

# BMC Marimba Client Automation Policy Management User Guide



Supporting

BMC Marimba Client Automation Policy Management  
version 9.0.00

November 2015



## Contacting BMC Software

You can access the BMC Software website at <http://www.bmc.com>. From this website, you can obtain information about the company, its products, corporate offices, special events, and career opportunities.

### United States and Canada

Address	BMC SOFTWARE INC 2101 CITYWEST BLVD HOUSTON TX 77042-2827 USA	Telephone	713 918 8800 or 800 841 2031	Fax	713 918 8000
---------	--	-----------	---------------------------------	-----	--------------

### Outside United States and Canada

Telephone	(01) 713 918 8800	Fax	(01) 713 918 8000
-----------	-------------------	-----	-------------------

© Copyright 2005 - 2015 BMC Software, Inc.

BMC, BMC Software, and the BMC Software logo are the exclusive properties of BMC Software, Inc., are registered with the U.S. Patent and Trademark Office, and may be registered or pending registration in other countries. All other BMC trademarks, service marks, and logos may be registered or pending registration in the U.S. or in other countries. All other trademarks or registered trademarks are the property of their respective owners.

BladeLogic and the BladeLogic logo are the exclusive properties of BladeLogic, Inc. The BladeLogic trademark is registered with the U.S. Patent and Trademark Office, and may be registered or pending registration in other countries. All other BladeLogic trademarks, service marks, and logos may be registered or pending registration in the U.S. or in other countries. All other trademarks or registered trademarks are the property of their respective owners.

IT Infrastructure Library® is a registered trademark of the Office of Government Commerce and is used here by BMC Software, Inc., under license from and with the permission of OGC.

Linux is the registered trademark of Linus Torvalds.

Oracle and Java are registered trademarks of Oracle and/or its affiliates. Other names may be trademarks of their respective owners.

UNIX is the registered trademark of The Open Group in the US and other countries.

The information included in this documentation is the proprietary and confidential information of BMC Software, Inc., its affiliates, or licensors. Your use of this information is subject to the terms and conditions of the applicable End User License agreement for the product and to the proprietary and restricted rights notices included in the product documentation.

## Restricted rights legend

U.S. Government Restricted Rights to Computer Software. UNPUBLISHED -- RIGHTS RESERVED UNDER THE COPYRIGHT LAWS OF THE UNITED STATES. Use, duplication, or disclosure of any data and computer software by the U.S. Government is subject to restrictions, as applicable, set forth in FAR Section 52.227-14, DFARS 252.227-7013, DFARS 252.227-7014, DFARS 252.227-7015, and DFARS 252.227-7025, as amended from time to time. Contractor/Manufacturer is BMC SOFTWARE INC, 2101 CITYWEST BLVD, HOUSTON TX 77042-2827, USA. Any contract notices should be sent to this address.

## **Customer support**

You can obtain technical support by contacting Customer Support by telephone or e-mail. To expedite your inquiry, see "Before contacting BMC."

### **Support by telephone or e-mail**

In the United States and Canada, if you need technical support and do not have access to the web, call 800 537 1813 or send an e-mail message to [customer\\_support@bmc.com](mailto:customer_support@bmc.com). (In the subject line, enter SupID:<yourSupportContractID>, such as SupID:12345). Outside the United States and Canada, contact your local support center for assistance.

### **Before contacting BMC**

Have the following information available so that Customer Support can begin working on your issue immediately:

- product information
  - product name
  - product version (release number)
  - license number and password (trial or permanent)
- operating system and environment information
  - machine type
  - operating system type, version, and service pack or other maintenance level such as PUT or PTF
  - system hardware configuration
  - serial numbers
  - related software (database, application, and communication) including type, version, and service pack or maintenance level
- sequence of events leading to the issue
- commands and options that you used
- messages received (and the time and date that you received them)
  - product error messages
  - messages from the operating system, such as file system full
  - messages from related software

## **License key and password information**

If you have questions about your license key or password, contact BMC as follows:

- *(USA or Canada)* Contact the Order Services Password Team at 800 841 2031, or send an e-mail message to [ContractsPasswordAdministration@bmc.com](mailto:ContractsPasswordAdministration@bmc.com).
- *(Europe, the Middle East, and Africa)* Fax your questions to EMEA Contracts Administration at +31 20 354 8702, or send an e-mail message to [password@bmc.com](mailto:password@bmc.com).
- *(Asia-Pacific)* Contact your BMC sales representative or your local BMC office.

# Contents

Preface . . . . .	13
Accessing product documentation . . . . .	13
Using the documentation channel . . . . .	13
Searching across the BMC Marimba Client Automation documentation set	14
Using the Channel Store . . . . .	15
Related documentation . . . . .	16
Section I    Getting started . . . . .	19
Chapter 1    Overview . . . . .	21
Policy Management components . . . . .	21
Policy Management system architecture . . . . .	22
Workflow for using policies to update endpoints . . . . .	24
Basic workflow . . . . .	25
Chapter 2    Initial configuration . . . . .	27
Editing attributes of a configuration object (optional) . . . . .	28
Setting directory services for repeaters . . . . .	28
Configuring Sun Java System Directory Server (Sun One) directory replicants	29
Creating the directory service mapping file . . . . .	29
Restarting Policy Manager . . . . .	31
Using Policy Service with a new plug-in . . . . .	32

Chapter 3	Integrating directory services . . . . .	33
	What are directory services? . . . . .	34
	Integrating with Active Directory and ADAM / AD LDS . . . . .	34
	Active Directory and ADAM / AD LDS overview. . . . .	34
	Integrating with Active Directory . . . . .	36
	When to use automatic discovery for Active Directory . . . . .	37
	Schema modifications to Active Directory and ADAM / AD LDS. . . . .	38
	Synchronizing Active Directory to ADAM / AD LDS . . . . .	43
	Installation considerations for Active Directory . . . . .	44
	Targeting Active Directory groups. . . . .	45
	The global catalog. . . . .	48
	Automatic discovery of the global catalog. . . . .	49
	Viewing a large number of groups . . . . .	50
	Integrating with Sun Java System Directory Server (Sun One) . . . . .	52
	Namespace design . . . . .	52
	Multiple enterprises . . . . .	53
	Extending the LDAP schema . . . . .	55
	Using schema extensions. . . . .	59
	Policy Management configuration properties . . . . .	60
	Using LDIF to create sample policies. . . . .	62
	Policy Manager naming conventions. . . . .	64
Chapter 4	Setting up user accounts . . . . .	65
	Policy Management plug-in publishing . . . . .	67
	Policy Manager CLI permissions . . . . .	67
	Policy Manager permissions . . . . .	68
	Roles and access to Policy Manager features. . . . .	68
	Identifying users who can access Policy Manager. . . . .	70
	Directory tree permissions needed for creating policies . . . . .	71
	Setting permissions for Active Directory . . . . .	71
	Setting permissions for Sun Java System Directory Server (Sun One) directory	73
	Setting permissions for ADAM / AD LDS. . . . .	74
	Permissions for a multidomain forest environment (Active Directory only) .	74
	Subscriptions container permissions . . . . .	75
	CMS directory service and Policy Management plug-in permissions . . . . .	77

Chapter 5	Setting up collections . . . . .	81
	What is a collection? . . . . .	82
	LDAP query collections . . . . .	82
	Overview of LDAP query collection . . . . .	83
	Prerequisites for using LDAP query collections . . . . .	84
	Configuring LDAP query collections. . . . .	84
	Creating an LDAP query collection . . . . .	85
	Viewing LDAP query collections . . . . .	87
	Previewing the results of an LDAP query collection . . . . .	89
	Modifying an existing LDAP query collection . . . . .	90
	Deleting LDAP query collections . . . . .	91
	Refreshing LDAP query collections . . . . .	92
	Database query collections: integration with the inventory module . . . . .	92
	Collection modes . . . . .	93
	Installation issues when using database query collections . . . . .	96
	Working with database query collections . . . . .	98
Section II	Using Policy Manager . . . . .	103
Chapter 6	Obtaining user and group information from a transmitter . . . . .	105
	Overview: Using a transmitter as the source for users and groups . . . . .	106
	Issues when sourcing with transmitters. . . . .	108
	Limitations when sourcing with transmitters . . . . .	109
Chapter 7	The subscription configuration object . . . . .	111
	Attributes of the subscription configuration object . . . . .	113
	The BMC Marimba Client Automation configuration object . . . . .	114
	Attributes of the configuration object . . . . .	115
	Editing attributes of a configuration object . . . . .	116
	Entry point for Policy Management policy objects. . . . .	117
	BMC Marimba Client Automation computer entries created by Report Center collections . . . . .	117
	Schema mapping configuration parameters. . . . .	118
	Example of a customized schema mapping configuration. . . . .	119
	Collections configuration parameters . . . . .	121

	Miscellaneous configuration parameters . . . . .	122
Chapter 8	Policy Service configuration and implementation . . . . .	125
	Policy Management state verification and retry . . . . .	126
	Client machine name . . . . .	127
	Unique identification of targets . . . . .	127
	Policy Service and schedule enforcement . . . . .	128
	Processing immediate and scheduled events. . . . .	129
	Assigned states and endpoint states . . . . .	131
	Types of schedules . . . . .	132
	Life cycle of a package schedule . . . . .	133
	Primary and secondary schedules . . . . .	134
	Update and repair schedules . . . . .	135
	Expiration . . . . .	135
	Blackout schedules . . . . .	136
	Varying the schedule . . . . .	137
	Policy Management control . . . . .	137
	Updating staged packages . . . . .	138
	Roaming users and multiple-user machines. . . . .	139
	User-controlled packages. . . . .	139
	Creating user-controlled packages. . . . .	140
	Package deployment using custom segmentation . . . . .	143
Chapter 9	Configuring Policy Manager . . . . .	145
	The Policy Service plug-in . . . . .	146
	Configuring and publishing the Policy Service plug-in . . . . .	146
	Starting or stopping the plug-in . . . . .	150
	Plug-in configuration page: directory service fields . . . . .	151
	Policy Plug-in configuration for Matrix42 Empirum connection settings for OS migration . . . . .	153
	Setting up access control lists . . . . .	153
	Using the access control feature . . . . .	154
	Configuring policy compliance settings . . . . .	155
	Creating and managing profiles for Windows Power Options. . . . .	156
	<b>Feature Properties</b> . . . . .	158
Chapter 10	Viewing targets and packages . . . . .	159
	Types of targets . . . . .	160

What is the All Endpoints target? . . . . .	160
What is a collection? . . . . .	161
What is an excluded target? . . . . .	162
What is a directly assigned target? . . . . .	162
What is a site? . . . . .	162
<b>Viewing targets . . . . .</b>	<b>164</b>
Browsing targets . . . . .	165
Viewing members of a target . . . . .	166
Searching for targets. . . . .	167
Viewing target details . . . . .	171
Viewing packages assigned to a target . . . . .	173
<b>What is a package? . . . . .</b>	<b>175</b>
<b>Viewing packages . . . . .</b>	<b>175</b>
Browsing packages . . . . .	176
Searching for packages. . . . .	177
Viewing targets that have been assigned a package . . . . .	177
Viewing package details . . . . .	180
<b>Using the Query Builder to search for custom packages . . . . .</b>	<b>181</b>
The Query Builder feature . . . . .	181
Configuring the Query Builder . . . . .	181
<b>Chapter 11 User Centric Deployment . . . . .</b>	<b>185</b>
<b>Introduction to User Centric Deployment . . . . .</b>	<b>186</b>
<b>Advantages of UCD . . . . .</b>	<b>186</b>
<b>A typical scenario where user centric deployment is used. . . . .</b>	<b>187</b>
<b>Workflow of UCD . . . . .</b>	<b>187</b>
Prerequisites . . . . .	187
<b>Enabling UCD on an endpoint . . . . .</b>	<b>188</b>
<b>Types of user centric deployments. . . . .</b>	<b>188</b>
Configure device level . . . . .	190
Using the file upload feature of UCD. . . . .	190
<b>Using a template for UCD . . . . .</b>	<b>191</b>
Assigning templates to channels. . . . .	193
Assigning static device level to channels . . . . .	193
Processing UCD templates at endpoint. . . . .	194

Chapter 12	Blackout Period . . . . .	199
	What is a blackout period? . . . . .	200
	Setting the blackout period for a target . . . . .	201
	Exempting packages from the blackout period. . . . .	204
	Setting blackout priorities . . . . .	204
	What are blackout priorities? . . . . .	204
	Configuring blackout priorities . . . . .	205
Chapter 13	Creating and editing policies . . . . .	207
	What is a policy? . . . . .	208
	General directions for creating and editing policies . . . . .	209
	Editing a policy for multiple packages (Edit All) . . . . .	211
	Previewing and saving policy changes . . . . .	211
	Notes about saving policies. . . . .	212
	Adding and removing packages from a policy . . . . .	213
	Adding packages to a policy . . . . .	213
	Removing packages from a policy . . . . .	214
	Specifying states and schedules for packages in a policy . . . . .	215
	Overview of installation states . . . . .	215
	Setting the primary and secondary states . . . . .	218
	Overview of schedules . . . . .	231
	Auto Inventory Scan . . . . .	236
	Setting the primary and secondary schedule for packages . . . . .	237
	Setting the update schedule for packages . . . . .	238
	Setting the repair schedule for packages . . . . .	241
	Setting common schedules for multiple packages . . . . .	243
	Conflict resolution: states and schedules in policies. . . . .	244
	Resolving differences in package states or schedules . . . . .	246
	Setting the install priority for packages in a policy . . . . .	248
	What is install priority? . . . . .	249
	Conflict resolution: Packages with the same install priority . . . . .	252
	Conflict resolution: when multiple users edit the same policy . . . . .	252
	Copying policies. . . . .	253
	Deleting policies. . . . .	254
	Managing software, data, and updates . . . . .	260
	Endpoint environment management concepts. . . . .	260

Use cases . . . . .	260
Editing policies from Package View . . . . .	264
Specifying policies for OS migration . . . . .	266
Specifying personal backup settings for OS migration . . . . .	267
Scheduling a personal backup through a policy . . . . .	268
Specifying the Policy Service schedule for a target . . . . .	270
What is the Policy Service? . . . . .	270
Setting the schedule for Policy Service updates. . . . .	271
Specifying reboot settings for Windows targets . . . . .	273
Setting Power Options for Windows targets . . . . .	273
Setting tuner and package properties for a target . . . . .	275
Overview of tuner and package policies. . . . .	275
Setting tuner properties . . . . .	285
Setting package properties . . . . .	286
Tuner and package properties format . . . . .	287
Deleting tuner and package properties . . . . .	289
Conflict resolution: property values . . . . .	291
Conflict resolution: When multiple users edit properties . . . . .	292
Chapter 14 Integration with Patch Management . . . . .	295
Prerequisites for integration with Patch Management . . . . .	296
What is a patch group? . . . . .	296
What is a patch group assignment state? . . . . .	297
What is Patch Service? . . . . .	297
Assigning patch groups to targets . . . . .	297
Removing patch groups from a policy . . . . .	301
Simulating the installation of patches . . . . .	302
Viewing more details about the installation of patch groups . . . . .	303
Overriding the Patch Service update schedule for target machines. . . . .	304
Exempting Patch Service from the blackout period . . . . .	305
Policy compliance for patch groups . . . . .	307
Specifying transmitter permissions for a target . . . . .	308
What are transmitter permissions?. . . . .	308
Adding or editing transmitter permissions . . . . .	308
Deleting transmitter permissions . . . . .	309
Specifying the profile for a target . . . . .	310

What is a profile? . . . . .	310
Changing the profile for a target. . . . .	311
Provisioning unprovisioned Intel AMT vPro computers . . . . .	313
Configuration . . . . .	313
 Chapter 15 Viewing policy compliance . . . . .	317
What is policy compliance? . . . . .	318
Definition of compliance . . . . .	318
Prerequisites for policy compliance . . . . .	321
Setting up the console server for policy compliance . . . . .	321
Setting up endpoints for policy compliance . . . . .	322
Best practices for policy compliance . . . . .	322
How does policy compliance work? . . . . .	324
Components of policy compliance. . . . .	324
Limitations . . . . .	327
Viewing policy compliance for targets and packages . . . . .	328
Overall compliance queries and compliance reports . . . . .	329
Locating targets and running queries. . . . .	329
Compliance reports . . . . .	338
OS migration compliance. . . . .	340
 Chapter 16 Integrating with Deployment Manager . . . . .	341
Prerequisites for integration with Deployment Manager . . . . .	342
Assumptions and limitations . . . . .	343
Enabling and disabling immediate policy updates . . . . .	345
Performing an immediate policy update . . . . .	345
Monitoring and viewing the status of a policy update . . . . .	347
Stopping and retrying policy updates . . . . .	348
How does the immediate policy update work?. . . . .	349
Blackout periods and immediate policy updates. . . . .	350
 Chapter 17 Matrix42 OS Migration for BMC Marimba Client Automation-Windows 351	
Architecture and components. . . . .	351
Empirum Server or Empirum Masterdepot . . . . .	352
OS packaging workstation . . . . .	352

Console server . . . . .	353
Master Transmitter . . . . .	353
Repeater Transmitter . . . . .	353
Empirum Subdepot . . . . .	353
Proxy . . . . .	353
Clients . . . . .	353
<b>Preparing your system for OS migration . . . . .</b>	<b>354</b>
Installing the Empirum Server . . . . .	354
Configuring Empirum Connection settings . . . . .	355
Configuring and deploying the Empirum Masterdepot . . . . .	355
Configuring and deploying the Empirum Subdepot . . . . .	356
Configuring the DataSync plug-in . . . . .	357
Understanding the DataSync channel workflow . . . . .	357
The OS migration Service channel . . . . .	358
Client system requirements . . . . .	358
Using the new Report Center queries for OS migration . . . . .	358
Understanding the Scanner Service enhancements for OS migration . . . . .	359
<b>OS migration workflow . . . . .</b>	<b>359</b>
Initiating OS migration . . . . .	359
OS migration actions . . . . .	361
Tuner installation after OS migration . . . . .	363
Replicating content across the Empirum Masterdepot, Master Transmitter, and Empirum Subdepot components . . . . .	364
<b>Troubleshooting using logging codes . . . . .</b>	<b>365</b>
Policy plug-in . . . . .	365
OSM Replicator channel . . . . .	366
DataSync channel plug-in . . . . .	367
Infrastructure Administration . . . . .	368
Policy Service channel . . . . .	368
Table 17-7 lists the log IDs and corresponding log messages for the Policy Service channel. . . . .	368
Infrastructure Service . . . . .	369
<b>Chapter 18 Provisioning a new operating system - Windows . . . . .</b>	<b>371</b>
Architecture of OS provisioning. . . . .	372
Prerequisites for OS provisioning . . . . .	373
OS provisioning workflow . . . . .	373

Provisioning an OS . . . . .	374
Creating a group . . . . .	374
Adding and editing a template to provision an OS . . . . .	377
Restoring a backup to a computer . . . . .	378
Self-provisioning an operating system . . . . .	379
Activating and deactivating OS provisioning for computers . . . . .	380
Activating and deactivating OS provisioning for groups . . . . .	381
Viewing the compliance report for OS provisioning tasks . . . . .	381
Editing variables . . . . .	381
<b>Section III Appendices . . . . .</b>	<b>383</b>
Appendix A Command-line reference . . . . .	385
<b>Command-line basics . . . . .</b>	<b>386</b>
Location of the runchannel program. . . . .	386
Syntax for the runchannel program . . . . .	386
Before using runchannel . . . . .	387
Running multiple command-line sessions . . . . .	387
Using runchannel with Policy Manager stopped . . . . .	387
Logging in . . . . .	388
Providing user authentication. . . . .	388
Case sensitivity . . . . .	388
Interpreting return codes . . . . .	388
Setting Tuner properties . . . . .	388
Specifying multiple command instances . . . . .	389
<b>Using runchannel options . . . . .</b>	<b>389</b>
runchannel options by alphabetical order . . . . .	390
runchannel options by function . . . . .	392
Authentication options . . . . .	392
Configuration options . . . . .	393
Policy options . . . . .	400
Policy Reporter options . . . . .	415
ACL and permission options. . . . .	416
Deprecated options. . . . .	417
Specifying schedules . . . . .	418

---

	Specifying blackout schedules. . . . .	418
	Specifying primary, secondary, update, and repair schedules. . . . .	419
Appendix B	Improving Sun ONE directory LDAP performance. . . . .	423
	Creating indexes for LDAP entries used by BMC Configuration Automation for Clients . . . . .	424
	Tuning database performance. . . . .	425
	Calculating maximum supportable cache size . . . . .	426
	Calculating maximum cache size and maximum entries in cache . . . . .	427
	Monitoring database caching performance . . . . .	428
Appendix C	Troubleshooting . . . . .	429
	Troubleshooting Policy Manager . . . . .	430
	Troubleshooting system settings . . . . .	434
Appendix D	Removing Policy Management entries from the directory service . . . . .	439
	Removing Policy Management entries from directory services . . . . .	440
Appendix E	State mappings for policy compliance . . . . .	445
	Mapping of package states for policy compliance . . . . .	446



# Preface

This manual contains configuration and administration information about the Policy Management component of BMC Marimba Client Automation. Policy Management is a highly configurable and robust application that enables you to make sure that the state of installed software and data on each machine in your organization complies with the policies of your enterprise. This manual is for the system administrator responsible for carrying out these policies.

The following topics are provided:

- Accessing product documentation (page 13)
- Related documentation (page 16)

## Accessing product documentation

You can access BMC Marimba Client Automation documentation in the following ways:

- Using the documentation channel (page 13)
- Using the Channel Store (page 15)

The most recent information is located on the Channel Store.

## Using the documentation channel

If you are using a Windows platform, you can subscribe to the BMC Marimba Client Automation Product Documentation channel. You can download the contents to a Windows computer from the Channel Store.

You can find the latest documentation channel in 8.3.00 - Current category of Channel Store.

## Searching across the BMC Marimba Client Automation documentation set

BMC Marimba Client Automation comes with a large documentation set in PDF format, which can make it difficult to quickly find exactly what you need. With Adobe Reader version 7.0 and later, you can perform a full-text search across all of your PDFs that reside in the same directory, including subdirectories.

### ► To search for a word or phrase in the PDFs contained in the BMC Marimba Client Automation Product Documentation

- 1 Ensure that you have installed the documentation channel as described in “Using the documentation channel” on page 13.
- 2 If you do not have Adobe Reader version 7.0 or later, you can download the latest Adobe Reader version from <http://www.adobe.com> for free.
- 3 Start Adobe Reader.
- 4 Click the search icon (binoculars) in the toolbar.

If the search icon is missing, try right-clicking the toolbar to find more toolbar options. Different versions of Adobe Reader put the option in different places.

- 5 After the Search window is displayed, select All PDF Documents in under **Where would you like to search?**
- 6 Click the folder selection box and choose **Browse for Location**.
- 7 Browse to the top-level folder that contains the PDF documents installed with the BMC Marimba Client Automation Product Documentation channel.
- 8 Type a search term in the **What word or phrase would you like to search for?** box and click **Search**.

The search tool searches for the term you entered in all of the PDFs in the chosen directory and its subdirectories and displays the results.

## Using the Channel Store

The BMC Marimba Client Automation documentation is located on the Channel Store.

## Related documentation

BMC provides BMC Marimba Client Automation documents in PDF format. These documents are written for system administrators and are listed in the following table.

Guide	Description
<i>BMC Marimba Client Automation Product Introduction</i>	Introduces you to BMC Marimba Client Automation and its components and defines basic concepts about its core technology.
<i>BMC Marimba Client Automation Installation Guide</i>	Provides: <ul style="list-style-type: none"><li>▀ information needed to design an infrastructure for your enterprise, which involves determining the machines you will use for the various components and whether you need to purchase additional hardware and software</li><li>▀ instructions for a first-time installation of BMC Marimba Client Automation and its associated components</li><li>▀ instructions about upgrading to the current version</li><li>▀ hardware requirements (such as processing speed, disk space, and RAM) and operating system requirements for supported platforms. This guide also lists the supported databases, directory services, and locales</li></ul>
<i>BMC Marimba Client Automation Installation Notes</i>	Lists supported platforms and system requirement.
<i>BMC Marimba Client Automation Application Packager Guide</i>	Provides information about packaging software for distribution to desktops or servers. This guide also includes information about command-line usage, policies, XML templates, and Windows system macros.
<i>BMC Marimba Client Automation CMS and Tuner Guide</i>	Provides information about the Common Management Services (CMS) and tuner infrastructure components. This guide also describes the tools and features you use to configure these components.
<i>BMC Marimba Client Automation Configuration Discovery Integration for CMDB Implementation Guide</i>	Provides instructions about planning, installing, and configuring the Configuration Discovery integration. This guide also includes information about relationship classes and mappings, data exchanges, and reconciliation definitions.
<i>Database Schema Guide</i>	Provides reference information about database schema, such as table names, field names, indexes, and primary, foreign, and unique key constraints.

