the multicast session.
A	Download the service and install it on the RCA agent AFTER the multicast session.
S	SPLIT the process so that it downloads the service to the RCA agent 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 an RCA agent to run before terminating its multicast "receive" activity. The default is 0, which allows an RCA agent 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 RCA agent process does not run longer than is necessary to receive the data it requested.

> **Note:** For more information on calculating a value for MWINDOW, see Appendix D, "Implementation and Diagnostics" on page 52.

BYPASCON

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

However, if all the RCA agents' data payloads are identical, RCA multicasting can be configured to save on network bandwidth usage by limiting which RCA agents 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 RCA agents will exchange a control object and a needs list object with the Multicast Server.

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

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

> **Note:** 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 RCA agents.

Burst Mode

> **Note:** This feature is for multicast software distribution only. It is not supported for use with RCA OS Manager.

The *Guaranteed delivery* option (see "Guaranteed Delivery" on page 13) provides higher data-transmission rates and more reliable transmission of large files, but is designed for OS image downloads only—not software downloads.

The **Burst Mode** option improves the reliability and speed of software downloads.

Using Burst Mode

The Multicast Server has the basic mode of operation of sending one packet of data, then waiting the specified number of milliseconds (the value of DELAYBP) before sending the next packet.

For more information on DELAYBP, see:

- "DELAYBP" in the table "MULTCAST Class Instance Attributes", and

- "Calculate DELAYBP and MWINDOW" on page 52.

After every 20 packets (this number is configurable), each packet is re-sent one or more times (the value of *RESENDS* [see "RESENDS" in the table "MULTCAST Class Instance Attributes"]). Although this increases reliability, it does not guarantee that large files will be received.

The **burst mode** feature works differently. Instead of sending 1 packet and delaying for *DELAYBP*, Multicast Server sends multiple packets with no delay and then waits *DELAYBP* before re-sending them. So, the packets are still resent one or more times (the value of *RESENDS*), but in *bursts*, which gives a throughput of n multiplied by the basic mode, which is equivalent to **n=1**.

> **Note:** This version of Multicast Server includes a version of nvdkit that is configured with a lower threshold for the delay—down to at least 2 milliseconds—which provides additional throughput that is independent of and in addition to burst mode.

Parameter Values Selection

Assuming the typical values of DELAYBP = 10 (ms) and -burst = 1, the following changes will affect the throughput as described.

- Setting DELAYBP = 5 will roughly double the throughput.

- Setting -burst = 5 will increase throughput by a factor of five.

When used in combination, the improvements in throughput, although not linear, will increase. For example:

- Setting DELAYBP = 5 and -burst = 5, the throughput will increase by a factor of approximately **9**.

> **Note:** These figures were obtained on a lightly loaded 1 GBps network. Actual results will vary according to network speed, reliability, and load.

A reasonable maximum value for -burst is 10, and the lowest value for DELAYBP would be 2.

Setting DELAYBP and Burst

To set DELAYBP and Burst, follow these steps:

1. To open the Admin CSDB Editor click **Start** >**Programs** > **Radia Client Automation Administrator** > **Client Automation Administrator CSDB Editor**.

2. In the tree-view, double-click **PRIMARY**, **POLICY**, and **MULTCAST**.

3. Specify a value for the DELAYBP variable in the MULTCAST instance.
   - Save the changes and close the Admin CSDB Editor.

4. Open Windows Explorer and navigate to *SystemDrive*:`\Program Files\Hewlett-Packard\CM\MulticastServer\etc`.

5. Specify a value for the -burst parameter in the Multicast Server's `mcast.cfg` file.
   - Save the changes and close Windows Explorer.

6. Stop the Multicast Server.

7. Restart the Multicast Server.

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" below 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 Client Automation Administrator** > **Client Automation Administrator CSDB Editor**.

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 shortcut 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 Admin CSDB Editor 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.

   > **Note:** 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 RCA agents.

Associating Groups with a Multicast Instance

In order to be included in a multicast session, an RCA agent must be part of a group (such as a workgroup or a department) that is associated with a multicast instance. After 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 RCA agent, and each RCA agent 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 an RCA agent can be associated with several multicast sessions, at any given time, only one session can be active for an RCA agent.