Guide	Description
<i>BMC Marimba Client Automation Deployment Manager Guide</i>	Describes how to use Deployment Management and Content Replicator to control and monitor the distribution of content and applications across heterogeneous server platforms and data centers. Deployment Manager extensions to Report Center and Application Packager are also described.
<i>BMC Marimba Client Automation Device Management Guide</i>	Describes how to use Configuration Management products to manage your mobile devices. This includes Scanner Service to perform inventory scans on your mobile device endpoints; Report Center to run queries against your scanned data; Application Packager, using the PDA Packager, to package and publish files and applications to mobile devices; and Policy Service to define subscription policies for your mobile devices.
<i>BMC Marimba Client Automation Package Deployment CLI Guide</i>	Provides syntax and usage information about the command-line options used with Content Replicator, Deployment Manager, and Application Packager. Using the SOAP interface feature is also described.
<i>BMC Marimba Client Automation Patch Management Guide</i>	Helps you configure and administer Patch Management and the Patch Service plug-in. This guide also includes working with the patch repository, patches, patch groups, and custom patches, and deploying patches.
<i>BMC Marimba Client Automation Policy Management Guide</i>	Helps you configure and administer Policy Management and the Policy Service plug-in. This guide also includes integration procedures for directory services, such as Active Directory, ADAM / AD LDS, and Sun ONE Directory.
<i>BMC Marimba Client Automation Reference Guide</i>	Provides reference information, such as command-line options, tuner properties, proxy properties, transmitter properties, channel properties, channel parameters, channel states, ports, and log IDs with associated log messages.
<i>BMC Marimba Client Automation Report Center Guide</i>	Provides instructions about running queries of inventory information, configuring the Inventory and Logging plug-in, configuring endpoints, and integrating Report Center with other Configuration Management applications.
<i>BMC Marimba Client Automation Transmitter and Proxy Guide</i>	Provides information about the transmitters and proxy infrastructure components. This guide also describes the tools and features you use to configure these components.
<i>Definitive Software Library Administrator's Guide</i>	Provides a description of the Definitive Software Library and explains how the DSL is useful to you, how to use the DSL console, and how to access the DSL using Configuration Management products, such as Report Center and Application Packager.



Section

# I Getting started

Part 1 discusses the following topics:

- “Overview” on page 21
- “Initial configuration” on page 27
- “Integrating directory services” on page 33
- “Setting up user accounts” on page 65
- “Setting up collections” on page 81



# 1 Overview

Policy Management provides centralized control for distributing applications and data across your organization. As a BMC Marimba Client Automation administrator, you have centralized control of how the product distributes packaged data and applications across your enterprise. You determine which package the product distributes to which end user, when, and how the product installs the package.

You can use Policy Manager in conjunction with the Inventory module to distribute a package to specific subsets of end users whose computers exhibit certain characteristics, such as installed memory or CPU type.

The following topics are provided:

- Policy Management components (page 21)
- Policy Management system architecture (page 22)
- Workflow for using policies to update endpoints (page 24)
- Basic workflow (page 25)

## Policy Management components

In broad terms, you can use BMC Policy Management to control the distribution of applications and data across your organization by creating policies that describe which packages to deliver to which targets, when to deliver the packages, and how to install the packages.

Policy Management consists of the following components:

- **Policy Manager**—The central management application for Policy Management. This application, through its GUI and command line, is your primary interface with Policy Management. Many administrators can log in simultaneously to a single instance of Policy Manager through its browser-based interface. Only one instance of Policy Manager is necessary and the system expects only one instance to be installed. Note the contrast to earlier versions, in which an instance was required for each logged administrator.
- **Policy Service**—The client software that resides on each endpoint managed by Policy Manager. This software, known as a *service*, is implemented as a BMC Marimba Client Automation channel and has no graphical user interface (GUI). Policy Service is responsible for applying the policies assigned to the endpoint on which it is running.
- **Policy Service plug-in**—A transmitter extension that applies to Policy Service. The plug-in implements the middle-tier portion of Policy Management and is responsible for obtaining the policy for each endpoint. The plug-in determines the target groups, if any, to which an endpoint belongs.
- **Common Management Services (CMS)**—A BMC Marimba Client Automation channel that provides the interface through which you configure system settings. CMS provides services to Web-based BMC Marimba Client Automation applications that you can switch between.

---

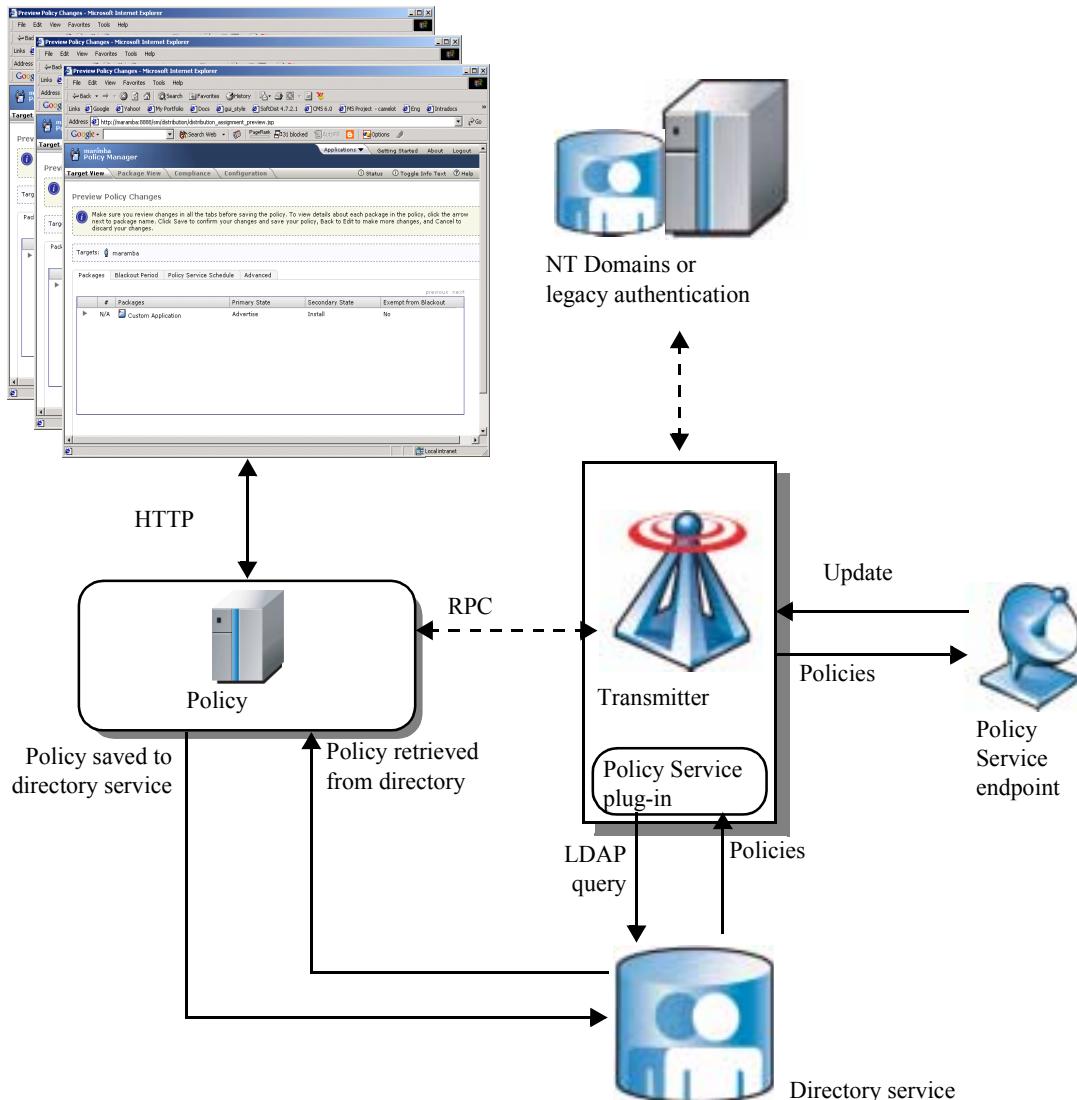
**WARNING:** Each time you log in to the CMS Console, you begin a *browser session*. Do not specify Web pages by typing in a URL, following bookmarks, or using the browser's Back and Forward buttons during a session. Many Web pages associated with a session depend on your having visited previous pages that set variables in the session.

---

## Policy Management system architecture

Figure 1-1 illustrates a Policy Manager session in which the CMS Console communicates with Policy Manager through HTTP. Using HTTP, you can manage policies from any computer that can communicate with the host on which Policy Manager is installed. You can also work with Policy Manager through a command-line interface.

Figure 1-1: Policy Management system architecture (simplified network diagram)



You can create and edit policies from the CMS Console in a Policy Manager session. Policy Manager stores the policies in a directory service. If another administrator edits and saves the plan you have been editing, Policy Manager merges that plan with yours and presents the results before saving your edits.

After you save a policy, the transmitter is prepared to serve endpoints that request policy updates.

# Workflow for using policies to update endpoints

Each endpoint contains Policy Service software running on a computer on the network connected through the Internet. The endpoint is configured to request updates periodically from the transmitter, identifying itself with the logged-in user and machine name of the computer where it is running.

The transmitter looks up the logged-in endpoint user and machine name in the directory service or through a legacy authentication system such as NT Domains, as indicated by the dashed arrows in Figure 1-1 on page 23.

---

Note: Using a legacy authentication system is possible only if you obtain users and user groups from the transmitter.

---

The Policy Management plug-in then queries the directory service to determine which policies apply to the endpoint. At this point, the plug-in resolves any conflicting package states that may result when more than one policy has been specified for the endpoint, usually as a result of the endpoint being a member of different target groups. The plug-in makes no distinction between policies that target the endpoint as part of a group and those that target the endpoint individually—all policies are treated equally and only the conflicting package states are resolved according to precedence rules. Finally, the transmitter delivers the policies, together with endpoint property settings and other information, to the endpoint.

When the Policy Service plug-in determines the policies for an individual endpoint, it must account for all groups of which the endpoint is a member. The endpoint can be part of several targeted groups, each with its own policy. Any conflicts in schedule, installation state, and priority, must be resolved before the endpoint policies are delivered. Policy Manager shows these conflicts while you are creating a policy. To see how these conflicts are resolved before the policies are actually delivered to an endpoint, you can use the Policy Reporter tool to perform a runtime test.

Finally, Policy Service enforces the policies on the endpoint, locating, downloading, and installing packages on a schedule dictated by the policies.

# Basic workflow

In broad terms, you can use BMC Policy Management to control the distribution of applications and data across your organization by creating policies that describe which packages to deliver to which targets, when to deliver the packages, and how to install the packages.

## ► To use Policy Management to control distribution of change

- 1 Target—Decide which tuner endpoints will receive your distribution. A target represents one or more computer endpoints.
- 2 Packages—Decide what the targets should receive. Your applications and data are first bundled into packages and then published to one or more servers, called transmitters.
- 3 Policy—Using Policy Manager software, determine a policy for each target and save the policy to a directory service. The policy describes which packages to deliver to the targets, when they will be delivered, and how they will be installed, as follows:
  - State—Decide how the distribution will happen, such as automatic or user-manipulatable.
  - Schedule—Decide when the distribution will happen by setting a schedule.
- 4 Preview—Confirm that the policy is well-formed.
  - On each endpoint, client software communicates with the transmitter, identifying the endpoint user and machine name.
  - The transmitter consults the policies stored on the directory service, determines the policies for the endpoint, and sends them to the client software.
  - The client software at the endpoint enforces the policies by downloading and installing packages as specified in the policies.
- 5 Verify—Confirm that the distribution was successful.



Chapter

# 2

# Initial configuration

This section provides the information you need to configure BMC Policy Management. Before you can perform the tasks described in this chapter, you must download your BMC Marimba Client Automation components, set up your master transmitter and console server, and install Policy Manager. You can find instructions for the installation process in the *BMC Marimba Client Automation Installation Guide*. See the section on Setting up Policy Management. Also, be sure to read the Policy Management section of the release notes before beginning the installation process. Both documents are available on the BMC Customer Support website.

If you are upgrading from an earlier version of Policy Management instead of performing a fresh new installation), refer to the upgrade section of the *BMC Marimba Client Automation Installation Guide*.

The following topics are provided:

- Editing attributes of a configuration object (optional) (page 28)
- Setting directory services for repeaters (page 28)
- Restarting Policy Manager (page 31)

## Editing attributes of a configuration object (optional)

If you decide to change the name or location of containers used by the Policy Management module, you must edit one or more attributes of the Subscription or BMC Marimba Client Automation configuration object. “Editing attributes of a configuration object” on page 116 provides instructions for using the command-line interface to edit attributes.

After you edit attributes for the Subscription configuration object, you must restart Policy Manager and republish the Policy Service plug-in. If desired, you can postpone publishing the Policy Service plug-in until you come to the configuration part of installation and setup, which is described in “Restarting Policy Manager” on page 31.

## Setting directory services for repeaters

If you are not using repeaters (only a master transmitter and mirrors), skip this section. Your installation is finished. If you are using repeaters, however, now is a good time to set them up.

If you are using repeaters, you may generate a large amount of network traffic when each repeater contacts the directory service associated with the master transmitter.

Directory services are typically replicated across an organization to improve response and minimize network traffic. You can take advantage of replicated directory services by configuring Policy Management so that each repeater contacts a nearby directory service, eliminating the need to contact the one assigned to the master transmitter. Moreover, you can assign a list of directory services to each repeater. If one directory service in the list fails, the repeater attempts to contact the next one, eliminating single point of failure problems.

The mechanism for mapping directory services to repeaters is the *directory service mapping file*. In the mapping file, you associate each repeater name with a list of directory services identified by host, port, base DN, bind DN, and password. For more information about how to create a mapping file, see “Creating the directory service mapping file” on page 29.

## Configuring Sun Java System Directory Server (Sun One) directory replicants

The BMC Policy Management product requires schema additions to Sun Java System Directory Server Directory. If you are replicating directory services (so that they are local to repeaters), the schema changes must be applied after you configure replication. Otherwise, the schema changes will not be replicated.

If you must bring an additional replica online after you have applied the required schema changes to the master directory service, you can install the schema changes by copying the following file from the master directory service (whose schema has been extended) to the replica directory services. This file contains the BMC Marimba Client Automation schema additions:

```
C:\Sun ONE Directory\Servers\slapd-  
<host_name>\config\schema\99user.ldif
```

---

Note: The path, and possibly the name of the file, may be different, depending on your specific Sun Java System Directory Server Directory configuration. See the Sun Java System Directory Server documentation. The replica Sun Java System Directory Server Directory must be stopped and restarted after this operation.

---

## Creating the directory service mapping file

The directory service mapping file is published from Policy Manager to the master transmitter along with the Policy Service plug-in (publishing the plug-in is described in “Restarting Policy Manager” on page 31). In this section, you will create the directory service mapping file.

Use a text editor to create the directory service mapping file. Describe the mapping with a series of simple entries, such as the “Mapping file example” on page 31.

---

Note: You can use this method of failover with Sun Java System Directory Server Directory and Active Directory with no automatic discovery only; for Active Directory with automatic discover, failover is handled by the Active Directory and DNS infrastructure.

---

For each repeater you want to map, enter information in the following format:

```
<transmitter_name>,server=<server_value>
<transmitter_name>,basedn=<basedn_value>
<transmitter_name>,binddn=<binddn_value>
<transmitter_name>,password=plain:<password_value>
<transmitter_name>,usessl=<ssl_value>
<transmitter_name>,poolsize=<poolsize_value>
```

where:

*transmitter\_name*

is the machine name or IP address of the repeater, such as xxx.xx.x.xxx.

*server\_value*

is a comma-separated list of one or more *host:port* values that designate directory services, such as server1:389,server2:389. If you specify more than one *host:port* value, the list of servers is used for failover. Note that each server or repeater in the list must be configured to use the same “*basedn\_value*”, “*binddn\_value*”, “*password\_value*” and “*ssl\_value*” setting.

*basedn\_value*

is the distinguished name of a container in the directory service, such as dc=company,dc=com.

*binddn\_value*

is the distinguished name of the user. This value is used by the Policy Service plug-in to connect to the directory service.

*password\_value*

is a plain-text (unencoded) password.

**Upgrade Note:** If you are upgrading from 4.7, the base64-encoded password that is saved in the Policy Management 4.7 directory service mapping file will be recognized.

*ssl\_value*

If true, the Policy Service plug-in will try to connect the directory service in secure (SSL) mode; if false, the plug-in will connect in insecure mode.

*poolsize\_value*

An integer that specifies the pool size for the directory service. By default, the value specified when you added the directory service to CMS is used.

## Mapping file example

The following lines are an example of mapping file content. Note that equal signs (=) following the initial one must be preceded by the backslash escape character (\):

```
mytransmitter,server=server1:389  
mytransmitter,basedn=dc\=mycompany,dc\=com  
mytransmitter,binddn=uid\=curly,ou=\people,dc\=mycompany,dc\=com  
mytransmitter,password=plain:opensesame  
mytransmitter,usessl=false  
mytransmitter,poolsize=25
```

## Restarting Policy Manager

To complete the installation, you must configure Policy Manager before using it. For configuration procedures, see “Configuring Policy Manager” on page 145. Before configuring and using Policy Manager, you might need to restart it.

### ► To restart Policy Manager

- 1 Choose Applications > Console > System Settings.
- 2 On the General tab, click the Applications Manager link.

The Applications Manager page lists the applications that you have and their current status.

- 3 Select the check box that corresponds to Policy Manager and click Stop.  
The status changes from *running* to *subscribed*.
- 4 Select the check box that corresponds to Policy Manager and click Start.  
The status changes from *subscribed* to *running*.
- 5 Click OK to return to the General Settings page.

If you have created a directory service-to-repeater mapping file, you must use the Policy Manager command-line interface to configure and publish the Policy Service plug-in. See the `-publish` command in “Command-line reference” on page 385. If you have not created a directory service mapping file, you can use the Policy Manager browser-based interface to configure and publish the Policy Service plug-in, as described in “The Policy Service plug-in” on page 146.

---

Note: To change the configuration settings, you must log in as primary administrator. Part of the initial installation and configuration is to specify which users are primary administrators and which users are standard administrators. Standard administrators are prevented from accessing and changing the configuration settings for Policy Manager.

You can find out whether you are logged in as a primary administrator by placing your mouse pointer over the Status button in the upper-right corner of the console window.

Make sure you have the appropriate read/write permissions to the Subscriptions container in the directory service. See “Setting up user accounts” on page 65.

---

## Using Policy Service with a new plug-in

If you've published a Policy Service plug-in that is newer than the Policy Service on your endpoints, you might encounter unexpected behavior the first time the service contacts the plug-in. In the log files, you might see the following error message:

Policy Service plug-in older than 5.0 is not supported by this version of the service. Please upgrade the Policy Service plug-in.

This error is generated because the service, the first time it updates and runs, does not match the profile information for the newer plug-in. Profile information, such as the version string, is not written until after the update. The next time that Policy Service updates and runs, you will no longer see this error and unexpected behavior.

Chapter

# 3

# Integrating directory services

Policy Management relies on a directory service to store policies and target information, and supports third-party directory services such as Active Directory, ADAM / AD LDS, or Sun Java System Directory Server (Sun One) Directory.

The following topics are provided:

- What are directory services? (page 34)
- Integrating with Active Directory and ADAM / AD LDS (page 34)
- Integrating with Sun Java System Directory Server (Sun One) (page 52)
- Extending the LDAP schema (page 55)

## What are directory services?

Directory services are repositories for data about targets such as applications, files, users, printers, and so on. Policy Management requires a directory service to store policies and target data, and supports third-party directory services such as Active Directory, ADAM / AD LDS, and Sun Java System Directory Server (Sun One) Directory.

See the Policy Management section of the *BMC Configuration Automation for Clients Release Notes* for system requirements, available on the BMC Customer Support website.

Using a directory service provides the following advantages:

- Makes use of existing directory service infrastructure.
- Accommodates a large number of endpoints.
- Provides disaster recovery from backed up and replicated directory services.
- Accommodates multiple administrators working on policies simultaneously.

## Integrating with Active Directory and ADAM / AD LDS

Because Policy Manager integration with Active Directory and Active Directory Application Mode (ADAM) / Active Directory Lightweight Directory Structure (AD LDS) are very similar, this section discusses them together. Where the integration differs or applies only to Active Directory or ADAM / AD LDS, that information is identified as Active Directory only or ADAM only.

### Active Directory and ADAM / AD LDS overview

This section describes how Policy Manager integrates with an Active Directory infrastructure, including required schema changes, security considerations, and support for multi-domain topologies. In addition, it describes the effect on, and the integration points to, the Active Directory Global Catalog.

Active Directory is part of the post Windows NT 4.0 network architecture. It improves on the domain architecture of the Windows NT 4.0 operating system to provide a directory service designed for distributed networking environments. Active Directory Services in Windows 2000 provide a focal point for managing and securing Windows user accounts, clients, servers, and applications. In addition, Active Directory is designed to integrate with existing systems, applications, and devices to provide a single storage mechanism and a method for managing an entire network infrastructure.

Active Directory operates as a Network Operating System (NOS) Directory, in contrast to a standalone directory service such as the Sun Java System Directory Server (Sun One) Directory. It is rare that an organization would implement an Active Directory infrastructure dedicated only to a software distribution project. However, such a scenario is common for Sun ONE Directory implementations. For this reason, systems administrators and networking groups must understand the consequences to their shared Active Directory infrastructure of deploying Policy Manager.

Active Directory Application Mode (ADAM) / AD LDS is a Lightweight Directory Access Protocol (LDAP) directory service that runs as a user service, rather than as a system service. Because ADAM / AD LDS runs as a non-operating system service, it does not require deployment on a domain controller. Running as a non-operating system service means that multiple instances of ADAM / AD LDS can run concurrently on a single server, with each instance being independently configurable.

---

Note: You can use ADAM / AD LDS with Policy Manager if you need a flexible, standalone directory service but you do not want the network infrastructure requirements of Active Directory. If you want to use ADAM / AD LDS because you do not want to incorporate Policy Management into the enterprise's Active Directory schema, then you must copy and synchronize all machine and user information for targeting policies from the enterprise Active Directory to ADAM / AD LDS regularly. For more information, See "Synchronizing Active Directory to ADAM / AD LDS" on page 43.

---

## Integrating with Active Directory

Directories implement what is referred to as a *directory-enabled computing model*. Systems that use relational databases (RDBMS) to store persistent data typically rely on a single database instance in a known location.

Additionally, systems that use an RDBMS typically assume data access for reading, writing, and performing complex queries in approximately equal ratios.

The directory-enabled computing model assumes a network of highly available replicated directories with clients that perform rapid queries and seldom write data. This model fits perfectly as the storage mechanism for Policy Manager.

Microsoft Windows 2003/2008 Active Directory and Windows 2003 ADAM / Windows 2008 AD LDS also follow the directory-enabled computing model. By using Active Directory, Policy Manager leverages an already existing and managed directory infrastructure and does not need to implement its own.

---

Note: The Global Catalog is not browsable directly by Policy Manager. Do not configure Policy Manager to use it through the Directory Services page (Applications > Console > System Settings > Data Source > Directory Services).

---

Common Management Services (CMS) automatically discovers the Global Catalog for the domain in which it is running. It uses the Global Catalog to authenticate universal, global, and local groups in the domain. CMS passes the discovered Global Catalog to Policy Manager for browsing all groups. CMS also automatically discovers the domain controller for the domain in which it is running. It passes the domain controller information to Policy Manager for storing policies. See “Automatic discovery of the global catalog” on page 49.

## When to use automatic discovery for Active Directory

If you choose Active Directory for the directory type, you have the option of using automatic discovery to find the Active Directory domain, site, Global Catalog, and domain controller using DNS information. If you choose to use automatic discovery and you are using Policy Manager, automatic discovery will be used also when connecting to Active Directory from Policy Manager and the Policy Service plug-in.

### When to use automatic discovery

Using automatic discovery has the following advantages:

- The console, Policy Manager, and the Policy Service plug-in connect to the domain controller and Global Catalog from the same site as the machine where they are running.
- You automatically get load balancing for the domain controllers in a particular site.
- Users from other domains will be able to log in to the console and use applications (if they are given the appropriate role and access).
- When using Policy Manager, users can assign policies to targets across domains.
- When using Policy Manager, users do not need to create an `ldapservers.txt` file to map the Policy Service plug-in to the closest Active Directory server.

One disadvantage when using automatic discovery and Policy Manager is that policies have to replicate to the Global Catalogs of the plug-in.

Depending on how Active Directory is configured in your enterprise, this replication might take time to occur and cause a delay before policies become available to endpoints in remote sites.

### When to avoid automatic discovery

Do not use automatic discovery when connecting to Active Directory in the following cases:

**Administration tools.** Do not use automatic discovery if you are using Active Directory as the source for users that you want to give remote administration access for the administration tools (such as Tuner Administrator and Transmitter Administrator). The administration tools require that you explicitly enter the host name, port number, and base DN for Active Directory.

**Policy Manager and user authentication.** When using Active Directory with Policy Manager and for user authentication when logging in to the console, *not* using automatic discovery has the following advantages:

- When DNS is not set up with Active Directory information (for example, in a test environment or during a demonstration), you can configure the console and Policy Manager to connect directly to an Active Directory server.
- If administrators do not want the plug-in to go to the Global Catalog closest to it and they do not mind the network traffic that might be generated, they can configure the plug-in so that it goes directly to the same Active Directory server being used by Policy Manager. Forcing the plug-in to go to the same Active Directory server as Policy Manager allows policy changes to immediately without having to wait for Global Catalog replication to take place.

## Schema modifications to Active Directory and ADAM / AD LDS

In order to support Policy Manager, the schema of the Active Directory infrastructure must be extended. Several new objects must be added to the schema. These objects are used to represent policies and the configuration of Policy Manager and Policy Service plug-in. A series of containers also must be created in default locations to store these new objects. They are added when you run the LDIF scripts during installation.

**Schema changes registered with Microsoft.** The changes that the LDIF scripts make to extend the Active Directory schema are registered with Microsoft. The LDIF scripts generated by Policy Manager use a unique object identification number (OID) and prefix to make sure that there will be no conflicts with schema changes made by other vendors' products:

- OID: 1.3.6.1.4.1.3403.1.1.1
- Schema naming prefix: marimbaCom1996

The LDIF scripts also use the naming format recommended by Microsoft for attributes and classes.

**List of classes and attributes.** The following two tables list the classes and attributes that must be created for Policy Manager, in the order in which they appear in the LDIF scripts generated by Policy Manager.

Table 3-1 lists the classes that must be created for Policy Manager.

Table 3-1: Active directory and ADAM / AD LDS classes

Class name
marimbaCom1996-Castanet-SubscriptionSubscription
marimbaCom1996-Castanet-SubscriptionCollection
marimbaCom1996-Castanet-MarimbaACL (added in 6.0)
marimbaCom1996-Castanet-MarimbaProperties (added in 6.0)
marimbaCom1996-Castanet-MarimbaComputer (added in 6.0; ADAM / AD LDS only)

Table 3-2 lists the attributes that have indexes created or that are replicated in the Global Catalog. For more information about the Global Catalog, see “The global catalog” on page 48.

Table 3-2: Active directory and ADAM / AD LDS object attributes

Attribute name	Syntax	Multi-valued	Indexed	GC	Since version
marimbaCom1996-Castanet-SubscriptionConfig	Case-ignored string	Yes	No	Yes	4.7
marimbaCom1996-Castanet-SubscriptionLastUpdated	Integer	No	No	No	4.7
marimbaCom1996-Castanet-SubscriptionSQL	Case-ignored string	No	No	No	4.7
marimbaCom1996-Castanet-SubscriptionSQLCondition	Case-ignored string	Yes	No	No	4.7
marimbaCom1996-Castanet-SubscriptionTargetAll	Case-ignored string	No	Yes	Yes	4.7
marimbaCom1996-Castanet-SubscriptionTargetTxGroup	Case-ignored string	Yes	Yes	Yes	4.7
marimbaCom1996-Castanet-SubscriptionTargetTxUser					

Table 3-2: Active directory and ADAM / AD LDS object attributes (Continued)

Attribute name	Syntax	Multi-valued	Indexed	GC	Since version
	Case-ignored string	Yes	Yes	Yes	4.7
marimbaCom1996-Castanet-SubscriptionChannel					
	Case-ignored string	Yes	No	Yes	4.7
marimbaCom1996-Castanet-SubscriptionCreateFlag					
	Case-ignored string	No	No	No	4.7
marimbaCom1996-Castanet-SubscriptionChannelOrder					
	Case-ignored string	Yes	No	Yes	5.0
marimbaCom1996-Castanet-SubscriptionChannelSecondary					
	Case-ignored string	Yes	No	Yes	5.0
marimbaCom1996-Castanet-SubscriptionChannelInitSched					
	Case-ignored string	Yes	No	Yes	5.0
marimbaCom1996-Castanet-SubscriptionChannelSecSched					
	Case-ignored string	Yes	No	Yes	5.0
marimbaCom1996-Castanet-SubscriptionChannelUpdateSched					
	Case-ignored string	Yes	No	Yes	5.0
marimbaCom1996-Castanet-SubscriptionChannelVerRepairSched					
	Case-ignored string	Yes	No	Yes	5.0
marimbaCom1996-Castanet-SubscriptionBlackoutSched					
	Case-ignored string	Yes	No	Yes	5.0
marimbaCom1996-Castanet-SubscriptionTargetType					
	Case-ignored string	No	No	Yes	5.0
marimbaCom1996-Castanet-SubscriptionChannelTitle					
	Case-ignored string	Yes	No	No	5.0
marimbaCom1996-Castanet-SubscriptionTargetDN2					
	Distinguished name	Yes	Yes	Yes	5.0
marimbaCom1996-Castanet-SubscriptionChannelExemptBlackout					
	Case-ignored string	Yes	No	Yes	6.0

Table 3-2: Active directory and ADAM / AD LDS object attributes (Continued)

Attribute name	Syntax	Multi-valued	Indexed	GC	Since version
<b>marimbaCom1996-Castanet-MarimbaConfig</b>					
	Case-ignored string	Yes	No	Yes	6.0
<b>marimbaCom1996-Castanet-ACLResourceDN</b>					
	Distinguished name	No	Yes	Yes	6.0
<b>marimbaCom1996-Castanet-ACLVersion</b>					
	Integer	No	No	No	6.0
<b>marimbaCom1996-Castanet-ACLPrincipal</b>					
	Case-ignored string	Yes	No	No	6.0
<b>marimbaCom1996-Castanet-ACLEntry</b>					
	Case-ignored string	Yes	No	No	6.0
<b>marimbaCom1996-Castanet-ACLPrincipalType</b>					
	Case-ignored string	No	No	No	6.0
<b>marimbaCom1996-Castanet-ACLPermission</b>					
	Case-ignored string	Yes	No	No	6.0
<b>marimbaCom1996-Castanet-ACLResource</b>					
	Case-ignored string	Yes	No	No	6.0
<b>marimbaCom1996-Castanet-ACLPrincipalDN</b>					
	Distinguished name	Yes	Yes	Yes	6.0
<b>marimbaCom1996-Castanet-SubscriptionPackageName</b>					
	Case-ignored string	Yes	No	No	6.0
<b>marimbaCom1996-Castanet-SubscriptionPackageGroupMemberOf</b>					
	Distinguished name	Yes	Yes	Yes	6.0 (for use in future versions)
<b>MarimbaCom1996-Castanet-Subscription-ARReferenceTag</b>					
	Case Ignored String	Yes	Yes	Yes	7.0

---

Note: The `marimbaCom1996-Castanet-SubscriptionTargetDN2` attribute supersedes the `marimbaCom1996-Castanet-SubscriptionTargetDN` attribute used in releases of Policy Manager earlier than 5.x. The `marimbaCom1996-Castanet-SubscriptionTargetDN` attribute contained an incorrect matching rule for a distinguished name (DN) based value and has, therefore, been made obsolete.

---

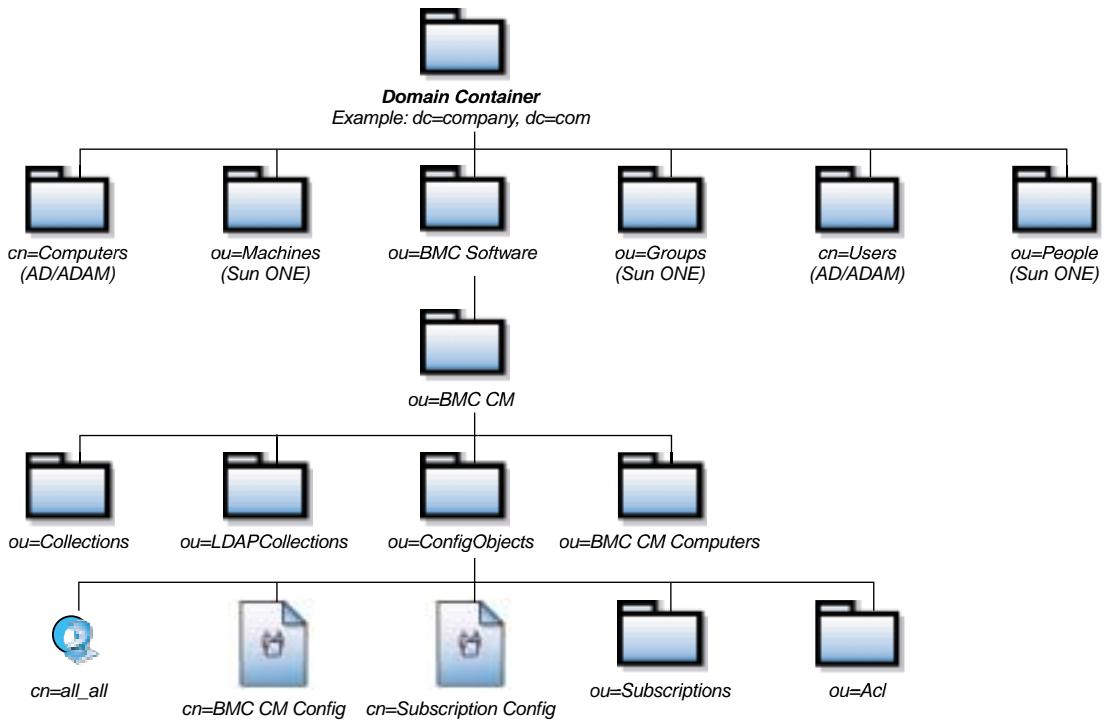
In addition to the schema entries, the installation LDIF script creates your LDAP container structure, including the following containers:

Table 3-3: Active directory and ADAM / AD LDS containers

Container name	Purpose
<code>ou=ConfigObjects</code>	The ConfigObjects container stores the <code>all_all</code> object, subscriptions, ACL objects, and configuration objects. For information about configuration objects, see “The subscription configuration object” on page 111.
<code>ou=Acl</code>	Container for storing ACLs and permissions entries.
<code>ou=Subscriptions</code>	Container for storing Policy Manager entries.

The arrangement of these containers is shown in Figure 3-1 on page 43. These locations are the defaults created by the installation script. You can change the locations, with the single exception of the Subscription configuration object. See “The subscription configuration object” on page 111.

Figure 3-1: LDAP container structure



## Synchronizing Active Directory to ADAM / AD LDS

If you do not want to incorporate Policy Management into your enterprise's Active Directory schema, you can use Active Directory Application Mode (ADAM) // AD LDS with Policy Manager for a flexible and stand-alone directory service.

To apply this feature, the computer and the user information should be copied and synchronized for targeting policies from the enterprise Active Directory to ADAM // AD LDS periodically. You can copy and synchronize the computers and user information using the Active Directory to ADAM Synchronizer, `adamsync`.

Before using this feature, ensure that you meet the following prerequisites:

- Delete computer entries in “BMC Configuration Management Computers,” if any.

- Delete policies directly assigned to the computers that have been synced, if any. This avoids an error in displaying synced computers as users in compliance screens.
- The AD-ADAM Sync task should be scheduled before collections run on its schedule to avoid duplicate entries.
- The AD-ADAM Sync task should be configured to execute when ADAM is idle and not being used by any of the BMC Configuration Automation for Clients components.
- AD and ADAM / AD LDS should be synced in diff Mode because it places too much load on the LDAP sync process and causes it to fail if the sync happens in full mode every time.

### ► To use ADAM / AD LDS with Policy Manager

- 1 Synchronize ADAM / AD LDS with Active Directory by using the Active Directory to ADAM Synchronizer tool from Microsoft.
- 2 Use the following command to modify the marimba.ldap.machineclass computer class property to include the additional computer objectClass:  
`runchannel <Policy_Manager_URL> -user <user_name> -password <password> -configSet marimba.ldap/browse.machineclass marimbacom1996-castanetmarimbacomputer,computer.`

For example:

```
runchannel http://10.10.51.28:5282/ant/nrao/SubscriptionManager -  
usernrao -password nraopassword -configSet  
marimba.ldap/browse.machineclass marimbacom1996-castanet-  
marimbacomputer,computer
```

## Installation considerations for Active Directory

This section contains information about the scripts that you use during the installation process to prepare Active Directory for use with Policy Management. It includes the following sections:

- “Installation script considerations” on page 44.

### Installation script considerations

The case in which Policy Management runs from a child domain requires an additional step when installing the installation scripts.

By default, the Active Directory installation script assumes that the domain controller that will contain the Policy Management-specific containers and policy information is also the domain schema master. As a result, one script is generated. This script contains a base DN equivalent to the domain that operates as the schema master. The single script works well for the single domain model and the model in which Policy Management runs from a root domain in a multidomain environment.

For domain environments in which the schema master is in a domain different from the one where Policy Management runs, two scripts are needed: one for the schema definitions and one to create the default containers specific to Policy Management. In previous releases, the two scripts were produced by manually splitting the generated script. BMC Policy Management allows the schema extension script and the container generation script to be generated separately, with distinct base distinguished names (domains).

The separate scripts are generated from the command line using the `-installscript`, `-schemafile`, and `-schemabasedn` options. See “Initial configuration” on page 27 and “Command-line reference” on page 385.

## Targeting Active Directory groups

Administrators who use Policy Manager typically target groups, rather than individual users and computers, when assigning policies. Active Directory accommodates different types of groups. It is important to understand each group’s use, as related to Policy Manager.

Active Directory supports two kinds of groups: *distribution groups*, which are used primarily for e-mail distribution lists, and *security groups*, which are used to manage access control. You can target both types of groups using Policy Manager.

The group scope determines whether the group’s members are included in the Global Catalog. The purpose of group scope in Active Directory is to limit Global Catalog replication to only a subset of groups and to avoid replication network traffic when members are changed for groups that are local to a single domain.

Policy Manager allows you to target many types of groups and at different levels of scope. During the group resolution of a target, Policy Manager connects to the domain of the group if it is a domain local or global group. Expect some delay in the GUI when the domain of the group being browsed is geographically far from the domain where Policy Manager is running.

Collections generated using Report Center are created as distribution, domain local groups. A domain local group and its membership list are only visible in the local Global Catalog. However, the group will not be replicated to Global Catalogs in other domains. This is why Report Center and Policy Manager must use the same domain. For detailed information about collections, see “Setting up collections” on page 81.

## Forest targeting and browsing

This section contains information about targeting and browsing in a forest environment:

- “Required connections” on page 46
- “Policy storage” on page 46
- “The all endpoints target” on page 48
- “Targeting domains” on page 48

### Required connections

To support browsing a forest environment, Policy Management requires the following connections to the directory service:

- **Global catalog connection.** Policy Manager connects to the Global Catalog to browse domains, organizational units, containers, collections, and universal groups.
- **Policy Management domain controller connection.** Policy Manager connects to the domain controller to write the policies and to resolve policies.
- **Group’s domain controller connection.** Policy Manager connects automatically to the target group’s domain controller to resolve the members of domain local and global groups, including the browsing of collections. For more information about collections, see “Setting up collections” on page 81.

### Policy storage

Policy Manager stores policies in one domain, in the Subscriptions container. The container was created by running the LDIF script during installation. See the *BMC Marimba Client Automation Installation Guide*, available on the BMC Customer Support website.

Every Policy Manager in the forest environment obtains the domain information from the Subscription configuration object. The attribute in the Subscription configuration object that stores the domain information is `marimba.subscriptionplugin.subscriptionbase`. The value is a distinguished name, such as `ou=Subscriptions,ou=ConfigObjects,ou=BMC CM, ou=BMC Software,dc=company,dc=com`.

## Group resolution and the Policy Service plug-in

When resolving policies for endpoints that belong to a group, the Policy Service plug-in resolves the policies by connecting to the directory service. For Active Directory, the plug-in resolves an endpoint's membership in the following types of groups:

- Universal groups
- Domain local and global groups in the target user's and target machine's domain. (This is equivalent to using collections in distributed mode.)
- Domain local groups from the collections domain (in centralized mode).

---

Note: The Policy Service plug-in cannot resolve group membership for endpoints that belong to a domain local group in any domain besides the target's domain and the collections domain.

---

### ► Policy Service plug-in workflow for resolving groups in Active Directory

- 1 The Policy Service plug-in connects to any Global Catalogs (GCs) in the plug-in's site. This connection allows the plug-in to obtain information about the universal group membership that contains the targets.
- 2 The plug-in connects to the domain controller (DC) to which the machine and the user (representing the target) belongs. This connection allows the plug-in to obtain information about the global and domain local group membership in the target's domain.
- 3 If using collections in centralized mode, the plug-in connects to the GC of the collections domain to resolve collections membership.

## The all endpoints target

In a forest environment, the All Endpoints target represents all targets (users or machines) that have a Policy Service configured to the LDAP in which the machine resides. Accounts not in the domain logging in to computers not in the domain receive policies you assign to the All Endpoints target. There is only one All Endpoints target in the LDAP container structure.

## Targeting domains

Policy Manager allows you to target domains. When you create a policy for a parent domain, it affects all machines and users in the child domains.

## The global catalog

The Global Catalog contains a replica of all objects in its own domain and a partial replica of objects in other domains. The Global Catalog is primarily used for fast searches across all domains in the forest. It is used by the Policy Service plug-in and is accessed using port 3268.

There are several important points to consider when using the Global Catalog:

- You must use the Global Catalog if your architecture uses multiple domains.
- Policies are replicated to the Global Catalog. Resolution of policies is asynchronous—plug-ins will only resolve the policies when the Global Catalogs replicate.
- Replication between Global Catalogs is subject to a delay. The delay is determined by the replication interval set by the Active Directory administrator. The lower limit is 15 minutes. Therefore, you cannot expect immediate results when running Policy Manager on the Global Catalog.
- All domain controllers and Global Catalogs within the same domain contain exactly the same information. However, only certain objects are replicated to Global Catalogs in other domains and sometimes only partially replicated. For example, domain local groups are not replicated at all to Global Catalogs in other domains. Global groups are replicated, but their membership lists are not, resulting in a partial replication. Universal groups, on the other hand, are fully replicated.

- Installing the schema extensions will cause a complete replication of the Global Catalog. Care should be taken to avoid swamping network links when performing this operation.

## Automatic discovery of the global catalog

In previous releases, Policy Management relied on the administrator to explicitly enter the appropriate information for Active Directory and the domain before making any connections. However, this approach is error-prone and difficult in complex environments. You have the option of having Common Management Services (CMS), Policy Manager, and the Policy Service plug-in all automatically discover the Global Catalog, domain controllers, and the current domain.

If you configure CMS to use automatic discovery when adding Active Directory to the directory services list, CMS automatically discovers the Global Catalog for the domain in which it is running. It uses the Global Catalog to authenticate universal, global, and local groups in the domain. CMS passes the discovered Global Catalog to the Policy Manager for browsing all groups. CMS also discovers the domain controller for the domain in which it is running. It passes the domain controller information to the Policy Manager for storing policies.

CMS uses the service records (SRV) in the DNS server to discover the domain. These SRV records are registered by Active Directory domain controllers and Global Catalogs when initiating connections. It also uses properties in the configuration partition to find the closest available Global Catalog. It also attempts to connect to a domain controller in the domain with the lowest load.

For UNIX machines, use the `marimba.ldap.admanagementdomain` tuner property to specify the domain, such as

`marimba.ldap.admanagementdomain=company.com`. This property must be set on both the machine where Policy Manager and CMS are running and the endpoint machines where Policy Service is running.

For more information, see

“`marimba.ldap.admanagementdomain=<domainOfCollectionsContainer>`” on page 93.

## Viewing a large number of groups

By default, Active Directory returns only 1000 group names for each query. You can change the defaults in Active Directory and Policy Manager to view larger numbers of groups. To change the maximum number of group names returned by Active Directory, you must change the MaxPageSize property. To change the maximum number of groups that can be displayed in Policy Manager, you must change the `groupdisplaylimit` property in Policy Manager.

### Setting the maximum number of groups per query in Active Directory

To change the maximum number of group names returned by Active Directory, you must change the MaxPageSize property. This also enables you to see all the users and groups when using Transmitter Administrator.

#### ► To set the MaxPageSize value in Active Directory

- 1 Choose one of the following options:
  - If you have not installed ADSI Edit, proceed to step 2.
  - If you have installed ADSI Edit, proceed to step 5.
- 2 From the Windows Start menu, select Run. Type `mmc` and click OK to run the `mmc` command.
- 3 In the console window, check the File menu and choose Add/Remove Snap-in.
- 4 Select the ADSI Edit snap-in and click Add.

In Windows 2000, if ADSI Edit is not shown in the list of available snap-ins, you must access it from the Windows 2000 CD-ROM:

- a Insert the CD-ROM and open it in Windows.
- b In the Tools folder on the CD-ROM, open `2000rkst.msi`, which will install the snap-in.
- 5 In ADSI Edit, right-click and choose Connect to.
- 6 For the Naming Context field, choose Configuration Container, and click OK.
- 7 Navigate to CN=Services, CN=Windows NT, CN=Directory Services, CN=Query Policies, CN=Default Query Policy.

- 8 Select CN=Default Query Policy, right-click, and choose Properties.
- 9 Select Both for properties to view.
- 10 Select LDAPAdminLimits for property to view.
- 11 In the Values box, search for MaxPageSize. Select it and click Remove.
- 12 After you click Remove, the value for MaxPageSize will move to the Edit Attribute field. Change the value to the number you want, such as 5000. Click Add, Apply, and OK.

## Setting the maximum number of groups displayed in Policy Manager

To change the maximum number of groups that Policy Manager can display, you must change the `groupdisplaylimit` property in Policy Manager.

### ► To set the `subscriptionmanager.groupdisplaylimit` property for Policy Manager

- 1 Using Workspace Explorer (part of the BMC Marimba Client Automation Resource Kit) or a similar tool, locate the `application.txt` file for Policy Manager in the tuner workspace on the computer where Policy Manager is installed. Add the following property to the `application.txt` file:

```
subscriptionmanager.groupdisplaylimit=<value>
```

where:

`value`

is an integer representing the upper limit to the number of groups displayed in the Policy Manager GUI. The default value for this setting is 1000.

- 2 Save and close.
- 3 Stop and restart Policy Manager.

**Note:** Windows 2000 Active Directory limits the number of targets in a group. The workaround is to create sub-groups within Active Directory. From Windows 2003 forward, Active Directory sets no limit on the number of targets in a group.

# Integrating with Sun Java System Directory Server (Sun One)

This section contains essential information to consider when using Policy Manager with the Sun Java System Directory Server (Sun One) Directory.

The following topics are provided:

- Namespace design (page 52)
- Extending the LDAP schema (page 55)
- Using schema extensions (page 59)
- Policy Management configuration properties (page 60)
- Using LDIF to create sample policies (page 62)
- Policy Manager naming conventions (page 64)

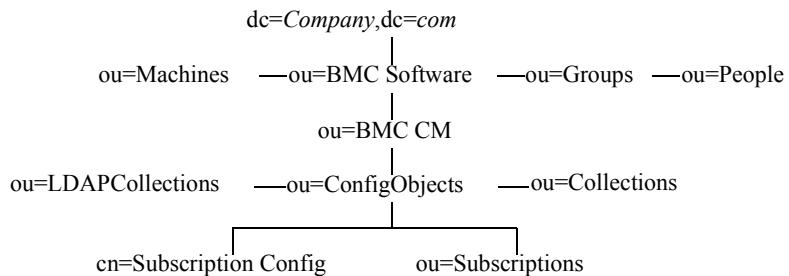
Policy Manager stores and retrieves all persistent information (user, machines, groups, collections, and policies) from a distributed directory service system, like Sun ONE Directory. All server-side logic is implemented as a transmitter plug-in, referred to as the Policy Service plug-in. Whenever the Policy Service plug-in receives a request from the Policy Service endpoint, the Policy Service plug-in obtains the policies that are relevant for that endpoint from the directory service. It then sends back a policy to the client in an XML file.

## Namespace design

Policy Manager makes no assumptions about the layout of the LDAP Directory Information Tree (DIT). You perform all queries based on optional search scopes set in the Policy Service plug-in configuration, which is itself stored in the directory service.