This section describes how to create this association using the Admin CSDB Editor. For detailed instructions, see the *Radia Client Automation Enterprise CSDB Editor Online Help*.

> **Note:** See the *Radia Client Automation Enterprise Application Manager and Application Self-Service Manager Reference Guide* for comprehensive information on how to include users in workgroups and departments.

To associate a group with a multicast instance:

1. Open the Admin CSDB Editor, and navigate to and open PRIMARY.POLICY.WORKGRP.

2. Right-click on the workgroup that you want to associate with multicast, and from the shortcut 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 Admin CSDB Editor, 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 Server and other RCA Components

The Multicast Server can be used with RCA components apart from RCA agents and the Configuration Server. For more information, see:

- "Multicast and the HPCA OS Manager" below
- "Multicast and the RCA Proxy Server" on page 34

Multicast and the HPCA OS Manager

> **Note:** The OS Manager will work with the Multicast Server in a Windows environment only.

> Therefore, the information in this section is not applicable to other operating-system environments.

The HPCA OS Manager supports guaranteed-delivery multicast so that large numbers of operating system images can be concurrently rolled out.

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 OS Manager System Agent with multicast support.

- The operating system images will be downloaded as by default the **Service Multicast Eligible** option is enabled for a service.

Configuring Multicast for use with the RCA OS Manager System Agent

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

To configure guaranteed-delivery multicast:

1. Open the RCA Administrator Database Editor.

2. Go to the appropriate Behavior instance.

3. Modify the OS Manager System Agent **Parameters** field as shown in the following sample:
   ```
   -multicast multicastIPAddress:3463 -mcastretrycount 1
   -mcastretrywait 240
   ```

   The following table describes the OS Manager System Agent parameters.

   RCA OS Manager System Agent 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 minutes).

4. If necessary, modify the Multicast Server configuration file, `mcast.cfg`, which is located in `<InstallDir>\MulticastServer\etc`.
   The following table describes some of the parameters of `mcast.cfg`.

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 26.
minref	The minimum number of RCA agents 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 RCA agents must register their list of required files.

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

Multicast and the RCA Proxy Server

The Multicast Server's dynamic-windows feature (see "Multicast Dynamic Windows" on page 26) can be used to pre-load a Proxy Server. For more information, including an example, see the *Radia Client Automation Enterprise Proxy Server Reference Guide*.

Preloading a Proxy Server

The Multicast Server can be used to preload the static cache of a Proxy Server.

Note: *Preloading* is the loading, onto a Proxy Server, of the applications that it is configured to distribute to RCA agents, before RCA agents request the application, and in anticipation of such requests. This practice eliminates the need to involve the Configuration Server.

Proxy Servers that are preloaded are configured in the same way as RCA agents that receive resources from a Multicast Server.

Note: Optionally, the Multicast Server dynamic windows feature can be used to preload Proxy Servers (see "Multicast Dynamic Windows" on page 26).

For additional information on preloading Proxy Servers, see the *Radia Client Automation Enterprise Proxy Server Reference Guide*.

The following section details how to preload a Proxy Server; it assumes that an administrator has a working knowledge of the Admin CSDB Editor. For additional instructions, see the *Radia Client Automation Enterprise CSDB Editor Online Help*.

To preload a Proxy Server:

Caution: It recommends that administrators become familiar with the operation of the Multicast Server before using it to preload a Proxy Server.

1. For the Proxy Servers that will participate in the preload, use the Admin CSDB Editor to create, in the CSDB, a POLICY.USER instance that matches the user ID that was specified during the Proxy Server installation, or later specified in the -static-user parameter in the `/etc/rps.cfg` file.
    The default user ID is **RPS**. However, it is often set to RPS_*machine_name*.

2. Use the Admin CSDB Editor to create a POLICY.WORKGROUP instance for the Proxy Servers in order to specify their distribution model for the preload of the static cache.

3. Use the Admin CSDB Editor to create a POLICY.MULTCAST instance (for example, MCPRELOAD) that is exclusively for Proxy Server preloads.

4. Edit the MCPRELOAD instance attribute values as follows.
    - ALTADDRM: Specify the IP address of where the Multicast Server is installed.

    - CGMTDATE and CGMTTIME: Specify a multicast session start date and time that will be active when the Proxy Server preload command is executed.