The following DIT is an example of a Policy Manager namespace. LDIF examples elsewhere in this chapter use this namespace.

Figure 3-2: Example directory information tree



In this example, the directory is partitioned into several organizational units, identified by distinguished names of type `ou`.

The organizational units in the example provide entry points for the following objects:

- `ou=People`—User entries.
- `ou=Groups`—Groups, typically groups of users.
- `ou=Machines`—Machine entries.
- `ou=Subscriptions`—Policy Management policy objects.
- `ou=Collections`—Collection objects, which are groups of machines based on inventory queries.
- `ou=LDAPCollections`—Machines and groups you create with LDAP query collections.
- `ou=BMC CM`—Product configurations. In this example, the configuration for the Policy Service plug-in is stored beneath this container, in the Subscription configuration object.

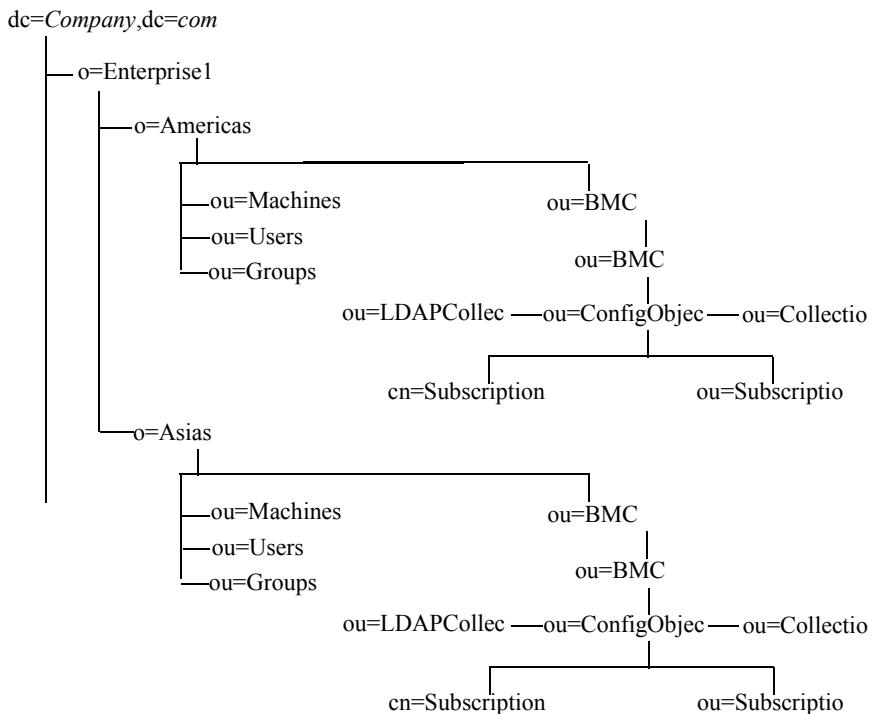
## Multiple enterprises

BMC Marimba Client Automation supports multiple enterprises from a single LDAP server. IT providers are no longer required to maintain a separate LDAP server for each enterprise. The LDAP server remains the centralized information server for each enterprise, holding the data about the enterprise and its software, users, user groups, machines, and machine collections.

Each enterprise must install its own BMC Marimba Client Automation product and plugins. You must set the location of each BMC Marimba Client Automation product under `o=<each enterprise>`.

To use a single LDAP server for multiple enterprises, you must create an additional layer in the hierarchical tree, indicated by `o=company`. Figure 5-2 shows two fictitious enterprises: Enterprise1 has organizational subunits by region, and Enterprise12 has organizational subunits by department.

Figure 3-3: Example directory information tree



## Limitations

The use of a single LDAP server for multiple organizations requires knowledge of the following limitations:

- This is not a supported option for Active Directory auto-discovery.

- You must set up separate installations of BMC Marimba Client Automation and plugins for each organization.
- You must use the `basedn` (distinguished name) of the organization, rather than `root`. If you use `root`, each company will have the ability to access the information about all the companies on your LDAP server.
- To merge company customers that are currently using separate LDAP servers, you must merge like LDAP server types. If you have some company customers using ADAM / AD LDS, for example, and others using Active Directory, you cannot migrate them all to a single LDAP server. You must migrate those using ADAM / AD LDS to one server, and those using Active Directory to another server.

## Extending the LDAP schema

Table 3-4 describes Policy Management attributes that extend the standard LDAP schema.

Table 3-4: Attributes that extend the Sun ONE directory schema

Attribute name and description	Syntax	Multi-valued	Since version
<code>mrbaConfig</code>  Used to describe tuner or package property name-value pairs as part of a policy, such as <code>marimba.appearance.backgroundcolor=255,0,0</code>	String	Yes	4.7
<code>mrbaTargetType</code>	String	No	5.0
<code>mrbaLastUpdated</code>  ( <i>Deprecated</i> <sup>1</sup> ) Indicates the last time the Collection object's inventory query was run.	Integer	No	4.7
<code>mrbaSQL</code>  ( <i>Deprecated</i> <sup>1</sup> ) Describes the SQL statement used by the Collection object to perform queries against the inventory database.	String	No	4.7
<code>mrbaSQLCondition</code>  ( <i>Deprecated</i> <sup>1</sup> ) Describes each of the conditions present in the SQL statement used by the Collection object to perform queries against the inventory database.	String	Yes	4.7

Table 3-4: Attributes that extend the Sun ONE directory schema

Attribute name and description	Syntax	Multi-valued	Since version
mrbaTargetAll  A string representation of a boolean that, when true, indicates that a Policy Management entry (representing a policy) applies to all targets (all users and machines that have Policy Service running). Legal values are true and false.	String	No	4.7
mrbaTargetDN  Defines one or more targets to which a Policy Management entry (representing a policy) applies. The target is a distinguished name of another LDAP object. Policy Manager will create policies with only one value for the target distinguished name.	Distinguished name	Yes	4.7
mrbaTargetTxGroup  Describes a user-group-based target in which the group is defined by the transmitter's source of user information—a flat file, a secondary directory service, or a transmitter authenticator extension connected to a legacy system. This attribute allows policies to be applied to user groups defined outside the directory service.	String	Yes	4.7
mrbaTargetTxUser  Describes a user-based target in which the user is defined by the transmitter's source of user information—a flat file, a secondary directory service, or a transmitter authenticator extension connected to a legacy system. This attribute allows policies to be applied to users defined outside the directory service.	String	Yes	4.7
mrbaChannel  Used to describe a package state as part of a policy.	String	Yes	4.7
mrbaChannelSecondary  Used to describe the secondary state of a package, such as http://prod:5282/Office2000=subscribe.  This is an optional attribute. Its value depends on the initial state attribute “mrbaChannel”.	String	Yes	5.0

Table 3-4: Attributes that extend the Sun ONE directory schema

Attribute name and description	Syntax	Multi-valued	Since version
mrbaChannelOrder  Used to describe a package installation order as part of a policy, such as <code>http://moe:5282/LetsEdit=2</code> , where 2 is the package installation order number for package <code>http://moe:5282/LetsEdit</code>	String	Yes	4.7
mrbaChannelTitle	String	Yes	5.0
mrbasrn  ( <i>Deprecated</i> <sup>1</sup> ) The presence of this attribute denotes that a mrbaPolicy Management entry was created using Policy Manager. Only Policy Management entries with this attribute can be seen and edited using Policy Manager.	String	Yes	4.7
mrbaChannelInitSched  Used to describe the start and end date/time for the initial state of the specified package.  For scheduling details, see “Policy Service and schedule enforcement” on page 128.	String	Yes	5.0
mrbaChannelSecSched  Used to describe the start and end date/time for the secondary state of the specified package.  This value is used only when the attribute “mrbaChannelSecondary” is defined. For the schedule format, see “Format for primary or secondary schedules” on page 422.	String	Yes	5.0
mrbaChannelUpdateSched  Used to describe the update schedule for the application. For the schedule format, see “Format for update or repair schedules” on page 422.	String	Yes	5.0
mrbaChannelVerRepairSched  Used to describe the repair schedule for the application. For the schedule format, see “Format for update or repair schedules” on page 422.	String	Yes	5.0
mrbaChannelExemptBlackout	String	Yes	6.0

Table 3-4: Attributes that extend the Sun ONE directory schema

Attribute name and description	Syntax	Multi-valued	Since version
mrbaBlackOutSched Used to describe the blackout schedule for the tuner. For the schedule format, see “Specifying blackout schedules” on page 418.	Case-sensitive string	Yes	5.0
mrbaACLResource This attribute and the ones that follow are used to describe the ACLs and permissions.	String	Yes	6.0
mrbaACLVersion	Integer	No	6.0
mrbaACLPrincipal	String	Yes	6.0
mrbaACLEntry	String	Yes	6.0
mrbaACLPPrincipalType	String	Yes	6.0
mrbaACLPermission	String	Yes	6.0
mrbaACLResourceDN	Distinguished name	Yes	6.0
mrbaACLPPrincipalDN	Distinguished name	Yes	6.0
mrbaPackageGroupDN	Distinguished name	Yes	6.0 (for use in future versions)
mrbaPackageGroupMemberOf	Distinguished name	Yes	6.0 (for use in future versions)
mrbaARReferenceTag	String	Yes	7.0

<sup>1</sup>The current version of Report Center does not create Collections with these attributes. However, the schema will allow Collection objects with these attributes to exist to accommodate upgrades from earlier versions.

## Using schema extensions

Table 3-5 defines the schema extensions specific to Policy Management. The definitions are in Standalone LDAP Daemon (`slapd`) format.

Table 3-5: Class schema extensions for Policy Management

Object class	Definition	Description
mrbaMachine	objectclass mrbamachine superior device	The <code>mrbaMachine</code> class is used to represent machines and is a subclass of the <code>device</code> class type, as defined in ISO standard X.521. The <code>cn</code> attribute of this class is used to identify the machine name.
mrbaSubscription	objectclass mrbasubscription superior top	The <code>mrbaSubscription</code> class is used to represent a policy. The policy contains a list of packages and desired states, and a list of tuner and package properties to be set at the endpoint.  The policy is assigned to one or more targets, which can be other LDAP objects or user and groups as defined by the transmitter's user/group database ( <code>IUserDirectory</code> ).
mrbaCollection	objectclass mrbacollection superior groupofnames	The <code>mrbaCollection</code> object class is used to represent machine groups whose members are defined by an SQL query against the Inventory database. Because this object class is a sub-class of the X.521 defined <code>groupOfNames</code> class it can be treated by all other LDAP clients as a simple group of machines. See "Setting up collections" on page 81 for more information.
mrbaProperties	objectclass mrbaproPERTIES superior top	The <code>mrbaProperties</code> object class is used to represent Subscription configuration object.
mrbaACL	objectclass mrbaACL superior top	The <code>mrbaACL</code> object class is used to represent ACLs and permissions for Policy Manager objects.

## Policy Management configuration properties

Policy Management components store their LDAP configuration properties as attributes of an object in the directory service. For the Sun Java System Directory Server (Sun One) Directory, the distinguished name of this object is

`cn=Subscription Config,ou=ConfigObjects,ou=BMC CM,ou=BMC Software,dc=company,dc=com`

You can modify the search bases for users, machines, and groups according to the needs of your organization.

### ► To modify configuration object attributes in Sun ONE directories

- 1 Start the Sun ONE Directory Server Console GUI.
- 2 Select the Directory tab.
- 3 Expand the Directory Root folder, such as `MyCompany.com` in the left pane.
- 4 Select the BMC Marimba Client Automation organizational unit (OU) in the left pane.
- 5 Right-click on the Subscription configuration object in the right pane and select Properties from the shortcut menu. Modify properties as required.

**Note:** If you modify any of these configuration values using this procedure, you must republish the Policy Service plug-in and restart Policy Manager.

Table 3-6 describes the properties. See Table 7-2 on page 118 for the Active Directory equivalents of these properties, where applicable.

Table 3-6: Configuration properties description

Property name	Property default value	Description
<code>marimba.subscriptionplugin.userclass</code>	<code>inetorgperson</code>	Default Class for searching user objects.
<code>marimba.subscriptionplugin.useridattr</code>	<code>uid</code>	Default Attribute used to uniquely identify user objects.
<code>marimba.subscriptionplugin.groupclass</code>	<code>groupofnames,groupofuniquenames</code>	Default Classes for searching group objects.

Table 3-6: Configuration properties description (Continued)

Property name	Property default value	Description
marimba.subscriptionplugin.groupnameattr	cn	Default Attribute used to identify group objects.
marimba.subscriptionplugin.groupmemberattr	member,uniquemember	Default Attributes used to identify group members.
marimba.subscriptionplugin.machinenameattr	cn	Default Attribute used to identify machine objects.
marimba.subscriptionplugin.mode	online	Specifies if the Policy Service plug-in is in online or offline mode. In offline mode, the plug-in does not process requests for policies from the endpoints. A log entry appears in the plug-in log indicating that the plug-in is not processing requests. Possible values are online or offline.
marimba.subscriptionplugin.subscriptionbase	ou=Subscriptions, <DIR_ROOT>	Default container where policies are stored.
marimba.ldap.browse.machineimportbase (Stored in BMC Marimba Client Automation Config object.)	ou=Machine Groups, <DIR_ROOT>	Default container where machines and machine groups are created during import from machines.txt file in the Policy Manager.

## Using LDIF to create sample policies

This section contains examples of Lightweight Directory Interchange Format (LDIF) entries used to create the following object. The given samples are applicable to Sun One Directory Server. To run the same script against Active Directory or ADAM / AD LDS, change the object class as per the recommendation. For more information, see Schema modifications to Active Directory and ADAM / AD LDS (page 38).

- “Creating a machine” on page 62
- “Creating a machine group” on page 62
- “Creating a user” on page 63
- “Creating a user and machine group” on page 63
- “Creating a group of groups” on page 63

### Creating a machine

The following LDIF entry creates a machine called larry in the Machines organizational unit.

```
dn: cn=larry, ou=Machines, o=mycompany.com
changetype: add
objectclass: device
objectclass: mrbamachine
cn: larry
ou: Machines
description: A machine called larry
```

### Creating a machine group

The following LDIF entry creates a machine group called Engineering, containing two machines.

```
dn: cn=Engineering, ou=Machine Groups, o=MyCompany.com
changetype: add
objectclass: top
objectclass: groupofnames
ou: Groups
cn: Engineering
member: cn=larry, ou=Machines, o=MyCompany.com
member: cn=acme, ou=Machines, o=MyCompany.com
```

## Creating a user

The following LDIF entry creates a user called larry. Note that the example does not set a password.

```
dn: uid=larry, ou=People, o=MyCompany.com
changetype: add
objectclass: top
objectclass: person
objectclass: organizationalPerson
objectclass: inetorgperson
uid: larry
cn: Larry Berry
sn: Berry
```

## Creating a user and machine group

The following LDIF entry creates a group called Sales, which contains both a user named dave and a machine named acme. The user and machine are created first, then the group containing them both.

```
dn: uid=dave, ou=People, o=MyCompany.com
changetype: add
objectclass: top
objectclass: person
objectclass: organizationalPerson
objectclass: inetorgperson
uid: dave
cn: Dave Smith
sn: Smith

dn: cn=acme, ou=Machines, o=MyCompany.com
changetype: add
objectclass: device
objectclass: mrbamachine
cn: acme
ou: Machines
description: A machine called acme

dn: cn=Sales, ou=Custom Groups, o=MyCompany.com
changetype: add
objectclass: top
objectclass: groupofnames
ou: Custom Groups
cn: Sales
member: uid=dave, ou=People, o=MyCompany.com
member: cn=acme, ou=Machines, o=MyCompany.com
```

## Creating a group of groups

The following LDIF entry creates a group that contains multiple sub-groups.

```
dn: cn=All groups, ou=Custom Groups, o=MyCompany.com
changetype: add
objectclass: top
objectclass: groupofnames
ou: Custom Groups
cn: All groups
member: cn=Sales ou=Custom Groups, o=MyCompany.com
member: cn=Engineering, ou=Machine Groups, o=MyCompany.com
```

## Policy Manager naming conventions

Although the Policy Service plug-in and LDAP schema provide arbitrary naming rules for Policy Management entries and a many-to-many reference model (one policy can be targeted to many groups), Policy Manager places the following limitations on this flexibility:

- `mrbaSubscription` entries can only target one entity—a machine, machine group, user, user group, or collection.
- `mrbaSubscription` entries must have the `mrbasn` attribute present to be recognized by the BMC Policy Manager application. Note that only the presence of this attribute is required—it can have any value, such as `true`.

Chapter  
**4**

# Setting up user accounts

The BMC Policy Management infrastructure involves several components that enable you to manage a secure network. This section explains how to configure these components for best security.

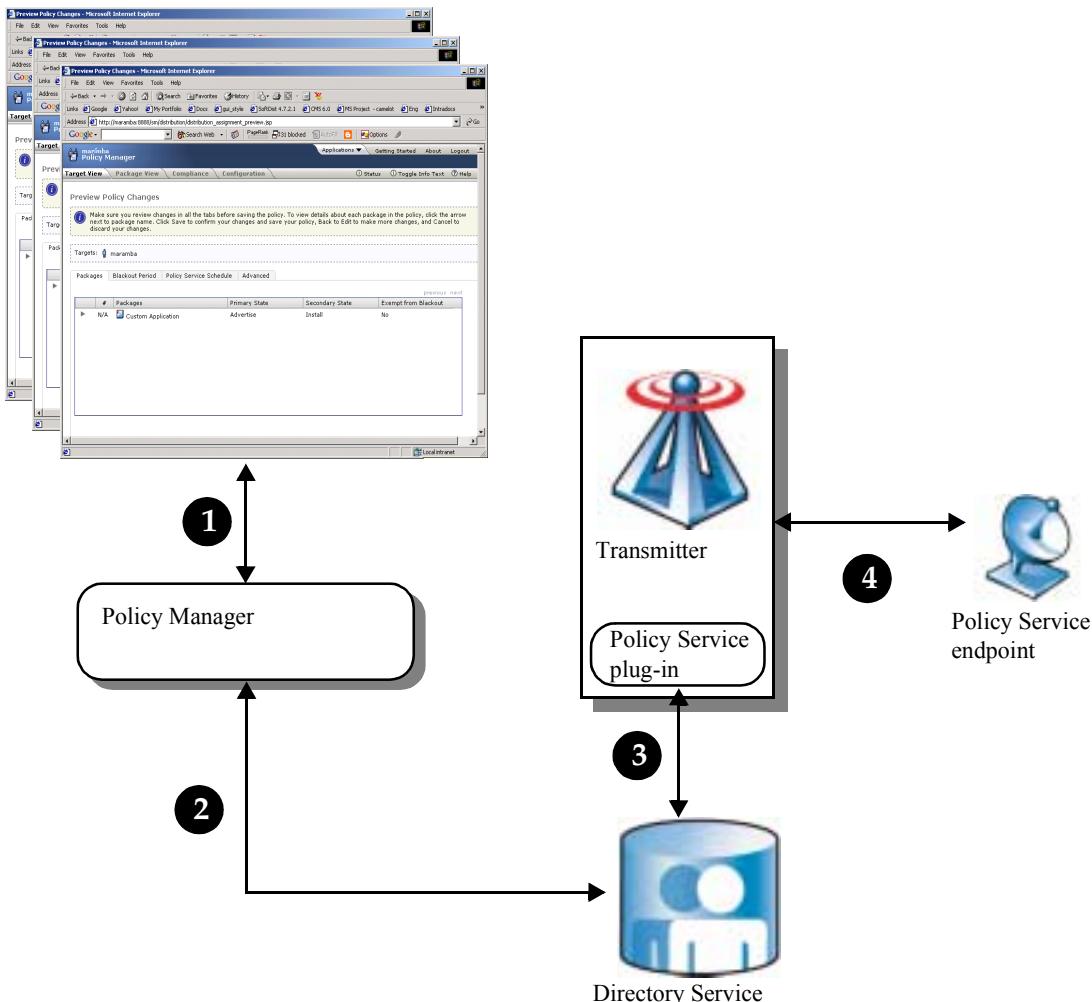
The following topics are provided:

- Policy Management plug-in publishing (page 67)
- Policy Manager CLI permissions (page 67)
- Policy Manager permissions (page 68)
- Directory tree permissions needed for creating policies (page 71)

# Security between components

Figure 4-1 summarizes the flow of information between the Policy Management components: Policy Manager and its Web browser GUI, directory service (or LDAP), the Policy Service plug-in, and Policy Service. “Information flow” on page 67 describes the interaction between components in terms of security.

Figure 4-1: Overview of connections made in Policy Management.



## ► Information flow

- 1 The Web browser GUI sends a request to Policy Manager. This message is not secure because CMS, which handles the requests, does not support SSL (Secure Sockets Layer) for Web applications. Therefore, browser communications to Policy Manager are susceptible to interception unless the network itself is secure.
- 2 Policy Manager saves policies to the directory service, but does *not* support authentication methods provided by the directory service.
- 3 The Policy Service plug-in connects to the directory service to obtain policy information, and does not support authentication methods provided by the directory service.
- 4 Policy Service contacts the Policy Service plug-in to request the policy for the endpoint.

## Policy Management plug-in publishing

When you publish the Policy Service plug-in from Policy Manager, the plug-in is configured to connect to the directory service that stores policies. The configuration information is crucial if targets are to receive policies. Therefore, only members of the primary administrator groups that have been specified in the system settings (CMS) are allowed to publish the plug-in.

Users who publish the plug-in from the command line interface (CLI) must authenticate to the directory service. This level of security is usually sufficient because only administrators who have access to the machine that hosts the Policy Manager can use the CLI.

## Policy Manager CLI permissions

Because the command-line interface provides most of the capabilities of the browser-based interface, you must carefully consider who will be allowed to have execute permission to use `runchannel` in the tuner. Also, the CLI requires you to specify a user name and password (using the `-username` and `-password` options), just as you would when logging into the console to use the browser-based interface. See “Providing user authentication” on page 388.

# Policy Manager permissions

Policy Manager is a Web application that allows several administrators to use the system simultaneously. Any user listed in the directory service who belongs to the *primary administrators* group or *administrator* group can log in to the console and use Policy Manager. However, when you log in to the console, your view of targets and policies is limited by the permissions you have been given in the directory service. By thoughtfully setting these permissions, each administrator's area of access and responsibility can be controlled. The best time to set these permissions is during installation, before you give access to more than one administrator.

Also, the access control lists (ACLs) feature (added in version 6.0) allows you to control which targets each administrator can see and assign policies. See “Setting up access control lists” on page 153.

## Roles and access to Policy Manager features

Part of the initial installation and configuration is to specify which users are primary administrators, which users are standard administrators, and which users are operators.

At the minimum, you must set up two user groups. One will be assigned the *primary administrator* role, and the other will be assigned the *administrator* role. Any user listed in the directory service who belongs to the primary administrator group or administrator group can browse to Policy Manager and use it. By carefully setting permissions for these users, you can control each administrator's area of access and responsibility.

---

Note: Currently, users with the *operator* role do not have access to Policy Manager.

---

User roles control the access of administrators to certain pages in the GUI. For Policy Manager, you must at least be part of the group that is given the administrator role in order to log in. Members of a group given the primary administrator role have the additional ability to configure Policy Manager (see “Setting up user accounts” on page 65 for more information.) The views of targets and policies that you and other administrators see when logging in to Policy Manager, and the ability to assign targets to a policy, are controlled by the permissions granted in the directory service.

---

Note: To change the configuration settings, you must be logged in as a primary administrator. Part of the initial installation and configuration is to specify which users are primary administrators and which users are standard administrators. Standard administrators are prevented from accessing and changing the configuration settings for Policy Manager.

---

Users fall into one of the following classifications, based on permissions and roles:

- **Primary administrators.** Primary administrators can assign users and groups any of the four permissions (described above). They can assign permissions for targets and policies, as well as permissions for ACLs. Only primary administrators can assign ACL read and write permissions. Primary administrators belong to a group in the directory service that has been assigned the Primary administrators role on the User Roles page.
- **ACL administrators.** ACL administrators can assign permissions to policy administrators. They can assign permissions for targets and policies. ACL administrators belong to a group in the directory service that has been assigned the Administrators role on the User Roles page. They have been given ACL read and write permissions by a primary administrator.
- **Policy administrators.** Policy administrators have been assigned permissions for targets and policies by primary administrators or ACL administrators. They can use Policy Manager to create and edit policies for targets. However, they do not have access to the configuration pages in Policy Manager, and, therefore, cannot view or set permissions and ACLs. Policy administrators belong to a group in the directory service that has been assigned the Administrators role on the User Roles page. They have been given policy permissions by a primary administrator.