    > **Note:** The Proxy Server preload command must be issued at the same time as, or shortly after, the multicast collection session starts.

    The following figure illustrates an MCPRELOAD Instance with CGMTDATE and CGMTTIME entries that will initiate a multicast session for the Proxy Server preload on November 4, 2005 at 2 P.M.

Multicast Class MCPRELOAD Instance Attributes
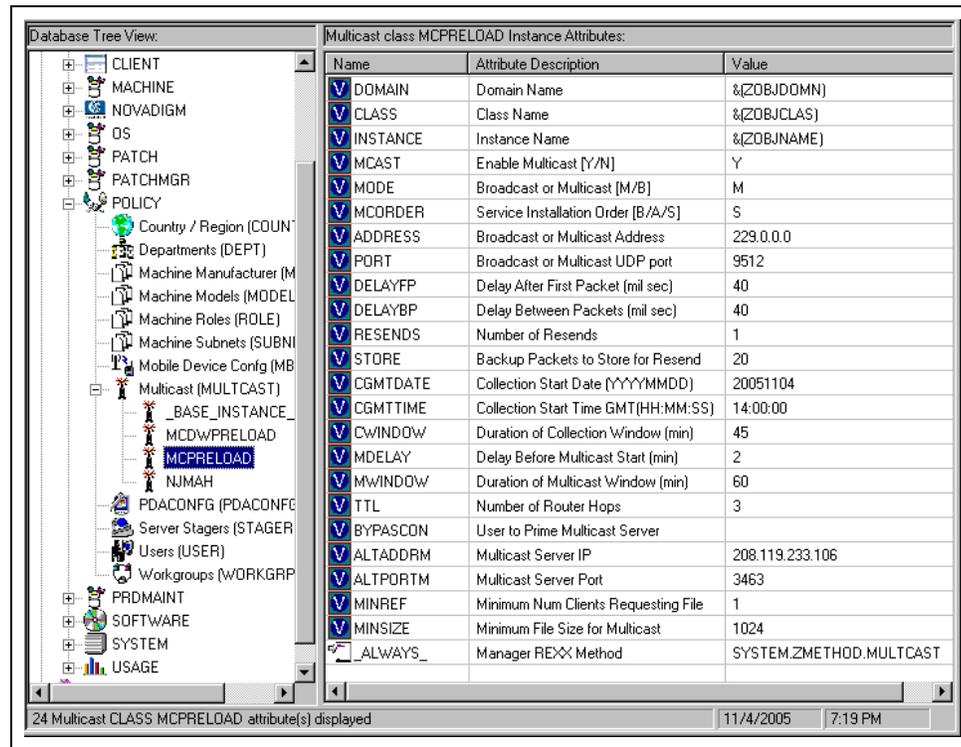
Name	Attribute Description	Value
DOMAIN	Domain Name	&(ZOBJDOMN)
CLASS	Class Name	&(ZOBJCLAS)
INSTANCE	Instance Name	&(ZOBJNAME)
MCAST	Enable Multicast [Y/N]	Y
MODE	Broadcast or Multicast [M/B]	M
MCORDER	Service Installation Order [B/A/S]	S
ADDRESS	Broadcast or Multicast Address	229.0.0.0
PORT	Broadcast or Multicast UDP port	9512
DELAYFP	Delay After First Packet (mil sec)	40
DELAYBP	Delay Between Packets (mil sec)	40
RESENDS	Number of Resends	1
STORE	Backup Packets to Store for Resend	20
CGMTDATE	Collection Start Date (YYYYMMDD)	20051104
CGMTTIME	Collection Start Time GMT(HH:MM:SS)	14:00:00
CWINDOW	Duration of Collection Window (min)	45
MDELAY	Delay Before Multicast Start (min)	2
MWINDOW	Duration of Multicast Window (min)	60
TTL	Number of Router Hops	3
BYPASCON	User to Prime Multicast Server	
ALTADDRM	Multicast Server IP	208.119.233.106
ALTPORTM	Multicast Server Port	3463
MINREF	Minimum Num Clients Requesting File	1
MINSIZE	Minimum File Size for Multicast	1024
ALWAYS	Manager REXX Method	SYSTEM.ZMETHOD.MULTCAST

Database Tree View:
- CLIENT
- MACHINE
- NOVADIGM
- OS
- PATCH
- PATCHMGR
- POLICY
  - Country / Region (COUNT
  - Departments (DEPT)
  - Machine Manufacturer (M
  - Machine Models (MODEL
  - Machine Roles (ROLE)
  - Machine Subnets (SUBNI
  - Mobile Device Confg (MB
  - Multicast (MULTCAST)
    - _BASE_INSTANCE_
    - MCDWPRELOAD
    - MCPRELOAD
    - NJMAH
  - PDACONFG (PDACONFG
  - Server Stagers (STAGER
  - Users (USER)
  - Workgroups (WORKGRP
- PRDMAINT
- SOFTWARE
- SYSTEM
- USAGE

24 Multicast CLASS MCPRELOAD attribute(s) displayed   11/4/2005   7:19 PM

As shown, the collection window will last 45 minutes—as specified by CWINDOW—after which the Multicast Server will stop the collection process.

■ There will 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 will start multicasting the files that were requested by the various RadiaCA agents.

5. Connect the POLICY.WORKGROUP instance for the Proxy Server Preload Apps to the POLICY.MULTCAST.MCPRELOAD instance.
   For details on making this connection, see "Associating Groups with a Multicast Instance" on page 31.

6. Issue the Proxy Server preload command when the CGMTTIME and CGMTDATE apply.

Preloading an RCA Proxy Server with Dynamic Windows

The Multicast Server offers the ability to configure dynamic (collection and transmission) windows, as detailed in "Multicast Dynamic Windows" on page 26. This means that, for a group of Proxy Servers that is being preloaded, multiple multicast distributions can be concurrently active.

The following MULTCAST instance values will initiate a daily (because no value is specified for CGMTDATE) multicast session that will start at 05:01:00 GMT. The multicast session will use up to 22 dynamic windows.

> **Note:** A Proxy Server preload must also be scheduled for this same time each day in order for it to participate in the multicast session.

```
ADDRESS = 235.0.0.000-021
CGMTDATE =
CGMTTIME = 05:01:00

CWINDOW = 45
MDELAY = 2
ALTADDRM = multicast_server_IP
```

Session Logs for a Preload Using Multicast

On a Proxy Server that receives a preload from a Multicast Server, the preload session logs are found in:

```
<ProxyServer>\logs\rps
```

The session logs that trace the activity of the collection, the requests, and the received items in a multicast session are, respectively:

● `connect.log`

● `radreqst.log`

● `radcrecv.log`

> **Note:** On a Multicast Server-preloaded Proxy Server, `connect.log` includes the collection information that is found in the `radclect.log` of an RCA agent multicast session.

Chapter 4

Using the RCA Multicast Server

Testing the RCA Multicast Server

Now that the RCA 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 an RCA agent perspective, and two from an administrator perspective.

The RCA agent-based method is not efficient in production because it is specific only for that RCA agent. 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 RCA agents (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 RCA agent machine.
- Connect the multicast instance to the RCA agent via a group.
- Set up a multicast instance with an active collection window.

RCA Agent vs. Administrator Testing

This section details the testing of Multicast Server from an RCA agent perspective (starting "RCA Agent Testing" below) and from an administrator perspective (starting "Administrator Testing" on next page).

RCA Agent Testing

To test the operation of the Multicast Server and the associated RCA agent software:

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

1. On the RCA agent machine, open a command window.

2. Change the directory to the database location where the RCA agent 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 RCA agent machine. The methods that are covered in the next section are more

appropriate for production because the Configuration Server Database can be accessed and then used to include RCA agents in a group, which can then be associated with a multicast instance.

This means that at a central location, multiple RCA agents can be configured for a multicast session. Also, 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

This section details testing Multicast Server from an administrator perspective.

Notify and TIMER

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

> **Note:** For comprehensive information on these features, see the *Radia Client Automation Enterprise Application Manager and Application Self-Service Manager Reference Guide*.

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

Appendix 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 correctly configured. 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). Also, **base-24** (a.k.a. military) time must be used when configuring this setting.