---

Note: You should only give permissions to users who have access to the console and to Policy Manager—that is, users who are members of the groups that you have mapped to roles on the User Roles page (Applications > Console > System Settings > User Authentication > User Roles).

---

For more information about assigning these roles to users or groups, see the *BMC Marimba Client Automation Installation Guide*, available on the BMC Customer Support website.

---

Tip: You can find out whether you logged in as a primary administrator, standard administrator, or operator by placing your mouse pointer over the Status icon ⓘ in the upper-right corner of the console.

---

## Identifying users who can access Policy Manager

Users who can access and create policies in Policy Manager must be identified in the directory service that is specified on the Directory Services page (System Settings > Data Source > Directory Services). Also, the groups to which these users belong must be specified on the User Roles page (System Settings > User Authentication > User Roles). For more information about these pages, click Help from the system settings pages.

When using Active Directory, Common Management Services (CMS) only resolves groups for which membership information is accessible in the connected Global Catalog. These can include the following groups:

- **Universal groups.** Users across any domain (as long as they are in the same forest) can be stored in a universal group. For example, CMS is running on the West domain and the `cn=UniversalCMSUsers,dc=abc,dc=com` universal group is the only group specified for the primary administrator role. Users from the East or West domains that belong to the universal group are able to log in to the West domain CMS. If the universal group is also specified for a role in another CMS running on the East domain, users from the East or West domains are able to log in to the East domain CMS as well.
- **Global groups.** Only users within the current domain can be members of a global group. For example, CMS is running in the West domain and the `cn=GlobalWestCMSUsers,dc=west,dc=abc,dc=com` global group is the only group specified for the primary administrator role. Because users from the West domain are the only ones that can be assigned to this global group, users from other domains will not be able to log in to the West domain CMS. If this group is specified for a role in another CMS running on the East domain, no users will be able to log in to the East domain CMS because member information for global groups is not replicated to Global Catalogs in other domains.

- **Domain local groups.** Users from the local domain and other domains can be members of the domain local group. For example, CMS is running in the West domain and the `cn=DomainLocalWestCMSUsers,dc=west,dc=abc,dc=com` domain local group is the only group specified for the primary administrator role. Because users from the West domain and other domains (such as the East domain) can be members of the domain local group, domain local groups can be used to allow occasional visitors from other domains to log in to the West domain CMS. However, if this domain local group is specified for a role in another CMS running on the East domain, no users will be able to log in to the East domain CMS because member information for domain local groups is not replicated to Global Catalogs in other domains. Using domain local groups for administrators is not recommended.

## Directory tree permissions needed for creating policies

Groups that are specified in the User Roles section for primary administrator groups and administrator groups can log in to the Policy Manager. However, adopting one of these roles does not control which targets can be viewed to define a policy. Instead, the view of targets is controlled by the permissions that have been set on various containers and entries used by Policy Manager in the directory service.

Permissions are defined as follows:

- **Write permission**—the administrator is able to create entries.
- **Read permission**—the administrator is able to view entries.
- **Create permission**—the administrator is able to create child entries.
- **Delete permission**—the administrator is able to delete child entries.

## Setting permissions for Active Directory

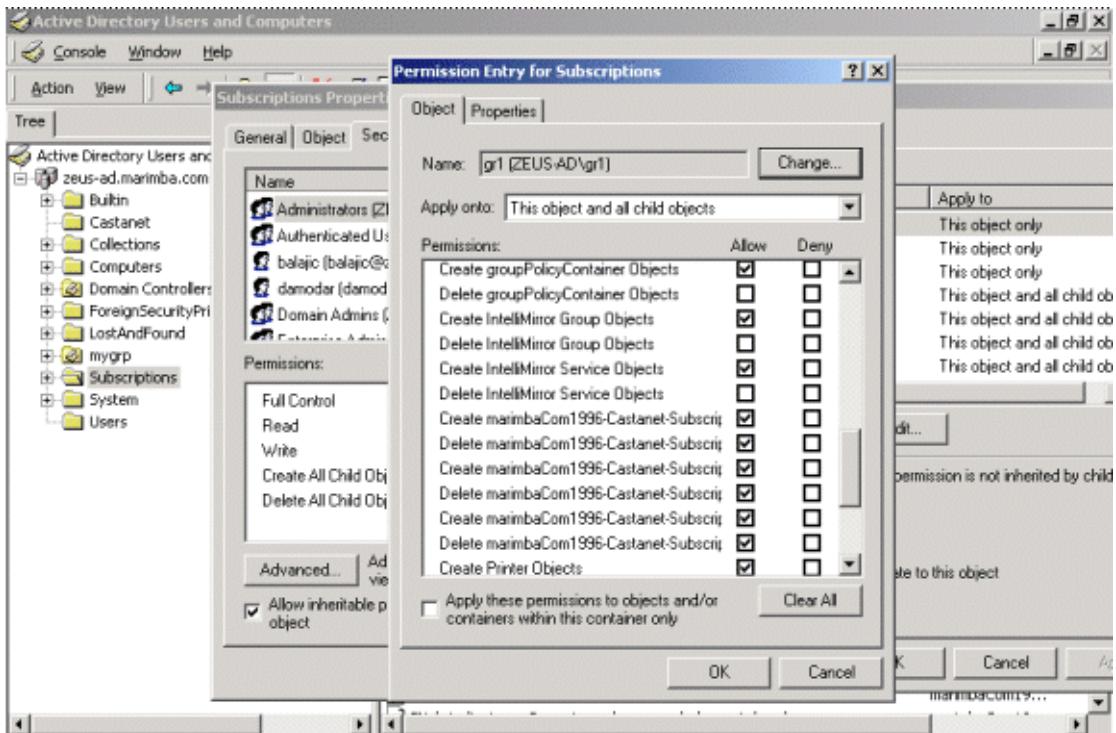
This section describes how control the view of targets by setting permissions for Active Directory. See the access control documentation for Active Directory for more detailed information.

### ► To set the permissions for an entry in Active Directory

- 1 Open the Windows Start menu and choose Programs -> Administrative Tools -> Active Directory Users and Computers.

- 2 Make sure that the advanced features are viewable by opening the View menu from the menu bar. Make sure that Advanced Features is checked.
- 3 Select the entry that needs to have permissions set for an administrator. Subscriptions has been chosen in Figure 4-2 on page 72

Figure 4-2: Setting permissions in active directory



- 4 Right-click and select Properties from the shortcut menu. A dialog box listing the properties for the entry will open.
- 5 Select the Security tab. On this page is a list of permission setting types. Set them as required. The following settings affect Policy Manager:
  - **Full Control**—The administrator can read, write, create, and delete child entries. For example, if this is applied to the Subscriptions object, the administrator can create policies and edit the policies created by another administrator.

- **Read**—The administrator can read the entry and the child entries created. For example, if this is applied to the Subscriptions object, the administrator can view policies that have been created, but will not be able to edit them.
- 6 If you only want an administrator to be able to create Policy Management entries under a particular container, click the Advanced button on the Properties window. A Permission Entry dialog box will open.
  - 7 Select check boxes that allow an administrator to create and delete MarimbaCom-1996-Castanet-Subscription\* entries. When you grant these permissions, an administrator can create and edit policies only within the entry.
  - 8 Finally, in order for any entry created under a container to inherit the permissions from the container, pull down the Apply onto menu and choose This object and all child objects.
  - 9 Save your work and close the console.

## Setting permissions for Sun Java System Directory Server (Sun One) directory

This section describes how control the view of targets by setting permissions for Sun Java System Directory Server (Sun One). See the access control documentation for Sun Java System Directory Server for more detailed information.

### ► To set the permissions for entries

- 1 Set up the Sun Java Enterprise System and the Sun Java System Applications Server.
- 2 Access the Sun Java Enterprise System Application Server and start the Admin Server.  
For example: Start > Program Files > Sun Java Enterprise System >Application Server > Admin Console.
- 3 Log into the Application Server. The Application Server Admin Console is displayed.
- 4 Under Common Tasks in the left pane, navigate to Applications > Web Applications.

- 5 Click the **amserver** Launch link to display the Sun Java System Access Manager login page.
- 6 Log into Access Manager, create users and groups, and give them administrator privileges.
- 7 Under the Configuration tab, click the Administration link.
- 8 Under Global Attributes, set the default permissions and select the **People Containers** and **Group Containers** permissions. These permissions are required for Policy Manager.

## Setting permissions for ADAM / AD LDS

For information about setting permissions for ADAM / AD LDS, see the *BMC Marimba Client Automation Installation Guide*, available on the BMC Customer Support website.

## Permissions for a multidomain forest environment (Active Directory only)

The following permissions are required for Policy Manager:

- **Permissions for browsing targets.** To browse any of the targets in the Global Catalog, administrators must have the appropriate read permissions to see targets in the forest. For this reason, giving administrators permissions for viewing targets in the Global Catalog is recommended. Additionally, if users are members of a domain local group, they must be given permissions to the Subscriptions container in the Policy Management domain.
- **Permissions for creating policies.** To create policies (and save them for targets), administrators must have write permissions to the Subscriptions container in the Policy Management domain. Administrators who are not from the Policy Management domain must be members of a domain local group in the Policy Management domain and must have the appropriate permissions to the Subscriptions container.

Additionally, if administrators will create or view collections, they must have write permissions to the Collections container in the Policy Management domain. Administrators who are not from the Policy Management domain must be members of a domain local group in the Policy Management domain and must have the appropriate write permissions to the Collections container.

## Subscriptions container permissions

In order to create and edit policies for a target, you must have write permissions to the Subscriptions container. The Subscriptions container is specified by the Policy Management configuration property `marimba.subscriptionplugin.subscriptionbase`. The default value for this property depends on the directory service:

- Active Directory: `ou=Subscriptions,ou=ConfigObjects,ou=BMC CM,ou=BMC Software,dc=company,dc=com`
- ADAM/ AD LDS: `ou=Subscriptions,ou=ConfigObjects,ou=BMC CM,ou=BMC Software,dc=company,dc=com`
- Sun ONE Directory: `ou=Subscriptions,ou=ConfigObjects,ou=BMC CM,ou=BMC Software,dc=company,dc=com`

## Defining read/write permissions for policies

For security and management reasons, you may not want to have administrators view and edit each other's policies. In many organizations, individual administrators are assigned to manage policies for specific groups.

To define permissions to read or write policies, you must first define Access Control Lists (ACLs) for each target, using the Common Management Services (CMS) console, and map each target to the appropriate ACL. See *Setting up access control lists*, in the *BMC Marimba Client Automation CMS and Tuner Guide*, available on the BMC Customer Support website.

## Collections

When collections are created from Report Center, they are stored in the location specified by `marimba.ldap.browse.collectionbase`. See “Attributes of the configuration object” on page 115. In Policy Manager, a Collection is a type of group (see “*Setting up collections*” on page 81).

Because Collections can be created even when no user is logged in to Policy Manager, the login and permissions of the user who specified the LDAP settings in the System Settings is used when a Collection is created (see the *BMC Marimba Client Automation Installation Guide*, available on the BMC Customer Support website). Therefore, this user must have write permission for the location specified by `marimba.ldap.browse.collectionbase`.

Any administrator who is to target Collections must have read permission for them. Therefore, this administrator must have read permissions on the location specified by `marimba.ldap.browse.collectionbase`. Specify that permissions be inherited from the parent so that Collections can be targeted as they are created by Report Center.

## Collections base for imported machines

Any machine added to a Collection must exist with in the Subscription domain (see “Setting up collections” on page 81). However, if Report Center creates a Collection with a member machine that does not exist, Report Center creates the machine entry. The directory server must have a container to hold these new machine entries. The new machines are stored differently in different directory servers:

Depending on your directory server, one of the following parameters specifies the location:

- **Active Directory** – The value of the `marimba.ldap/browse/machineimportbase` property specifies the location of new machines created in response to collections. The default value is `cn=BMC Computers,ou=BMC CM,ou=BMC Software,dc=subs,dc=com`.
- **Sun ONE** – Machines created by importing from version 4.7 machines flat files are stored in the location provided by the value of the `marimba.ldap/browse/importmachinebase` property. When forming Collections, Report Center creates machines in the same container in which it searches for them; by default, in `ou=machines`.

As with Collections, these machine entries can be created when no user is logged in to Policy Manager. Therefore, the login and permissions of the user who specified the LDAP settings in the System Settings is used when a machine account is created (see the *BMC Marimba Client Automation Installation Guide*, available on the BMC Customer Support website). This user must have write permission for the location specified by either of these parameters.

## CMS directory service and Policy Management plug-in permissions

The following permissions are required for the Policy Service plug-in and the CMS directory service:

- **Permissions for retrieving Policy Management configuration.** The administrator name and password used for the bind DN on the CMS Directory Services page (System Settings > Data Source > Directory Services) and Plug-in Configuration page (Policy Management > Configuration > Plug-in) must have the appropriate read permissions to retrieve the Policy Management configuration (the Configuration Management container in the domain) from the Global Catalog.

The administrator name and password used for the bind DN must have read permissions for users, machines, and groups for all membership information that must be retrieved, such as to authenticate users logging into CMS and Policy Manager. It must also have appropriate read permissions from the Policy Management domain to retrieve collection membership and policies, as well as write permissions for the Collections container (for distributed mode, the Collections container in each domain). Make sure that the permissions you set are recursive, so that the permissions are inherited by the child objects in the container. For example, in Active Directory, you should choose the This object and all child objects option.

It is recommended that the administrator whose name and password is used for the bind DN is a member of a universal group with the appropriate permissions for all the mentioned objects. The administrator name and password used for the bind DN must belong to at least one group besides the Domain User group.

- **Permissions for viewing the member-of attribute of groups.** The administrator name and password used for the bind DN on the CMS Directory Services page (Applications > Console > System Settings > Data Source > Directory Services) and Plug-in Configuration page (Configuration > Plug-in) must have the appropriate permissions to read the member-of attribute of groups from the Global Catalog to which the Policy Service plug-in connects. Because the member-of attribute of domain local groups and global groups are not replicated to the Global Catalog outside the groups' domain, using a universal group with the appropriate permissions for the member-of attribute is recommended.

- **Read permissions to the Subscriptions base container.** The base container is specified as “marimba.subscriptionplugin.subscriptionbase” in the Policy Management configuration entry. Read permissions are needed because the policies must be obtained for the targets.
- **Read and write permissions to the ACL container.** To use the access control lists (ACLs) feature of Policy Manager, the administrator name and password used for the bind DN must have read and write permissions to the container where ACLs are stored, for example,  
`ou=Acl,ou=ConfigObjects,ou=BMC CM,ou=BMC Software,dc=company,dc=com`. For more information about the ACLs feature, see “Setting up access control lists” on page 153.

To make sure that ACLs and permissions are secure, you should restrict access to the following objects:

- **ACL container.** This container contains the ACLs and permissions that have been assigned to Policy Manager administrators. You might want to restrict read and write access to this container so that administrators cannot change the ACLs and permissions assigned to them.
- **Subscription configuration object.** This object contains the attribute `marimba.subscription.acl`, which enables and disables ACL functionality. You might want to restrict write access to this object so that access to this object is limited to primary administrators who have the authority to turn off the ACLs feature. See “Enabling and disabling ACL functionality” on page 154.
- **Read and write permissions to the Subscription configuration object.** To configure compliance settings and use the Compliance Options page, the administrator name and password used for the bind DN must have read and write permissions to the Subscription configuration object. For more information about configuring the policy compliance settings, see “Configuring policy compliance settings” on page 155.

Table 4-1 on page 79 shows the permissions required for the different containers in the directory service. In this table, the term *administrators* refers to both primary and standard administrators. If different permissions are required for primary and standard administrators, the specific type is mentioned in the table.

Table 4-1: Permissions for directory service containers

Container	Users	Permissions	Comments
ACL	Administrators	None	Administrators do not need permissions to the ACL container. Restricting permissions to the ACL container is recommended so that administrators cannot change the ACLs assigned to them through Policy Manager.
	CMS bind DN user	Full control	Full control permissions are required because this is the account used to assign, edit, and retrieve ACLs for administrators.
Collections	Administrators	Read	Read permissions are required so that administrators can view and browse the machines and groups in the Collections container.
	CMS bind DN user	Full control	Full control permissions are required because this is the account used when creating collections.
LDAPCollections	Administrators	Read	Read permissions are required so that administrators can view and browse the machines and groups in the LDAPCollections container.
	CMS bind DN user	Full control	Full control permissions are required because this is the account used when creating LDAP query collections.
BMC CM	Primary administrators	Full control	Full control permissions are required so that administrators can change and retrieve configuration information for Policy Manager.
	Standard administrators	Full control	Full control permissions are required so that administrators can change and retrieve configuration information for Policy Manager.
	CMS bind DN user	Full control	Full control permissions are required so that administrators can change and retrieve configuration information for Policy Manager.

Table 4-1: Permissions for directory service containers

Container	Users	Permissions	Comments
Subscriptions	Administrators who can create and edit policies for all endpoints	Full control	Full control permissions are required so that these administrators can create and edit policies for all endpoints. The all_all object must inherit the Subscriptions container permissions.
User, machine, and group containers	Administrators	Read for the particular containers for which they are responsible	Read permissions are required so that administrators can view and browse the targets in the directory service.
CMS bind DN user		Read	Read permissions are required so that CMS can retrieve membership information for users and group to authenticate them when logging in to use BMC Marimba Client Automation applications.

Chapter

# 5 Setting up collections

A collection is a list of machine names that you can use for targeting by Policy Management. You can create collections by running either a database or LDAP query.

The following topics are provided:

- What is a collection? (page 82)
- LDAP query collections (page 82)
- Database query collections: integration with the inventory module (page 92)

## What is a collection?

A collection is a list of machine names that you can use for targeting by Policy Management. You can create collections by running either a database or LDAP query:

- **Running an LDAP query using Policy Manager.** To create a collection based on an LDAP query, you use Policy Manager to create a query that will return a dynamic list of users or machines that meet your criteria and then refresh the query, either manually or according to a schedule. The resulting list of users or machines is available for selection as a target in Policy Manager.
- **Running a database query using the BMC Inventory module.** If you have the BMC Marimba Client Automation Inventory module, you can use Report Center to create collections that you can then use as targets in Policy Manager. To create a collection, you use Report Center to create a query that will return a list of machines that meet your criteria, save this query in the Collections folder (in Report Center), and then run the query, either manually or according to a schedule. Depending on how permissions are configured and what permissions you have, the queries saved and run in the Collections folder result in a list of machines that is available for selection as a target in Policy Manager. See the note about [collections and ACLs](#) in “Setting up access control lists” on page 153.

## LDAP query collections

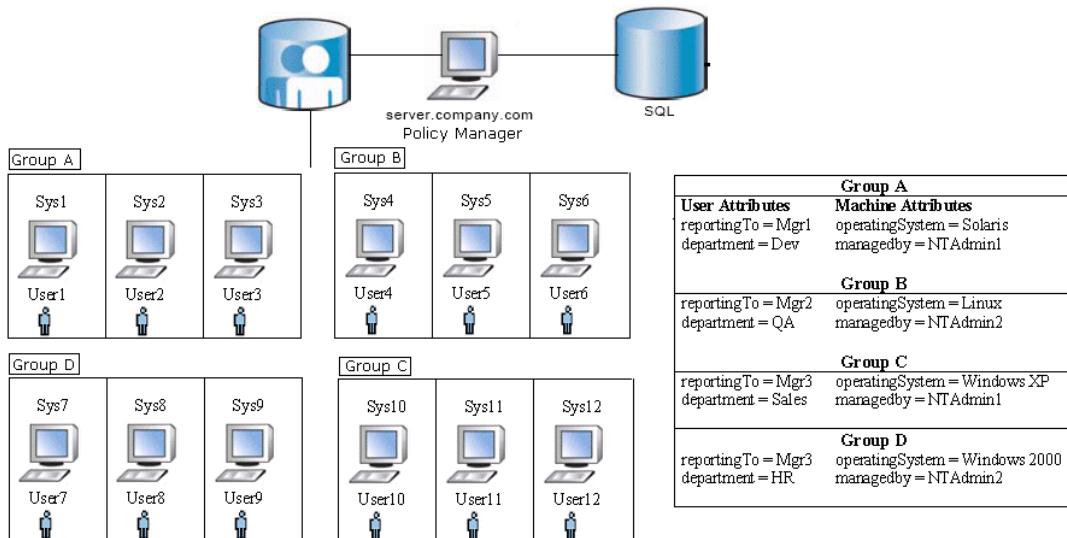
To create, list, preview, run, refresh, and delete LDAP query collections, you use the command-line interface for Policy Manager.

- “Overview of LDAP query collection” on page 83
- “Prerequisites for using LDAP query collections” on page 84
- “Configuring LDAP query collections” on page 84
- “Viewing LDAP query collections” on page 87
- “Previewing the results of an LDAP query collection” on page 89
- “Modifying an existing LDAP query collection” on page 90
- “Deleting LDAP query collections” on page 91
- “Refreshing LDAP query collections” on page 92

## Overview of LDAP query collection

The following figure provides a functional description of LDAP query collection.

Figure 5-1: Functional description of LDAP query collection



### Machine Attributes based LDAP Queries

To retrieve all machines installed with "Solaris" or "Unix"  
 $((operatingSystem=Solaris) \text{ } (operatingSystem=Unix))$

To retrieve all machines installed with "Windows XP" and administered by "NTAdmin1"  
 $(\&(operatingSystem=Windows\text{ }XP) \text{ } (managedby=NTAdmin1))$

### User Attributes based LDAP Queries

To retrieve all users working in "Sales" or "HR" department.  
 $((department=Sales)\text{ }(\department=HR))$

To retrieve all users in "Sales" and reporting to the manager "Mgr3"  
 $(\&(department=Sales)(reportingTo=Mgr3))$

The diagram shows four groups of machines/users in the directory server. To the right is a list of user and machine attributes. The example machine attributes-based and user attributes-based LDAP queries use those attributes.

In the first LDAP collection query based on machine attributes, you create a collection of machines with the Windows operating system. The result is Sys1 to Sys3. In the second LDAP collection query based on user attributes, you create a collection of users who report to Mgr3. The result is User7 to User12.

## Prerequisites for using LDAP query collections

You must meet the following prerequisites before using LDAP query collections:

- Policy Manager must be installed and configured.
- Report Center must be installed and configured. Policy Manager uses Report Center's database connection to verify that users or machines are managed by BMC Marimba Client Automation.
- You must have proper permissions for the database and the directory service, as required when you installed and configured Report Center and Policy Manager. See "Setting up user accounts" on page 65 and the *Report Center Administrator's Guide*, available on the BMC Customer Support website.
- You must configure LDAP query collections, as described in the following section.
- You must be familiar with the syntax required for LDAP queries.
- The endpoints that you want to include in your LDAP query collection must have Scanner Service and Policy Service running on them.

## Configuring LDAP query collections

Before you can use LDAP query collections, set the following attributes for the Subscription configuration object:

- `marimba.subscriptionplugin.ldapcollectionbase`—Distinguished name of the container object for storing the LDAP query collection group objects. By default, Schema Manager sets this attribute to `OU=LDAPCollections,ou=BMC CM,ou=BMC Software,DC=company,DC=com`.
- `marimba.subscriptionplugin.ldapcollectionsched`—Enables and disables the schedules you set for LDAP query collections.
- `marimba.subscriptionplugin.ldapcollectionmode`—The value for this attribute is centralized by default. At this time, ADAM / AD LDS and Sun Java System Directory Server (Sun One) support only the centralized mode for LDAP query collections.

## ► To configure LDAP query collections

- Using Policy Manager, configure LDAP query collections by setting the previously described attributes:

```
runchannel <PolicyManager_URL> -user <user_name> -password
<password> -ldapqc -config <key> <value>
```

where <key> and <value> represent the following:

Key	Value
marimba.subscriptionplugin.ldapcollectionbase	Distinguished name (DN) of the container object for storing the LDAP query collection group objects. For example: OU=LDAPCollections,DC=company,DC=com
marimba.subscriptionplugin.ldapcollectionsched	on or off
marimba.subscriptionplugin.ldapcollectionmode	centralized
	<b>Note:</b> Do not change this value. ADAM / AD LDS and Sun ONE require centralized collection mode.

You must restart Policy Manager after setting these attributes.

## Creating an LDAP query collection

To create an LDAP query collection, you must specify a collection name, the LDAP query, and, optionally, the schedule for refreshing the collection.