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

Therefore, a Configuration Server Database in New York, 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 Monday, July 09, 2007 at 2:35:15 (P.M.) local time on a Multicast Server in New York, USA, specify:

```
CGMTDATE=20070709
```

```
CGMTTIME=18:35:15
```

The value of the MULTCAST Class instance attribute, CGMTTIME (2:35:15 P.M.), 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 Monday, November 09, 2007 at 2:35:15 (P.M.) local time, on a Multicast Server in New York, USA, specify:

```
CGMTDATE=20071109
```

```
CGMTTIME=19:35:15
```

The value of the MULTCAST Class instance attribute, CGMTTIME (2:35:15 P.M.), 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 (19:35:15) that must be specified for the CGMTTIME attribute in order to schedule the specified multicast session.

Time Zone Overview

The figure Greenwich Mean Time (GMT) shows 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.

Greenwich Mean Time (GMT)
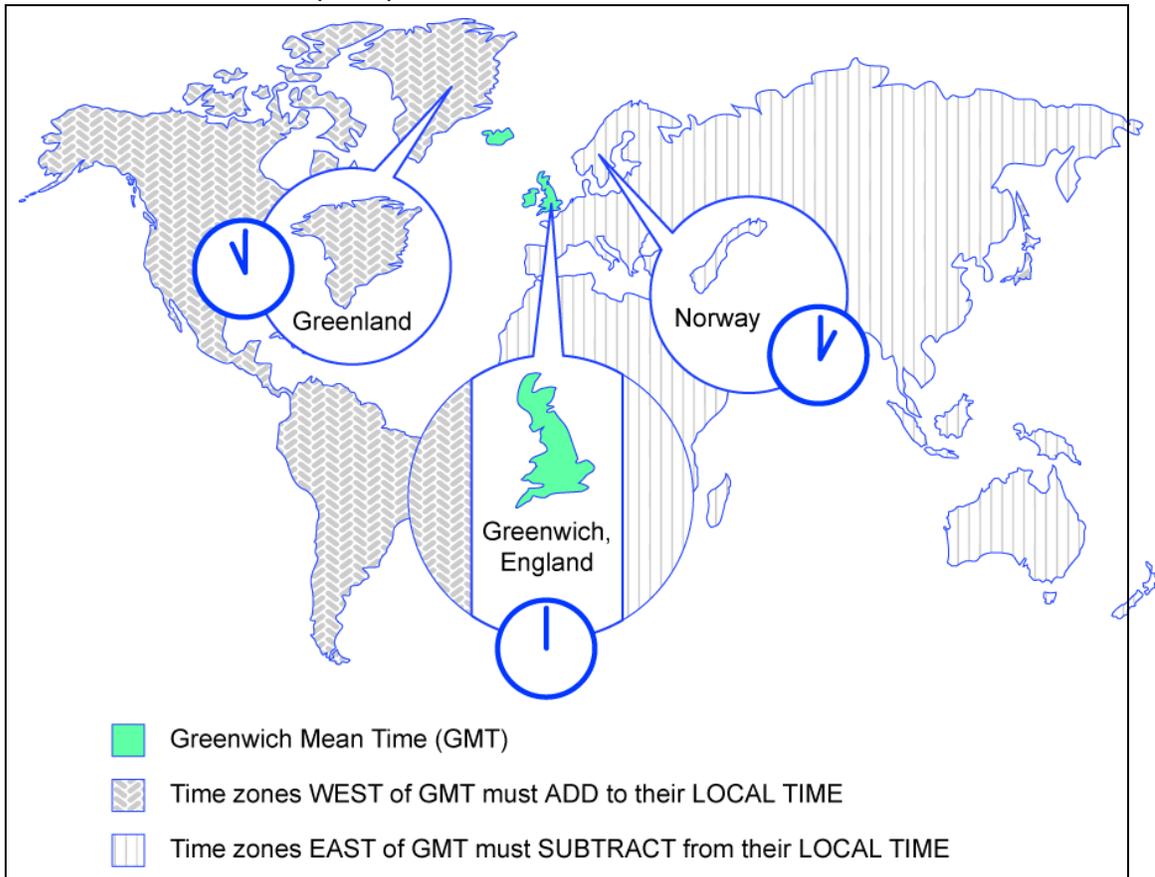
Automatic Adjustments for Daylight Saving Time

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

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

Appendix 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 Configuration Server Database. 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 the table "Multicast RMSSTATS Object Variables".

RMSSTATS Object

The RMSSTATS object can be sent to the Configuration Server Database. The database connections can be configured so that the multicast statistics can be examined.

For more information on the RMSSTATS object in the Configuration Server Database, see the section "RMSSTATS and the RCA Configuration Server Database" on page 44.

RMSSTATS

The table "Multicast RMSSTATS Object Variables" contains a list of the variables of the RMSSTATS object.

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*

Variable	Value
	• *IPaddr:port* are the address and port that the RCA agent 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*
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 RCA agent. Since the Multicast Server transmits files in the order of *most_files_requested* to *least_files_requested*, an RCA agent 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 an RCA agent will discard packets until its requested files are transmitted. Therefore, multicast durations need to be properly computed before being specified.

RMSSTATS and the RCA Configuration Server Database

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 host machine.

> **Note:** The `nvdcast.rc` file is for reporting purposes only; it is not required for standard multicast operations.

Multicast statistics can be mined and viewed with the Admin CSDB Editor. To enable this feature, configure the database connections that are shown in the following table.

Database Connections for Multicast Statistics

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

Appendix C

Network Test Modules

RCA provides Multicast test modules to help you tune the multicast and broadcast parameters for a network configuration.

> **Note:** The Multicast test modules are not included in the HPCA media. Contact HP Technical support to obtain these modules.

The `multicast_test_modules` directory contains:

- the **broadcast** send and receive files (`BRECV.CMD` and `BSEND.CMD`),

- the **multicast** send and receive files (`MRECV.CMD` and `MSEND.CMD`),

- the receive executable (`radcrecv.exe`),

- the send executable (`radcsend.exe`),

- the image-deployment send files (`gdmcsend.exe` and `gdmsend.cmd`),

- two application `.dll` files, and

- tour test files (`TESTDATA000`n) of varying sizes.

> **Note:** 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 requirements 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` to a temporary directory on the destination computer.

2. Run `mrecv.cmd`.

3. Copy the contents of `multicast_test_modules` 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.

> **Note:** If RCA is not installed, logs are created in the respective, current directories of each program. Otherwise, look for the `radcrecv.log` in the `log` subdirectory. 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 shown below. Following the syntax, the table "Syntax Parameters Defined" 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]]
```

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 one 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.
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.

Parameter	Definition
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).

Note: 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:

RADCSEND Parameters for BROADCAST

B	Indicates Broadcast
231.1.222.6	*address* (specified on the command line)
9511	*port*
TESTDATA000n	*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*

> **Note:** 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:

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:

RADCSEND Parameters for MULTICAST

M	Indicates Multicast

231.1.222.6	*address* (specified on the command line)
9511	*port*
TESTDATA000n	*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
```

> **Note:** 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:

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
```

Appendix D

Implementation and Diagnostics

This appendix offers information on:

- Implementing the multicast process
  - "Preliminary Parameter Calculation" below,
  - "Pre-Multicast Session Checklist" on page 54
  - "Multicast Program-Call Schematic" on page 55

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.**

MWINDOW1

```
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" on page 46 but substitute the value of DELAYBP with that which pertains to your network.

> **Note:** 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.

> **Note:** 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

> **Note:** The following examples contain explanatory comments (italicized test within parentheses) that will not be in the log.

RADCRECV

The `radcrecv.log` is in the same directory as the program. If the RCA agent is on the same machine, the log is in the `log` subdirectory of the RCA agent 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, 40 MB 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: 40 MB divided by 2353 sec = 0.017 MB/sec [remember, there were 2 resends])

Pre-Multicast Session Checklist

Before 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 RCA agents to:

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

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

   Note: Be advised that the internal clocks on the machines (RCA agents 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 RCA agents are expected to connect, and the CWINDOW time should be a few minutes *later* than the time that the last RCA agent is expected to complete the collection/request phase of multicast.

5. Specify identical values for the parameters DELAYBP and DELAYFP.

> **Note:** The value of DELAYBP is determined by the calculations shown in "Calculate DELAYBP and MWINDOW" on page 52 and modified by the results of the tests shown in "Examine the Test Logs" on page 53.

The value of MWINDOW (calculated in "Calculate DELAYBP and MWINDOW" on page 52) should be padded 20–50% to allow for unforeseen influences, such as network congestion.

> **Note:** 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

The figure "Call Sequence of the Multicast Session Programs" illustrates the sequence in which the various programs are called during a multicast session.