## ► To create an LDAP query collection

- Using Policy Manager, specify the required and optional information for creating an LDAP query collection:

```
runchannel <PolicyManager_URL> -user <user_name> -password
<password> -ldapqc -create -cname "<collection_name>" -query
"<ldap_query>" [-searchBase "<dn>"] [-schedule
"<date_time_range_frequency>"] [-filter {usersonly|machinesonly}]
```

where:

<collection_name>	The name that you want to give to the collection.
<ldap_query>	The query, in LDAP syntax, that is used for gathering the list of users and machines. You can enter simple queries, such as searching for entries with users in the CN: (cn=users) Or, you can enter more complex queries that use conditions, such as searching for entries with either users or groups in the CN by using the or notation ( ): ( (cn=users) (cn=groups)) For more information about LDAP search filters, see <a href="http://www.ietf.org/rfc/rfc2254.txt">http://www.ietf.org/rfc/rfc2254.txt</a> .
<dn>	The base DN that you want to use for the LDAP query. For example: dc=company,dc=com
<date_time_range_frequency>	The schedule that you want to use for refreshing the LDAP collection. For more information about the syntax for the schedule, see “Format for update or repair schedules” on page 422.
usersonly   machinesonly	Filter the results so that only users or machines are included in the list.

After you run this command, an object is created in the directory service (in the container pointed to by

`marimba.subscriptionplugin.ldapcollectionbase`. The object has the name you specified, and the other information that you specified (LDAP query, schedule, and so on) are stored as attributes of the object. The object is populated with the list of machines and users only when the query is refreshed from the command line or at the next scheduled refresh time; it is not populated at the time that the collection is created.

For example:

```
runchannel http://localhost:5282/Marimba/Current/  
SubscriptionManager -user alfonzo -password alfonzo -ldapqc -  
create -cname "all_bmc_users" -query "(department=bmcdev)" -  
searchBase "dc=company,dc=com" -schedule "every day update every  
30 minutes between 9:00AM and 5:00PM" -filter usersonly
```

Here are some more example LDAP queries: (Notice that some of these examples assume that you have extended your LDAP schema to add new attributes, such as department, manager, and deviceType.)

- All users in the department bmcdev: (department=bmcdev)

- All Windows 2000 machines: operatingSystem=Windows 2000 Professional
- All users who report to manager X: (manager=X)
- Any device or laptop that connects through a virtual private network (VPN) from outside the corporate network: (deviceType=roaming)
- All machines that belong to the department Accounts and managed by manager XYZ: (&(department=Accounts)(manager=XYZ))
- All machines that start with the serial number X12 or the department Marketing: (|(serialNumber=X12)(department=Marketing))
- All employees who have subordinates or has the employee type Manager, and has the department number 1001: (&(|(hassubordinates=yes)(employeetype=Manager))(departmentnumber=1001))
- All machines and users who have subordinates (assuming that the hassubordinates attribute is present for both machines and users): (hassubordinates=yes)

---

Note: When using Active Directory, Policy Manager searches the Global Catalog if you have configured CMS to use automatic discovery. Generally, only a few default attributes are kept in the Global Catalog. If the search that you want to perform for LDAP query collections is based on the other attributes, you must modify those attributes so that they replicate to the Global Catalog. For instructions for modifying these attributes, see Microsoft's website: <http://support.microsoft.com/kb/248717/EN-US/>.

---

## Viewing LDAP query collections

You can view a list of all the LDAP query collections or information about one LDAP query collection.

### ► To view a list of all LDAP query collections

- Using Policy Manager, specify that you want a list of all the available LDAP query collections:

```
runchannel <PolicyManager_URL> -user <user_name> -password <password> -ldapqc -list -all
```

For example:

```
runchannel http://localhost:5282/Marimba/Current/  
SubscriptionManager -user alfonzo -password alfonzo -ldapqc -  
list -all  
Logged in Policy Manager as user : alfonzo
```

List of all query collection objects.

```
Collection Name : myLDAPquery  
Query Schedule : every day update every 30 minutes between 9:00AM  
and 5:00PM  
DateTime Last Run : Thu Feb 10 10:00:00 PST 2005  
DateTime Next Run : Thu Feb 10 10:30:00 PST 2005  
Created By : CN=Alfonzo,OU=Engineering,DC=company,DC=com  
-----  
Collection Name : myLDAPquery1  
Query Schedule : every day update every 30 minutes between 9:00AM  
and 5:00PM  
DateTime Last Run : Thu Feb 10 10:00:00 PST 2005  
DateTime Next Run : Thu Feb 10 10:30:00 PST 2005  
Created By : CN=Alfonzo,OU=Engineering,DC=company,DC=com  
-----  
Collection Name : myLDAPquery2  
Query Schedule : every day update every 30 minutes between 9:00AM  
and 5:00PM  
DateTime Last Run : Thu Feb 10 10:00:00 PST 2005  
DateTime Next Run : Thu Feb 10 10:30:00 PST 2005  
Created By : CN=Alfonzo,OU=Engineering,DC=company,DC=com
```

## ► To view information about one LDAP query collection

- Using Policy Manager, specify the name of the LDAP query collection for which you want to view information:

```
runchannel <PolicyManager_URL> -user <user_name> -password  
<password> -ldapqc -list -cname "<collection_name>"
```

For example:

```
runchannel http://localhost:5282/Marimba/Current/  
SubscriptionManager -user alfonzo -password alfonzo -ldapqc -  
list -cname myLDAPquery2
```

Logged in Policy Manager as user : alfonzo

#### LDAP Query collection object details

Collection Name : myLDAPquery2

Query Syntax : (|(cn=users) (cn=groups))

Filter : usersonly

Query Search Base : dc=*company*,dc=*com*

Query Schedule : every day update every 30 minutes between 9:00AM and 5:00PM

DateTime Last Run : Thu Feb 10 10:00:00 PST 2005

DateTime Next Run : Thu Feb 10 10:30:00 PST 2005

Created By : CN=Alfonzo,OU=Engineering,DC=*company*,DC=*com*

## Previewing the results of an LDAP query collection

You can preview the results for an existing LDAP query collection or for a new collection that you have not yet created.

### ► To preview the results for an existing LDAP query collection

- Using Policy Manager, specify the LDAP query collection for which you want to preview results:

```
runchannel <PolicyManager_URL> -user <user_name> -password <password> -ldapqc -preview -cname "<collection_name>"
```

For example:

```
runchannel http://localhost:5282/Marimba/Current/ SubscriptionManager -user alfonzo -password alfonzo -ldapqc -preview -cname myLDAPquery2
```

Logged in Policy Manager as user : alfonzo

LDAP query has found no matching objects.

► To preview the results for a new LDAP query collection that you have not yet created

- Using Policy Manager, specify the query and other attributes for the new LDAP query collection for which you want to preview results:

```
runchannel <PolicyManager_URL> -user <user_name> -password  
<password> -ldapqc -preview -query "<ldap_query>" [-searchBase  
"<dn>"] [-filter {usersonly|machinesonly}]
```

For example:

```
runchannel http://localhost:5282/Marimba/Current/  
SubscriptionManager  
-user alfonzo -password alfonzo -ldapqc -preview -query  
"(cn=lab_machine*)"  
Logged in Policy Manager as user : alfonzo
```

#### Preview Results

```
CN=lab_machine1,CN=MarimbaComputers,CN=Computers,dc=company,dc  
=com
```

```
The total number of matching collections = 1
```

## Modifying an existing LDAP query collection

To modify an LDAP query collection, you must specify the collection name and the new information with which you want to modify the collection.

► To modify an LDAP query collection

- Using Policy Manager, specify the collection name and the new information for LDAP query collection:

```
runchannel <PolicyManager_URL> -user <user_name> -password  
<password> -ldapqc -modify -cname "<collection_name>" [-query  
"<ldap_query>"] [-searchBase "<dn>"] [-schedule  
"<date_time_range_frequency>"] [-filter  
{usersonly|machinesonly}]
```

For more information about the options for this command, see “Creating an LDAP query collection” on page 85.

## Moving collections in LDAP

When moving collections from one location to another location in LDAP, the Active Directory MaxValRange property of the LDAPAdminLimits attribute must be greater than or equal to the number of machines in the collection. The default value of the MaxValRange property is 1500.

For example, if a collection has 2000 member machines and you want to move the collection from one location in LDAP to another, then the value of the MaxValRange property must be set to at least 2000. If your collection has 5000 or more machines, set the value of the MaxValRange property to 5000 because collections that contain more than 5000 machines are divided into subgroups of 5000 machines.

After the move, you can reset the MaxValRange property to 1500 if needed. You can find more information on the MaxValRange property from the Microsoft Support site: <http://support.microsoft.com/>

## Deleting LDAP query collections

You can delete LDAP query collections when you no longer need them. If you are a primary administrator, you can delete all the LDAP query collections. If you are a standard administrator, you can delete the LDAQ query collections that you created only.

### ► To delete an LDAP query collection

- Using Policy Manager, specify the LDAP query collection that you want to delete:

```
runchannel <PolicyManager_URL> -user <user_name> -password  
<password> -ldapqc -delete {-cname "<collection_name>" | -all}
```

For example:

```
runchannel http://localhost:5282/Marimba/Current/  
SubscriptionManager  
-user alfonzo -password alfonzo -ldapqc -delete -cname  
"myLDAPquery1"
```

---

Note: When a collection or a folder created for a collection is deleted, the associated policies and acls are deleted from LDAP.

---

## Refreshing LDAP query collections

You can update the results of an LDAP query collection by refreshing it. Refreshing runs the LDAP query, so that the latest results appear in Policy Manager.

### ► To refresh an LDAP query collection

- Using Policy Manager, specify the LDAP query collection that you want to refresh:

```
runchannel <PolicyManager_URL> -user <user_name> -password  
<password> -ldapqc -refresh -cname "<collection_name>"
```

For example:

```
runchannel http://localhost:5282/Marimba/Current/  
SubscriptionManager  
-user alfonzo -password alfonzo -ldapqc -refresh -cname  
"myLDAPquery1"  
Logged in Policy Manager as user : alfonzo
```

Policy Manager Succeeded

## Database query collections: integration with the inventory module

To create, run, and delete database query collections, you use the Report Center component of the Inventory module. See the *Report Center Administrator's Guide*, available on the BMC Customer Support website. This section discusses issues you might encounter when setting up and using collections in Policy Manager, including the following topics:

- “Collection modes” on page 93
- “Working with database query collections” on page 98
- “Installation issues when using database query collections” on page 96
- “Forming a list of collection members” on page 99

## Collection modes

You can use database query collections in one of two modes:

- **Centralized mode.** In this mode, administrators from one Active Directory domain can create and manage collections in any domain. You can use this mode in a single-domain Active Directory environment, or in a multi-domain environment where the load for Policy Management requests is not expected to be high. If you use ADAM / AD LDS or Sun Java System Directory Server (Sun One) Directory, you must run in centralized mode. For an example of a centralized mode architecture, see Figure 5-2 on page 95.

If you want to use centralized mode and you want Report Center to run in a domain different from the one where you store information about collections (the Collections container), then you must set the domain where the Collections container is located in the following tuner property:

```
marimba.ldap.admanagementdomain=<domainOfCollectionsContainer>
```

This property is useful when you want to use machines that are not in a domain, such as UNIX machines, for collections in Policy Manager or Report Center. You must set this tuner property on the following:

- machines where Policy Manager, Report Center, and the CMS are running
- endpoint machines where the Policy Service is running
- master transmitter and repeater machines running on Linux

The Global Catalog must also be enabled in the domain where the Collections container is located.

- **Distributed mode.** This mode specifies that you have one Report Center installed per Active Directory domain and you want to restrict administrators within a given domain so that they manage only collections within their local domain. The Report Center in each domain creates collections for the machines in that domain. For an example of a distributed mode architecture, see Figure 5-3 on page 96.

You usually set the mode during installation. For Active Directory, the LDIF scripts that you run during installation specifies distributed as the default mode. If upgrading from a previous release, you can specify the mode that you want to use during the upgrade process (you specify your choice by running an upgrade command, as described in the upgrade section of the *BMC Marimba Client Automation Management Installation Guide*). If you want to change modes after installation, see the instructions in the collections chapter of the *BMC Marimba Client Automation Report Center User Guide*.

For ADAM / AD LDS and Sun Java System Directory Server (Sun One) Directory, the only mode available is centralized.

Figure 5-2: Example of centralized mode architecture (Single domain, active directory)

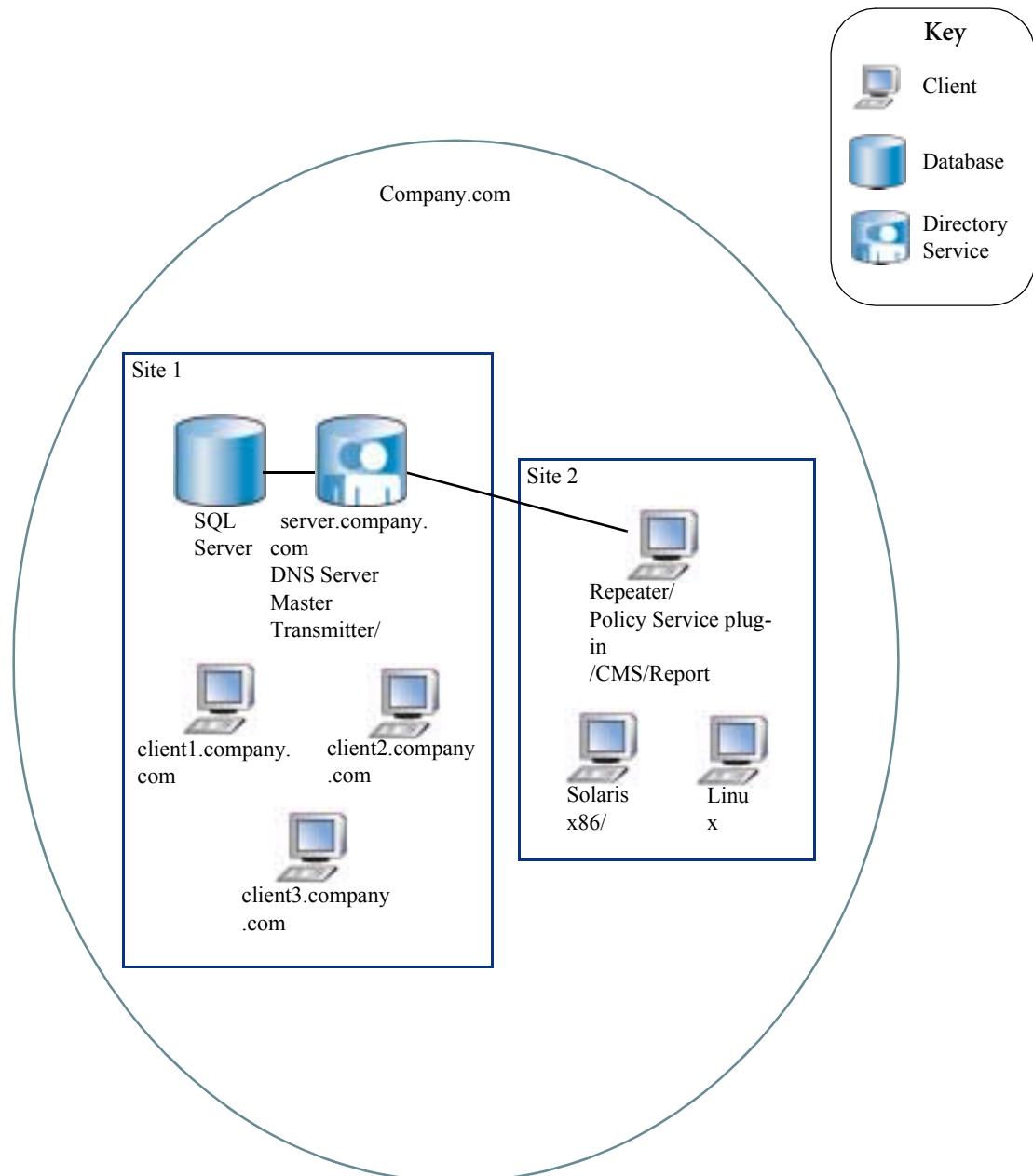
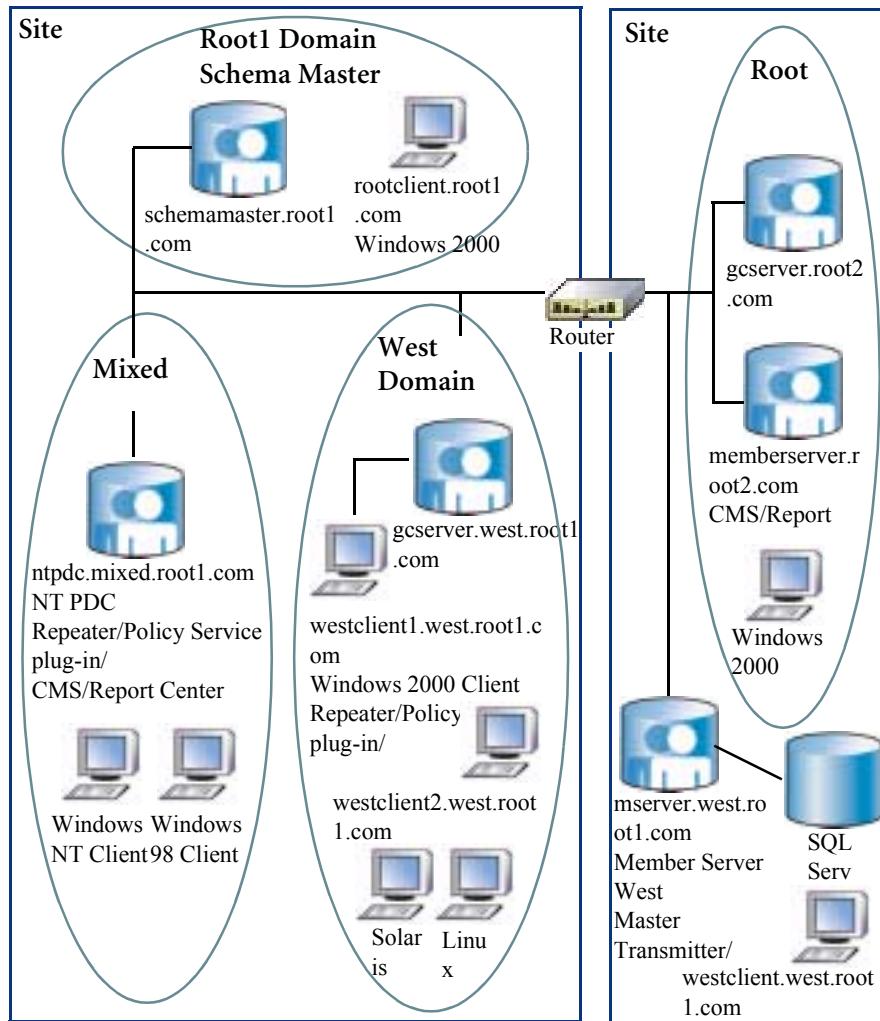


Figure 5-3: Example of distributed mode architecture (Multi-domain, Active Directory)



## Installation issues when using database query collections

This section discusses some issues to consider when using collections.

**Running LDIF scripts for collections.** The machine names in the collections need to be stored in the directory service. Therefore, before you can actually run collections queries, you must make sure that the Policy Management schema has been installed. That is, you must use Schema Manager to export and run the LDIF script, and then configure your directory service. For instructions, see the *BMC Marimba Client Automation Installation Guide*, available on the BMC Customer Support website.

The LDIF script you run when you install BMC Policy Management extends the directory service schema in order to let you use collections created by Report Center. The extended schema specifies the entry points for the various containers and sets properties for the Subscription configuration object. The Subscription configuration object identifies the base location for the Collections container, among others. The Subscription configuration object is located in the ConfigObjects container, a child of the BMC Marimba Client Automation container.

You can customize the schema by editing the LDIF scripts. Note that BMC Marimba Client Automation components create collection entries from the class mrbacollection, and the intermediate groups they refer to as groupOfNames entries. No hooks are provided to specify different objectclass types.

After the schema has been extended and the Subscription configuration object created, Report Center is able to save the results of its collections query in the directory service. Users of Report Center can create collection queries and save them, but they cannot save the results of the query until the directory service schema has been extended and the Subscription configuration object has been created. After the directory service has been set up, users of Report Center can create queries and save them to be run later. The query can be run either manually or on a schedule. Report Center then creates a collection object in the directory service to store the results of the query.

**Setting permissions for the Collections container.** Because collections can be created according to a schedule, possibly when no user is logged in to Report Center, the login credentials and permissions of the user specified on the Data Source > Directory Service page in the system settings are used when a collection is created. Therefore, this user must have permission to write to the collectionbase container (specified by the BMC Marimba Client Automation configuration object property marimba.ldap.browse.collectionbase). For more information about permissions and security issues, see “Setting up user accounts” on page 65.

**Creating Collections containers for multiple domains.** If you use Active Directory and use collections in a multidomain forest environment, and you want to run distributed mode, you must generate and then run an LDIF script to create Collections containers in each domain where Report Center runs. Distributed mode means that you have one Report Center installed per Active Directory domain and you want to restrict administrators within a given domain so that they manage only collections within their local domain. The LDIF script that you generate creates Collections containers for each domain where you need collections. For instructions, see the *BMC Marimba Client Automation Installation Guide*, available on the BMC Customer Support website.

**Configuration parameters for collections.** The Subscription configuration object contains configuration parameters that specify where and how collections are stored. For descriptions of these parameters, see “Collections configuration parameters” on page 121.

## Working with database query collections

To create, run, and delete collections, you use the Report Center component of the Inventory module. For instructions and detailed information, see the *BMC Marimba Client Automation Report Center User Guide*, available on the BMC Customer Support website.

In Policy Manager, you can browse and assign packages to collections just as you would other targets.

## Using the domain controller

The domain controller (DC) is used for group membership resolution of collections. A machine that belongs to a collection gets the policy assigned to the collection only if the machine and the plug-in run in the domain where the collection is created. Make sure that the bind DN specified for the plug-in has permissions to read the member-of attribute from all the DCs.

## Storing and displaying large collections

Directory services are limited in the number of members that can be contained in a single group. To overcome this limit in Active Directory, Microsoft recommends breaking a large group into smaller intermediate groups, then nesting those smaller groups within the large group. BMC Marimba Client Automation components take this approach when they create and display Report Center collections, which can be very large machine groups.

When you expand a collection in Policy Manager, you reveal the intermediate groups. These groups are assigned names such as mycollection\_resultset\_0, mycollection\_resultset\_1, and so on. You can expand each intermediate group to view the members of the collection, but you cannot target the members or the intermediate groups directly—you must target the collection as a whole. The reason is that a collection's intermediate groups and individual members may disappear from the collection on the next Report Center query and the members of an intermediate group may change.

---

Note: If you have configured Policy Manager to obtain user information from a transmitter, you will discover that intermediate groups can be targeted. This behavior is due to the way a directory service stores and returns information. Do not attempt to target the intermediate groups.

---

## Forming a list of collection members

A collection query returns a list of machines, in the form of host names. Before it stores the list, Report Center must convert it into a list of distinguished names. That is, it must use a host name to locate a computer or machine object in the directory service.

In Active Directory, for example, computer entries are typically stored in the Computers container. A computer distinguished name (DN) has the form `cn=<hostname>,cn=computers,<Suffix>`. Report Center adds each DN to the list of members of the collection, populating the collection with the DNs. A collection, then, is the list of computer DNs that results from a list of host names returned by a Report Center query. After all the DNs have been added to the collection, Report Center commits the transaction to the directory service.

**How the machine names are stored in the directory service.** When a collections query returns a machine name, such as `machine1.company.com`, the collection in the directory service contains an attribute of the form: `cn=machine1, cn=Computers, dc=company, dc=com` (for Active Directory), `cn=machine1, cn=MarimbaComputers, cn=Computers, dc=company, dc=com` (for ADAM / AD LDS), or `cn=machine1, ou=Machines, dc=company, dc=com` (for Sun ONE Directory).

## Creating collection members

Occasions arise in which Report Center cannot successfully find a computer or machine object corresponding to a host name. For example, consider a case in which Report Center and Policy Manager are configured to use Active Directory, and a Report Center scan returns the name of a UNIX machine as part of its result set. The UNIX machine is not a member of any Windows Active Directory domain and a search for the host name fails. In this case, Report Center creates a computer object for the machine and adds the DN of the object to the collection.

Sun Java System Directory Server (Sun One) Directory does not have the concept of domains, nor of computer entries. However, the LDIF installation scripts extend the schema to create an object class called `mrbamachine`. So, in the case of Sun ONE Directory, it is entirely likely that the first time Report Center runs a collection query, it will have to create all of the machine entries in the result set in the directory service. However, the next time the collection query runs, Report Center will find the machine entries that were created on the previous query and will not need to re-create them. ADAM / AD LDS creates machine entries in a similar manner.

The techniques required to create machine entries in the directory service are slightly different between Sun Java System Directory Server (Sun One) Directory and Active Directory. In the case of Active Directory, Report Center adds attributes, one of which is the `sAMAccountName` attribute. All users and computers have this attribute, but account names are often the same for users and computers—when user `curly` uses a computer with host name `curly`, for example. To draw a distinction between the two attribute values, a dollar sign (\$) is appended to the end of the `sAMAccountName` attribute value for a computer object, such as `curly$`. This technique makes the `sAMAccountName` attribute unique within the domain, as it must be.

When Report Center creates a computer object, its account is enabled by default. If you want the computer account to be disabled by default, then set the `marimba.subscriptionplugin.useraccountcontrol=514` property in Subscription Config objects under `OU=ConfigObjects,OU=BMC CM,OU=BMC Software` on LDAP. As a result, a computer that identifies itself with a disabled account name cannot log in. In the case of Active Directory, the entry point that specifies the location of computer entries that BMC Marimba Client Automation components create is `marimba.ldap/browse/collectionmachinebase`. The installation LDIF script creates a BMC Computers container and sets the value of this entry point equal to `cn=BMC CM Computers, ou=BMC CM, ou=BMC Software`. This arrangement keeps the created entries separate from other computer entries, which are typically used for login, printer access, and other purposes in addition to software distribution. BMC Marimba Client Automation components are designed to take advantage of existing Active Directory infrastructure, but do not cause conflicts or interrupt normal activity.

In Sun Java System Directory Server (Sun One) Directory, Report Center also creates machine accounts if it cannot find them. In that case, Report Center creates machines in the same container in which it searches for them; by default, `ou=machines`. The reason Report Center searches and creates entries in the same container is that in Sun ONE Directory, all entries in the container are BMC entries to start with, and can be manipulated as needed without concern for interfering with other uses of the directory service. ADAM / AD LDS creates machine entries in a similar manner.

The Sun ONE Directory implementation of Policy Manager also supports machine creation using the `machines.txt` flat-file import feature. This feature enables you to import machine definitions, either manually prepared or exported from 4.x releases of Marimba Subscription. When you import a machine definition, Policy Manager stores it in the container you specify in the property `marimba.subscriptionplugin.machineimportgroupbase`.

The directory service settings specified in CMS are used by both Report Center and Policy Service. This means that the directory service you specify in CMS is the one Report Center uses to store collection results and the one with which Policy Manager communicates in order to display targets.

When a machine account must be created, Policy Manager uses the login name of the user configured in the Directory Services page in the system settings. This approach is necessary because collections can be refreshed even if no one is actively logged in to CMS. For details on setting permissions, see “Setting up user accounts” on page 65.



Section

# III Using Policy Manager

Part 2 discusses the following topics:

- “Obtaining user and group information from a transmitter” on page 105
- “The subscription configuration object” on page 111
- “Policy Service configuration and implementation” on page 125
- “Configuring Policy Manager” on page 145
- “Viewing targets and packages” on page 159
- “Creating and editing policies” on page 207
- “Integration with Patch Management” on page 295
- “Integration with Security Policy Manager Remediation Groups” on page 313
- “Viewing policy compliance” on page 317
- “Integrating with Deployment Manager” on page 341



# 6 Obtaining user and group information from a transmitter

This chapter provides information about configuring Policy Management to use a transmitter as the source for users and user groups.

The following topics are provided:

- Overview: Using a transmitter as the source for users and groups (page 106)
- Issues when sourcing with transmitters (page 108)
- Limitations when sourcing with transmitters (page 109)

## Overview: Using a transmitter as the source for users and groups

Policy Manager allows both computers and users to be targeted by policies. Users and user groups are typically obtained from the same directory service that is used to store Policy Manager objects. This scenario is very common in Active Directory implementations because Policy Manager can take advantage of pre-existing domain users and groups.

There are, however, situations in which the directory service that is used to store policies does not contain the corporate user and group database, and is also not managed by the same functional team as that responsible for software distribution. An example of such a scenario is when a Sun Java System Directory Server (Sun One) Directory infrastructure has been deployed for Policy Management, but the user and group information needs to be obtained from an existing Windows NT 4 Domain Controller. To achieve this end, Policy Manager is configured to obtain its user information from the transmitter's authentication server connection. This connection is configured using Transmitter Administrator. See the appendix in the *BMC Marimba Client Automation Transmitter and Proxy User Guide*, available on the BMC Customer Support website.

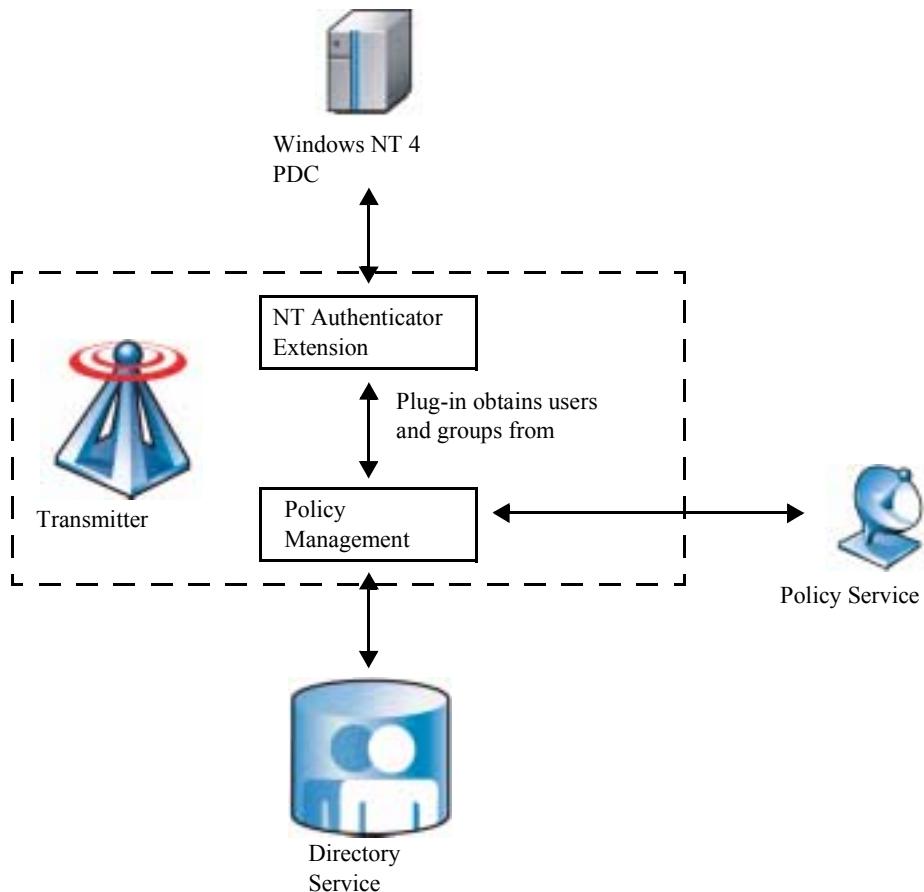
The following property in the Subscription configuration object configures the Policy Manager to use either the directory service or the transmitter to obtain users and groups.

Table 6-1: The marimba.subscriptionplugin.usetransmitterusers configuration property

Configuration property	Value
marimba.subscriptionplugin.usetransmitterusers	true to obtain users from the transmitter; false to obtain users from the directory service.

The following diagram illustrates how the various BMC Marimba Client Automation components are arranged when obtaining users from the transmitter, which in turn obtains users and groups from a Windows NT 4 Domain Controller.

Figure 6-1: Obtaining users from the transmitter



The sections that follow explain the setup required when obtaining users and groups from the transmitter.

## Issues when sourcing with transmitters

BMC recommends that you decide whether to obtain users and groups from the transmitter or from a directory service at installation time. When you generate scripts for installing the schema, make sure you select the Source users from a transmitter option (as described in the *BMC Marimba Client Automation Installation Guide*, available on the BMC Customer Support website).

If you want to configure Policy Manager to obtain users and groups from the transmitter after the product has been installed, you can do so by setting the configuration property

`marimba.subscriptionplugin.usetransmitters=true`. Follow the instructions described in “Editing attributes of a configuration object” on page 116. Note that you must restart Policy Manager and republish the plug-in. Also, if you take this route, any policies that have been created when users and groups were obtained from the directory service cannot be used, and must be deleted.

**Configuring the Policy Management plug-in to obtain user information from the transmitter.** You identify the source of user and group information when you configure and publish the Policy Service plug-in, either from the CLI or the GUI. If you want to obtain this information from the transmitter, specify the transmitter URL when you publish the plug-in.

**Providing transmitter authentication access.** If the transmitter has been configured to require a user name and password for authentication, you must furnish these credentials to Policy Manager by running `-txadminaccess` from the command line. See “`-txadminaccess <user_name> <password>`” on page 399. If the transmitter has been set up for password-only authentication, specify an asterisk (\*) for `<username>`.

**Entry points when obtaining user information from the transmitter.** When the system is configured to obtain users from the transmitter, Policy Manager and the Policy Service plug-in will browse the user list from the transmitter. The transmitter obtains the user list from its local user database, a directory service, or a custom authentication extension specified using Transmitter Administrator.

**Repeater configuration when obtaining user information from the transmitter.** Each BMC Marimba Client Automation repeater in the network must be configured to obtain users and user groups from the same source as the master transmitter. This information is not included when the repeaters are replicated from the master transmitter. Use Transmitter Administrator to configure each repeater so that it obtains user and user group information from the same location as the master transmitter. Note that after you reconfigure each repeater, you will have to stop and restart it.

## Limitations when sourcing with transmitters

Obtaining user and user group information from the transmitter gives you the flexibility to use a legacy system to provide this information. However, Policy Manager cannot predict exactly what information the legacy system will be able to provide. Therefore, the listing of users and user groups returned from the legacy system to Policy Manager may not be in distinguished name format. Instead, the name returned is the `name` attribute specified in Transmitter Administrator. In Transmitter Administrator's Security > Administration Access tab, select Custom Authentication Extension from the Access list source list, and then click Browse custom users.

For example, consider a user in an existing Windows NT 4 Domain Controller. The user's fully qualified name is `angela\NT-Domain`, where `NT-Domain` is the name of the domain to which `angela` belongs. In the list returned to Policy Manager, only `angela` will be displayed.

To use a custom authentication extension (such as NT Domain Authenticator), you must first publish it to the transmitter. You can use the `groups.in.domain` and `users.in.domain` properties in the `parameters.txt` file to specify that the domain names should be included. See the appendix in the *Infrastructure Administrator's Guide*, available on the BMC Customer Support website.

**Case-sensitivity when using the transmitter's local user database.** If you use a transmitter as the source for users and groups, and the users and groups are from the transmitter's local user database, Policy Manager is case-sensitive with regard to user names. If the transmitter is using a directory service (not the local user database) as the source for users and groups, then case-sensitivity is not an issue.



# 7 The subscription configuration object

Directory services store and display entries in a hierarchical tree structure, with each entry designated by a unique *distinguished name*, or DN. Although this arrangement offers a flexible way to browse entries, directory services are not designed for easy browsing when the number of entries becomes large. Obtaining good performance from an integrated directory management application such as Policy Manager requires some care setting up directory containers and limiting views into the directory. The location of all directory objects, and the scope of browsing and displaying policy targets, is determined by the settings in a special object called the Subscription configuration object.

The following topics are provided:

- Attributes of the subscription configuration object (page 113)
- The BMC Marimba Client Automation configuration object (page 114)
- Attributes of the configuration object (page 115)
- Editing attributes of a configuration object (page 116)
- Entry point for Policy Management policy objects (page 117)
- BMC Marimba Client Automation computer entries created by Report Center collections (page 117)
- Schema mapping configuration parameters (page 118)
- Example of a customized schema mapping configuration (page 119)
- Collections configuration parameters (page 121)
- Miscellaneous configuration parameters (page 122)

A directory service is a shared network resource that is used by many systems and applications. For that reason, directory administrators may not want an individual who is solely responsible for software distribution to be able to access the entire corporate directory. Another reason to limit a software distribution administrator's view of the directory is to reduce confusion by showing only those groups and targets that are relevant. For example, e-mail distribution lists can be hidden because they typically are not used for targeting software distribution.

The default Policy Manager installation creates containers that are used to store BMC Marimba Client Automation objects. This arrangement is usually satisfactory for directory systems dedicated to software distribution, but may be a problem with directories that service other applications and services, as is typical with Active Directory implementations. In these instances, you may want to place all Policy Manager containers in a separate container so that you can partition all BMC Marimba Client Automation functionality to one master container in the directory.

---

Note: Unlike other Policy Manager containers, the location of the Subscription configuration object cannot be moved.

---

The Subscription configuration object is located in the ConfigObjects container, a child of the BMC Marimba Client Automation container. It is of the object class `SubscriptionSubscription` (Active Directory or ADAM / AD LDS) or `mrbasubscription` (Sun Java System Directory Server (Sun One) Directory).

The Subscription configuration object contains Policy Management configuration information. These attributes can be edited during the initial LDIF script installation process. Later, these attributes can be edited using Policy Manager commands. See the instructions in “Editing attributes of a configuration object” on page 116. Note that you must republish the Policy Service plug-in after modifying attributes of the Subscription configuration object. You can also use the directory service vendor’s LDAP editing tools (or those of a third party). If you modify the configuration, you must restart both the CMS (Common Management Services) channel and Policy Manager before the changes can take effect.

# Attributes of the subscription configuration object

This section describes some editable attributes of the Subscription configuration object (CN=Subscription Config). For more information on editing attributes, see “Editing attributes of a configuration object” on page 116.

- `marimba.subscription.acl`—specifies whether the ACL and permissions feature is turned on or off (true or false). The default value is `false`. See “Setting up access control lists” on page 153.
- `marimba.subscriptionplugin.mode`—specifies if the Policy Service plug-in is in online or offline mode. The default is online mode. In offline mode, the plug-in does not process requests for policies from the endpoints. A log entry appears in the plug-in log indicating that the plug-in is not processing requests. Possible values are online or offline.
- `marimba.subscriptionplugin.overrideclientdn`—when set to `true`, the full DN sent up by Policy Service is ignored. The plug-in domain is used instead. Use this option when policies are stored in a domain other than the one where the machines and users are stored. This option applies only to Active Directory. The default value is `false`.
- `marimba.subscriptionplugin.requireentriesinldap`—when set to `true`, policies (including those assigned to All Endpoints) will not be sent to endpoints that do not have entries in the directory service. If this property is set to `false` or is not present (default), policies assigned to All Endpoints will be sent down, even if the machine is not present in the directory service.
- `marimba.subscriptionplugin.resolvetype`—specifies that group and policy resolution are performed only for users (if set to `user`) or machines (if set to `machine`). If this property is not set or is set to a value other than `user` or `machine`, the default behavior is to perform group and policy resolution for both users and machines. You might want to set this property if your enterprise uses either user-based or machine-based policies only. Because the plug-in only need to resolve groups and policies for one target type (user or machine), performance of the system improves.

- `marimba.subscriptionplugin.subscriptionbase`—specifies the path to the Subscriptions container, which stores the domain controller information. Policy Manager uses this information in order to resolve policies for a target. If you change the name or location of the Subscriptions container, you must edit this property to point to the new location.
- `marimba.subscriptionplugin.usednfromclientonly`—when set to true, policies will not be sent to endpoints that do not have fully qualified names.

## The BMC Marimba Client Automation configuration object

Like the Subscription configuration object, the BMC Configuration Automation for Clients configuration object (`cn=BMC CM Config`), located in the `ConfigObjects` container in the directory service, contains configuration information for Policy Manager. It also contains configuration information for BMC Configuration Automation for Clients components other than Policy Manager.

---

Note: In previous releases, BMC called the BMC Configuration Automation for Clients configuration object the Marimba configuration object.

---

The BMC Configuration Automation for Clients configuration object is of the object class `MarimbaProperties` (Active Directory or ADAM / AD LDS) or `mrbaproperties` (Sun ONE Directory). Currently, it contains an attribute that specifies the location of the ACLs container, such as  
`marimba.aclbase=ou=Acl,ou=ConfigObjects,ou=BMC CM,ou=BMC Software,dc=company,dc=com`. This attribute is included in the BMC Configuration Automation for Clients configuration object because BMC is reserving it for future use by BMC Configuration Automation for Clients components other than Policy Manager.

# Attributes of the configuration object

This section describes some editable attributes of the BMC Marimba Client Automation configuration object. For more information on editing attributes, see “Editing attributes of a configuration object” on page 116.

- `marimba.ldap.browse.collectionmachinebase`—specifies the path to the BMC Computers container, which stores any machines that Report Center creates when it generates a collection. If you move the BMC Computers container out of the BMC container (the default location), you must edit this property to point to the new location. By default, the value is `cn=BMC CM Computers,ou=BMC CM,cn=BMC Software,dc=company,dc=com`.
- `marimba.ldap.browse.hideentries`—specifies the containers that you do not want to display in Policy Manager. You can specify a comma-separated list of DNs, with each DN enclosed by double quotation marks, such as “`OU=Subscriptions, OU=ConfigObjects, OU=BMC CM, OU=BMC Software, DC=company, DC=com`”, “`OU=Acl, OU=ConfigObjects, OU=BMC CM, OU=BMC Software, DC=company, DC=com`”. You can only hide OUs, CNs, and DCs. By default, the Subscriptions, Acl, and BMC containers are hidden. Changing the value of this attribute requires restarting Policy Manager.
- `marimba.schemaversion`—specifies the LDAP schema version. Do not modify this attribute.
- `marimba.ldap.browse.collectionmode`—specifies the collection mode. Your options are centralized (the default), and distributed. For a thorough discussion of collection modes, see the *BMC Marimba Client Automation Report Center User Guide*, available on the BMC Customer Support website.
- `marimba.ldap.browse.collectionbase`—specifies the path to the Collections container, which stores any collections you create by using Report Center. If you change the name or location of the Collections container, you must edit this property to point to the new location.
- `marimba.ldap.browse.machineclass`—specifies the class for searching machine objects.
- `marimba.schemapatchversion`—specifies the patch level of the schema, which increments the default (matching the base BMC Marimba Client Automation version) if you make changes to containers.

# Editing attributes of a configuration object

If you decide to change the name or location of containers used by the Policy Management module, you will need to edit one or more attributes of the Subscription or configuration object. This section provides instructions for using the command-line interface to edit attributes. For more information about the attributes, see “Attributes of the subscription configuration object” on page 113, or “Attributes of the configuration object” on page 115.

## ► To change an attribute of a configuration object

- 1 On the command line, enter the following command:

```
runchannel <PolicyManager_URL>
-user <user_name> -password <password>
-configSet <key> <value> [-preview}
```

where:

<PolicyManager\_URL>

is the URL of the Policy Manager channel, such as `http://mycompany:5282/SubscriptionManager`.

<user\_name> and <password>

is the name and password for a user in the directory service.

<key> <value>

is the key name and value for the attribute in the Subscription configuration object that you want to change.

[-preview}

allows you to view the new and old values for the attribute. Note that although the command-line interface might tell you it succeeded, it has not made the attribute change. You must run the command without `-preview` to actually make the change.

Example:

```
runchannel http://mycompany:5282/SubscriptionManager
-user marimbaUser -password opensesame
-configSet marimba.subscription.acl true
```

This command allows you to turn on the ACL feature by setting the value of the `marimba.subscription.acl` attribute to `true`.

- 2 Restart Policy Manager as follows: In System Settings, go to the General > Applications Manager tab, stop Policy Manager, and then start it again.
- 3 Republish the Policy Service plug-in. For instructions, see “Configuring and publishing the Policy Service plug-in” on page 146.

## Entry point for Policy Management policy objects

Policy Management policies are stored using a BMC -specific LDAP object whose class is `mrbasubscription` in the Sun Java System Directory Server (Sun One) Directory implementation and `Marimba-Com-1996-Castanet-Subscription-Subscription` in Active Directory and ADAM / AD LDS. See “`marimbaCom1996-Castanet-SubscriptionSubscription`” on page 39. These objects are stored in a special container, `marimba.subscriptionplugin.subscriptionbase`, which is defined as follows:

Table 7-1: The `marimba.subscriptionplugin.subscriptionbase` configuration property

Configuration property and description	Sun ONE directory default*	ADAM / AD LDS default*	Active Directory default*
<code>marimba.subscriptionplugin.subscriptionbase</code>			
Location of policy objects	<code>ou=Subscriptions</code>	<code>ou=Subscriptions</code>	<code>cn=Subscriptions</code>

\* Distinguished names for default containers have been truncated, the actual entries also contain the directory base distinguished name (suffix) defined when the Policy Manager installation script was created.

## BMC Marimba Client Automation computer entries created by Report Center collections

The Report Center application creates machine groups called *Collections* whose members are based on the results of a query of the Inventory database. When the database query returns computer host names that Report Center cannot find in the directory service, Report Center creates computer objects for them in the directory service. See “Setting up collections” on page 81 for details of the implementation.

# Schema mapping configuration parameters

LDAP- and X.500-based Directory systems have established standards for naming conventions and placement of objects within the directory. However, they do not have standards for the object types and attributes used to represent entries such as users and groups. For example, the object class used to represent users in Sun Java System Directory Server (Sun One) Directory is called `inetOrgPerson`, but the object class used to represent users in Active Directory is called `user`. Similarly, the attribute used to represent a user's login ID in Sun ONE Directory is called `uid`, but is called `sAMAccountName` in Active Directory.

The Subscription configuration object and the BMC Marimba Client Automation configuration object contain parameters that define the object class names and attribute names that Policy Manager will interpret as users, machines, and groups. They are used to map Policy Manager to different directory implementations. Typically, you will need to modify these settings if you want to use a custom sub-classed object rather than the standard objects that come pre-installed with the directory service.

The following table describes the schema mapping attributes and their defaults.

Table 7-2: Schema mapping attributes

Configuration parameter	Description	Sun ONE directory default	Active Directory default
<code>marimba.subscriptionplugin.userclass</code>	Name of the class used to represent user entries.	<code>inetOrgPerson</code>	<code>user</code>
<code>marimba.subscriptionplugin.useridattr</code>	Name of the attribute used to represent user IDs.	<code>uid</code>	<code>sAMAccountName</code>
<code>marimba.subscriptionplugin.groupclass</code>	Name of the class used to represent groups.	<code>groupOfNames</code> , <code>groupOfUniqueNames</code> <sup>1</sup>	<code>group</code>
<code>marimba.subscriptionplugin.groupnameattr</code>	Name of the attribute used to identify group objects.	<code>cn</code>	<code>cn</code>

Table 7-2: Schema mapping attributes

Configuration parameter	Description	Sun ONE directory default	Active Directory default
marimba.subscriptionplugin.groupmemberattr	Name of the attribute used to identify group members.	member, uniqueMember <sup>1</sup>	member
marimba.subscriptionplugin.machineclass (Stored in the BMC configuration object)	Name of the class used to represent machine (computer) objects.	mrbamachine	computer
marimba.subscriptionplugin.machinenameattr	Name of the attribute used to identify machines (computers).	cn	cn

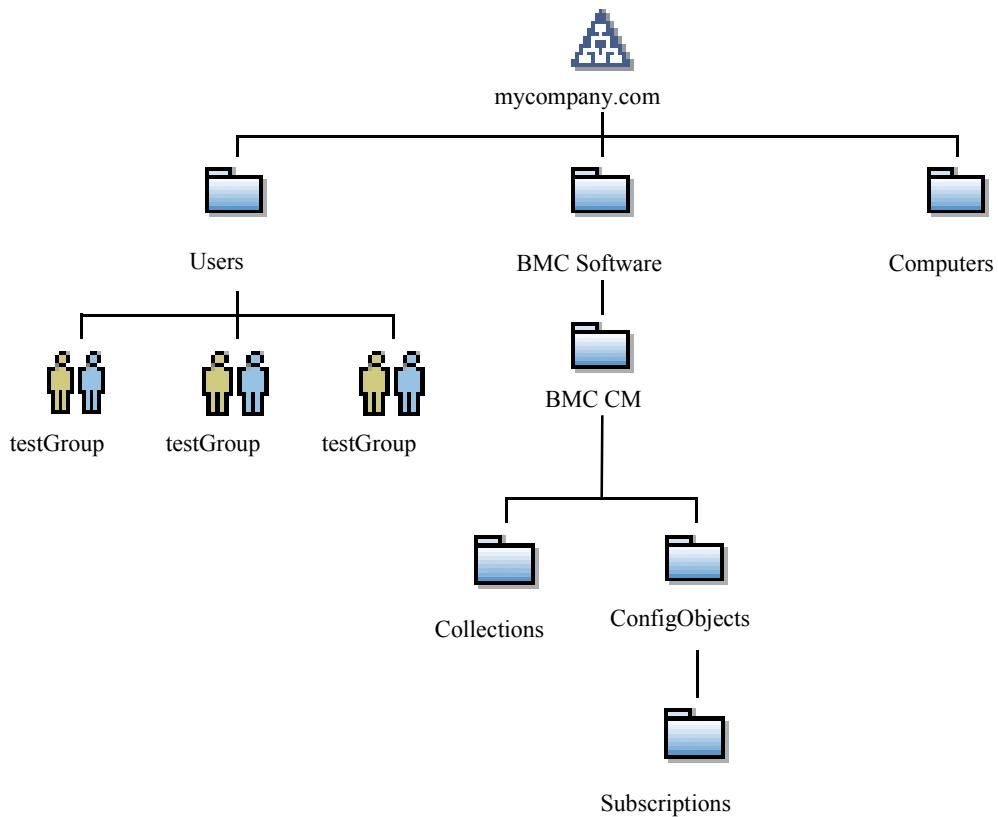
1. A comma-separated list, such as groupOfNames, groupOfUniqueNames, indicates that either class name can be used.

## Example of a customized schema mapping configuration

You can set permissions on targets so that only administrators who are meant to set and view policies on those targets have access to them. See “Setting up user accounts” on page 65. In addition to this level of control, it is possible to customize the schema to control administrators’ views.

In the following example, an administrator has subclassed the `group` object class to create a subclass called `testGroup`, solely for the purpose of software distribution. By setting this class name for the custom group as part of the schema mapping configuration, the administrator can filter out all ordinary `group` objects from the Policy Manager targets browser.

Figure 7-1: Customized schema mapping configuration—subclassing the “group” object



The following configuration entry would be added to the Subscription configuration object to enable the sub-classed groups testGroup to be recognized by Policy Manager.

Table 7-3: The marimba.subscriptionplugin.groupclass configuration property

Configuration property	Value
marimba.subscriptionplugin.groupclass	testGroup

# Collections configuration parameters

Collections configuration parameters specify where and how collections are stored. The Subscription configuration object and the BMC Marimba Client Automation configuration object contain these parameters, and the following table describes them.

Table 7-4: Collections configuration parameters

Configuration property	Description
<code>marimba.ldap.browse.collectionbase</code> (Stored in the BMC Marimba Client Automation configuration object)	Specifies the path to the Collections container, which stores any collections you create by using Report Center. If you change the name or location of the Collections container, you must edit this property to point to the new location. By default, it is <code>ou=Collections,\$Suffix</code> .
<code>marimba.ldap.browse.collectionmachinebase</code> (Stored in the BMC Marimba Client Automation configuration object)	Specifies the path to the MarimbaComputers container, which stores any machines that might be automatically created by Report Center when it creates a collection. If you move the MarimbaComputers container outside of the Computers container (the default location), you must edit this property to point to the new location. By default, it is <code>cn=MarimbaComputers,cn=Computers,\$Suffix</code> .
<code>marimba.ldap.browse.collectionmode</code> (Stored in the BMC Marimba Client Automation configuration object)	Specifies the collection mode, which is either centralized or distributed. For Active Directory, the LDIF scripts that you run during installation specifies distributed as the default; if upgrading from a previous release, you can specify the mode during the upgrade process (see the upgrade section of the <i>BMC Marimba Client Automation Installation Guide</i> ). For Sun ONE Directory, the default is centralized. For a thorough discussion of collection modes, see the <i>BMC Marimba Client Automation Report Center User Guide</i> .
<code>marimba.subscriptionplugin.usednfromclientonly</code> (Stored in the Subscription configuration object)	When set to true, resolves policies for targets sending unique identification only. By default, it is false.

# Miscellaneous configuration parameters

The Subscription configuration object contains several other configuration parameters. The following table describes these parameters.

Table 7-5:

Configuration property	Description
<code>marimba.subscriptionplugin.usetransmitterusers</code>	Determines the source for user and group information. See “Obtaining user and group information from a transmitter” on page 105.
<code>marimba.subscriptionplugin.useglobalcatalog</code>	When true, instructs the Policy Service plug-in to use the Global Catalog in an Active Directory environment. See “The global catalog” on page 48.
<code>marimba.subscriptionplugin.globalcatalogbase</code>	Determines the entry point for the Global Catalog in an Active Directory multidomain environment. See “The global catalog” on page 48.
<code>marimba.subscriptionplugin.logs.roll.policy</code>	<p>Specifies the policy for rolling the log files for Policy Management. You can set it to one of the following values:</p> <ul style="list-style-type: none"> <li>▪ <code>hourly</code></li> <li>▪ <code>daily</code></li> <li>▪ <code>weekly</code></li> <li>▪ <code>monthly</code></li> <li>▪ <code>yearly</code></li> <li>▪ <code>manually</code></li> <li>▪ <code>never</code></li> <li>▪ <code>bysize</code> (Default)</li> </ul> <p>If you choose <code>manually</code> or <code>never</code>, the log entries are recorded in a single file, which is never automatically rolled.</p> <p>If you choose <code>bysize</code>, the log file is rolled automatically when it reaches the size specified in “<code>marimba.subscriptionplugin.logs.roll.size</code>”.</p>
<code>marimba.subscriptionplugin.logs.roll.versions</code>	Specifies the number of previously rolled log files that can exist. By default, it is 1.

Table 7-5:

Configuration property	Description
marimba.subscriptionplugin.logs.roll.size	Specifies the size, in kilobytes, that the log file must reach before it is rolled automatically. The default value is 32 KB. The value of this property is used when “marimba.subscriptionplugin.logs.roll.policy” is set to bysize.
marimba.subscriptionplugin.usednfromclientonly	When set to true, resolves policies for targets sending unique identification only. By default, it is false.



Chapter

# 8

# Policy Service configuration and implementation

This chapter provides insight into Policy Service configuration and implementation.

The following topics are provided:

- Policy Management state verification and retry (page 126)
- Client machine name (page 127)
- Unique identification of targets (page 127)
- Policy Service and schedule enforcement (page 128)
- User-controlled packages (page 139)

Policy Service applies a received policy to the endpoint where it is running. Policy Service is configured to start at scheduled intervals, but stops itself after it applies all package states and properties in the policy.

# Policy Management state verification and retry

After a policy has been applied, the Policy Service at the endpoint verifies that each package that should be subscribed is indeed subscribed. Policy Service performs its verification by querying the package's workspace directory, not by tracking error conditions as the package is being subscribed. If Policy Service detects a failure, it schedules a retry operation in 60 seconds, by default. Policy Service then tries five times (by default) to apply the policy.

Both the retry time and retry count can be configured using tuner properties (see “[marimba.subscription.retrycount](#)” on page 277 and “[marimba.subscription.retrytime](#)” on page 277).

For packages that were created with the Application Packager and configured for automatic installation, Policy Service verifies the state by looking in the package's application.txt file. If the property `adapter.installed` is true, Policy Service assumes that the operation succeeded.

---

Note: Policy Service verifies states only, not actions that are meant to take place after a policy has been successfully applied. For example, if the policy imposes an `install_start` state, Policy Service does not actually verify that the process (if one is defined for the package) forks off correctly. A summary of package states and whether they are verified (to determine if a retry is warranted) is shown in the following table:

---

Table 8-1: Verification of state for automatic Policy Management retry

Policy Management state	Retry verification
Advertise	No verification
Exclude	No verification
Install	Verification
Install-Persist	Verification
Install-Start	Verification
Install-Start-Persist	Verification
Primary	Verification
Stage	Verification
Uninstall	No verification

## Client machine name

By default, Policy Service uses the Java method

`InetAddress.getLocalHost().getHostName()` to determine the machine name used by Policy Management. The string returned by this method is truncated to remove the domain name (if present). On Windows platforms only, the Windows NetBIOS Name can be used as the machine name returned by Policy Service by setting the tuner property `marimba.subscription.usecomputername=true`.

You can override the machine name by setting the tuner property `marimba.subscription.machinename`. You typically override the machine name when you do not have direct control over the host name of the endpoint. For example, if you are managing a number of business partners or customers that have already established their own naming schemes for endpoints. Overriding the machine name allows each endpoint to be given a non-conflicting organized name, even if two pre-designated names are identical. For example, if you were providing IT services to a group of automobile dealerships, you could use a dealer brand and number to distinguish each endpoint.

## Unique identification of targets

For users and machines with non-unique names, Policy Service obtains the fully-qualified login name of the user and machine through ADSI. Users who log into a machine from another domain will receive their user-based policies. If there is no domain information for a target, the default will be the Policy Service plug-in's domain.

If you want to resolve policies for targets sending unique identification only, set the following attribute in the Subscription configuration object to true:

`marimba.subscriptionplugin.usednfromclientonly`

When this attribute is set to true, policies will not be sent to endpoints that do not have fully-qualified names. Its default value is false.

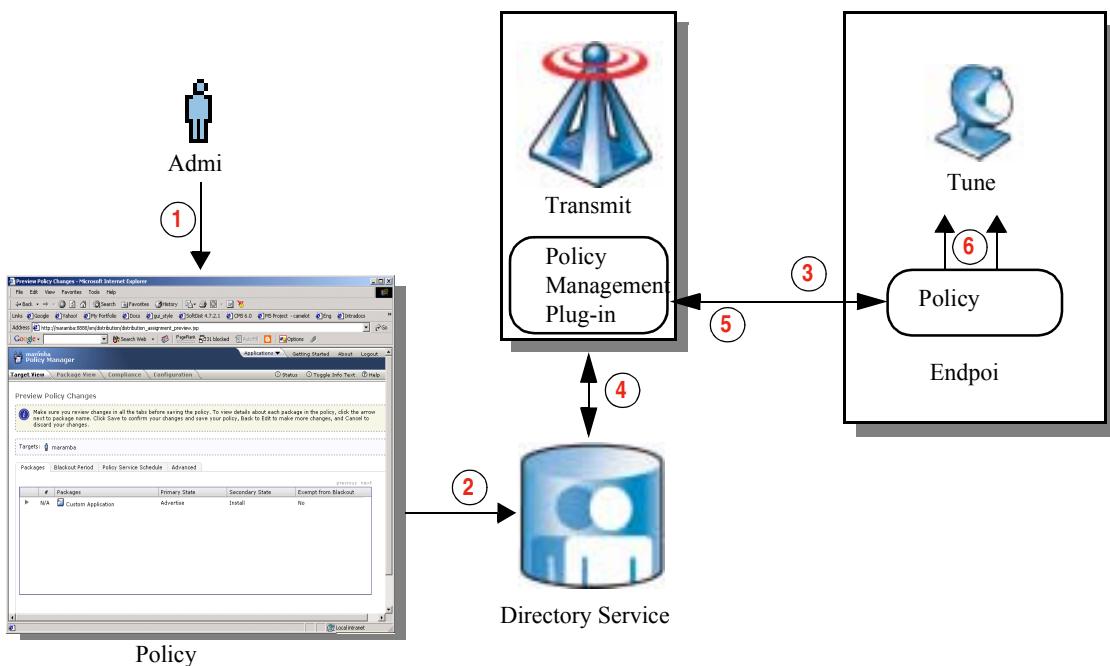
# Policy Service and schedule enforcement

The client portion of Policy Service is responsible for enforcing the schedules you set in Policy Manager. The process of setting and enforcing schedules involves all parts of Policy Management. The interaction between Policy Management components is illustrated in Figure 8-1 on page 129.

► **The actions illustrated in the figure are as follows**

- 1 An administrator enters schedule information as part of a policy in Policy Manager.
- 2 Policy Manager accepts the schedule information and stores it as part of a policy in a directory service.
- 3 Policy Service, running on an endpoint, connects to the plug-in.
- 4 The Policy Service plug-in, running on the transmitter, connects to the directory service, retrieves the policy, and sends it back to the Policy Service running on the endpoint.
- 5 Policy Service processes all events that need to occur immediately.
- 6 Policy Service determines when it should execute the next event in the schedule and requests the tuner to wake up at the correct time.

Figure 8-1: Creating and enforcing package schedules



The Policy Service activities at the endpoint take place in steps 5 and 6. The following sections describe these activities.

## Processing immediate and scheduled events

In earlier versions of Policy Management, Policy Service applied all of a policy as soon as it was received. In the current version, the enforcement of package state transitions, update schedules, and repair schedules can be scheduled to occur at times you specify. However, package and tuner properties are still applied immediately when the Policy Service receives them as part of a policy.

► **When Policy Service receives a policy, it initiates the following chain of events**

- 1 **Flush all knowledge of the previous policy.** The previous policy has no bearing on the processing of the current one.

- 2 **Resolve conflicts in the policy.** Conflicts can be created when the same package(s) are assigned to the same endpoint. This situation can occur when an endpoint is a member of several target groups, and those target groups are assigned the same packages. Conflict rules are applied, and only one instance of each package is kept. See “State precedence” on page 217, “Conflict resolution: When multiple users edit properties” on page 292, and “Conflict resolution: states and schedules in policies”.
- 3 **Apply all the tuner properties in the policy.**
- 4 **Set the tuner blackout period.** The blackout schedules from all assignments in the policy are inspected and used to construct a composite set of tuner blackout periods.
- 5 **Verify that the current policy has no errors.** The policy is checked against a set of schedule rules. If errors are found, such as if a schedule’s expiration time occurs before its activation time, a log message is written for each error and Policy Service exits.

**Note:** This error checking is only required because administrators can write directly into the directory service where policies are stored. Policy Manager will not let you save invalid policies to the directory service.

- 6 **Determine the next time of execution.** Policy Service inspects the policy and determines which packages have activities scheduled at a later time. It then calculates the next time of execution for these packages and sorts them in time order. It stores the earliest time as the time when the policy needs to be processed again.
- 7 **Apply the current policy.** Policy Service inspects the policy and applies the required state changes, updates, and verifications/repairs. The current time is used to inspect each package’s state transition schedule in the policy. If a package state transition should be applied at the current time, the transition is attempted. If a package update or repair is due, it is executed.
- 8 **Apply all the package properties in the policy.**
- 9 **Instruct the tuner to run Policy Service when the policy needs to be processed again.** After the list of times has been generated, the list is sorted and the time of the next event is used to instruct the tuner when next to start Policy Service.
- 10 **Policy Service exits.** When the tuner next starts Policy Service, it again processes the policy (as described in the previous steps).

## Assigned states and endpoint states

Before further describing the effects of scheduling in Policy Management, it is useful to review the concepts of package states, and to draw a distinction between the state you request in Policy Manager (the assigned state) and the actual state of a package at a targeted endpoint.

The assigned state is the directive provided by Policy Manager that attempts to produce an outcome for the package at the endpoint, and can be any of nine states. At the endpoint, a package can be in only one of four states. The situation is summarized in the following table:

Table 8-2: State assignment and resulting package states

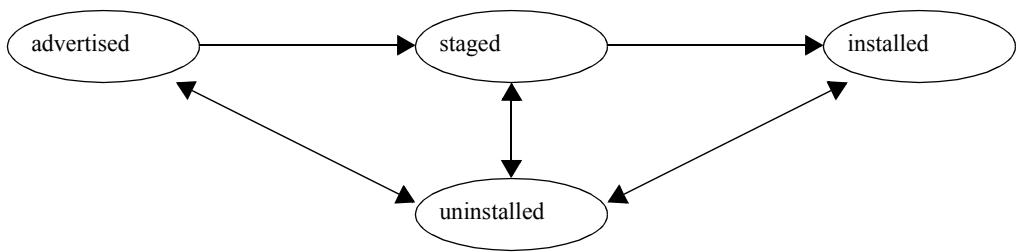
Assigned state	Endpoint state after enforcement
advertise	advertised
exclude	not present
install	installed
install-start	installed (and started)
install-persist	installed
install-start-persist	installed (and started)
primary	installed
stage	staged
uninstall	uninstalled, or not present

A package can be in an installed state on an endpoint if it has been installed by Policy Management or by other means, such as the user manually subscribing at the endpoint.

## Endpoint state transitions

At the endpoint, a package can only progress through states as shown in the following figure:

Figure 8-2: Package state transitions on endpoints



After a package has been placed in a *staged* state, it can no longer be *advertised*. After a package has been *installed*, it can no longer be *staged*. To move backward through the state order, you must remove the package from the endpoint (uninstall it). The package can be uninstalled from any of the three states.

For more information about primary and secondary states, see “Overview of installation states” on page 215.

## Types of schedules

Schedules are associated with packages that you assign to targets and are enforced at the Policy Service on each endpoint. Schedules are referenced to the endpoint’s local time, not to the local time of the Policy Manager.

You can set four types of independent schedules in Policy Manager. The schedules are:

- Primary
- Secondary
- Update
- Repair

Primary and secondary schedules are enforced between activation and expiration times. That is, you can specify that an activity will begin at a certain date and time (activation time), and you can specify another date and time when the activity will no longer occur (expiration time).

In addition to activation and expiration times, update and repair schedules can also be recurring. For example, you can specify that a package updates every two hours during a specified period.

## Life cycle of a package schedule

Package schedules are independent. For example, you do not need to set a Primary schedule in order to set a Secondary schedule.

Note the following relationships between schedules and package states:

- Update schedules and repair schedules can only be applied to packages that have been installed or staged on an endpoint, and are ignored otherwise.
- If you assign a Secondary state in Policy Manager, then you must also assign a Secondary schedule.
- The assigned primary state of a package does not require a primary schedule.

For example, assume that you have set a primary schedule for an *advertise* state, but you have not set a secondary schedule. You can still set an update schedule for the package, although the update schedule in this case will be meaningless, and no updates will occur for a package in the *advertised* state. In contrast, staged packages, while not yet installed, are present at an endpoint, and can be updated. See “Updating staged packages” on page 138.

Similarly, you cannot repair a package unless it has been installed (and was prepared with Application Packager). Therefore, a package that is in the *advertised* state will not be verified and repaired even if you have set a repair schedule.

---

Note: If a user has installed a package on an endpoint independent of Policy Management, and that same package has been given an update (or repair) schedule in a policy for the endpoint, then the update (or verification and repair) will occur as scheduled.

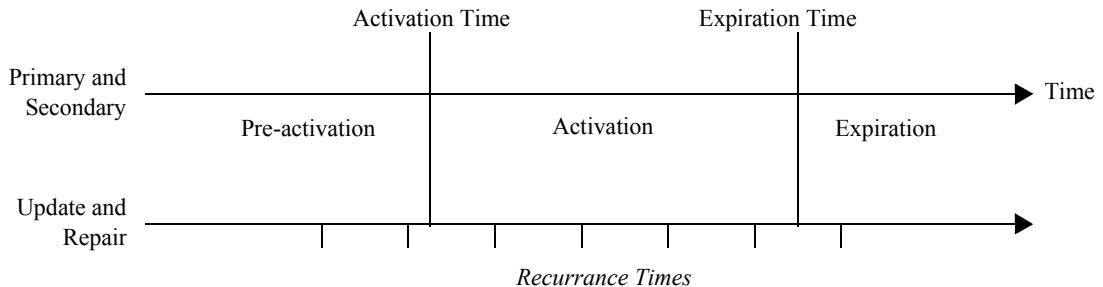
---

A package schedule can be divided into the following periods:

- pre-activation
- post-activation
- post-expiration

Although the following figure maps all the schedules on the same time line, keep in mind that each of the four schedules is independent—each activation time, expiration time, and recurrence time can be different.

Figure 8-3: Activation, expiration, and recurrence times for the four types of schedule



## Primary and secondary schedules

Primary and secondary schedules have activation and expiration times. Prescribed activities cannot occur before the activation time, nor after the expiration time. The concept is best explained through an example.

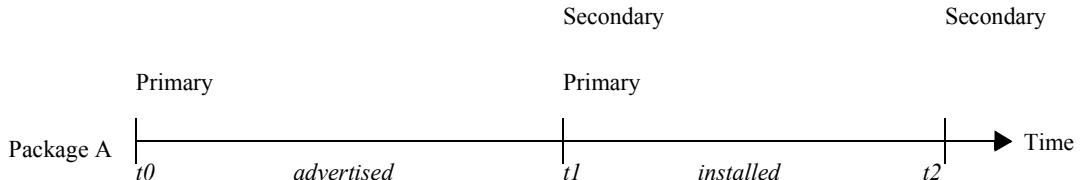
### Example

Assume that you have set a primary schedule for Package A that activates at time  $t_0$  and expires at time  $t_1$ . The requested state for Package A during this time is *advertise*. If Policy Management performs successfully, Package A will be advertised on the targeted endpoint between times  $t_0$  and  $t_1$ .

Assume further that Package A has a secondary schedule state of *install* between times  $t_1$  and  $t_2$ . Then, when the Policy Service at the endpoint updates during this period, Package A will be installed.

After time  $t_2$ , Policy Service will not try to force any state on the package. This means that the end user can delete the package, and Policy Service will not attempt to reinstall it. Policy Management maintains control of the package only between times  $t_0$  and  $t_2$ .

Figure 8-4: Primary and secondary schedules




---

Note: Because of the state transition rules described in “Endpoint state transitions” on page 131, applying an *advertise* or *stage* state to a package that has already been installed at an endpoint produces no change—that is, the package is not demoted from its installed state.

---

## Update and repair schedules

If Policy Service receives an update or repair schedule for a package as part of the policy, those schedules take precedence over any schedules that were set when the package was created.

Policy Service can install a package whether or not its policy specifies a repair or update schedule. If no schedule is specified by Policy Service, the package will update and repair on its own preset schedule, which was established when the package was created by Application Packager.

## Expiration

Schedules provided by Policy Management always take precedence over packaged schedules, even during expiration periods.

During the expired period of a package’s scheduled life, no action takes place, even if the package was provided a schedule at the time it was created. As a result, if a package that was created with a recurring update schedule is placed under Policy Management control, it will not update during the expiration period of the update schedule.

## Blackout schedules

The motivation for setting a blackout schedule usually is to eliminate possible interruptions to workday activities at the endpoints and to minimize network traffic during critical periods.

When Policy Service imposes a blackout period on a tuner, no scheduled activity can take place for any packages, even those not under Policy Management control.

End users can, however, perform manual updates during a blackout period. If a user updates Policy Service during a blackout period, the policy for the endpoint will be downloaded and enforced.

Channels can also run when directed by RPC connections or other programmatic means. Because the blackout disables the tuner's scheduler for the specified period, only scheduled events are affected.

---

Note: A blackout period cannot span midnight—that is, it cannot begin on one day and end the day after.

---

After a blackout period expires, schedules will immediately be reapplied to affected packages. As a result, any activities that were scheduled to take place as soon as possible during the blackout will occur immediately after the blackout period expires.

At the Daylight Savings Time changeover, the blackout period stays the same but the time moves forward or goes back one hour.

- Time advances one hour

If the blackout period ends during the hour after the time change, the blackout period is over when, at the moment of the time change, the time is reset to one hour later. Schedules are reapplied to affected packages.

For example, at 2.00 A.M., the time advances to 3.00 A.M. Even though the blackout period ends at 2.30 A.M., the blackout period is over when, at 2.00 A.M., the time is reset to 3.00 A.M.

- Time goes back one hour

If the blackout period ends during the hour before the time change, the blackout period is back in effect when, at the moment of the time change, the time is reset to one hour earlier. Schedules are not applied to affected packages until the blackout period is over.

For example, at 2.00 A.M., the time goes back to 1.00 A.M. Even though the blackout period ended at 1.30 A.M., the blackout period starts again when, at 2.00 A.M., the time is reset to 1.00 A.M.

## Varying the schedule

You can randomize the times that scheduled events occur by specifying a *vary time*. The vary time improves transmitter performance by spreading out client requests during periods of heavy load.

The default vary time is 10 minutes and you can change the time by setting the `marimba.subscription.varytime` tuner property through the GUI (although there is no pull-down menu for setting it). Set the value of this property in minutes, such as `marimba.subscription.varytime=15`.

When you specify a vary time, scheduled events are randomly postponed for a period up to the specified time. For example, an event scheduled to occur at 10:00 AM will, by default, occur anytime between 10:00 AM and 10:10 AM

The vary time applies to all packages controlled by Policy Service, but it does not apply to Policy Service itself.

---

Note: If Policy Service updates on an endpoint during the period specified by `marimba.subscription.varytime`, any channel that has been scheduled to update will do so immediately, and the vary time will be ignored. For example, if you schedule a channel to update at 10:00 AM and set the vary time to 30 minutes, you would expect the channel to update between 10:00 AM and 10:30 AM. However, if Policy Service happens to run its own update during this period, the channel will be updated immediately.

---

This behavior is typically unobjectionable and goes unnoticed unless the Policy Service updates are set to a far shorter time span than the vary time. For example, if Policy Service updates every five minutes and the vary time is set to 30 minutes, the effective vary time will be only 5 minutes.

## Policy Management control

At an endpoint, a package is said to be under Policy Management control if it is part of a policy that has been sent to that endpoint. Policy Management can seize control of a package that has already been installed simply by making it