> **Note:** The execution flows from *left to right* and *top to bottom*.

Call Sequence of the Multicast Session Programs
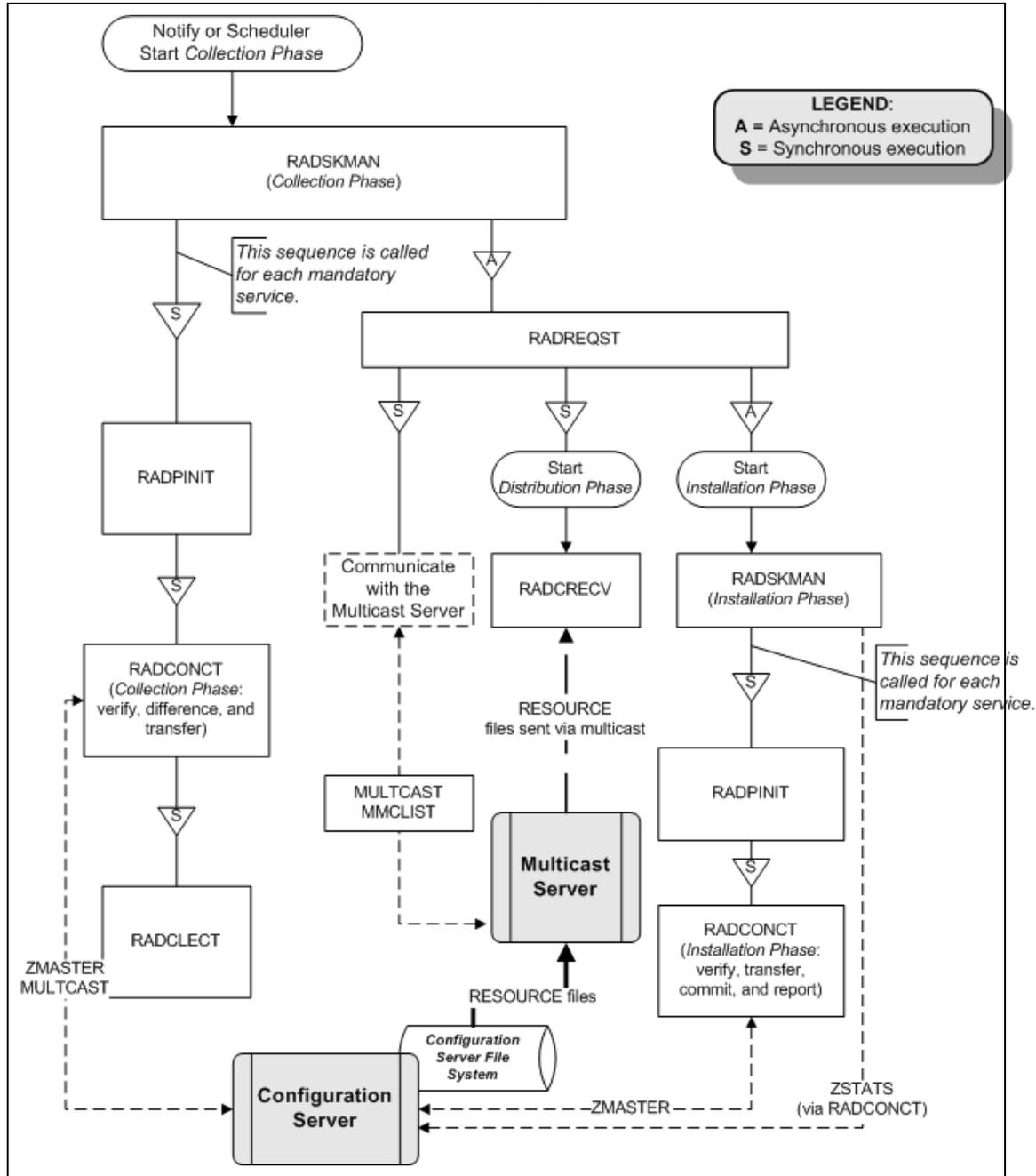
We appreciate your feedback!

If an email client is configured on this system, by default an email window opens when you click here.

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

Product name and version: Radia Client Automation Enterprise Multicast Server, 9.00

Document title: Reference Guide

Feedback: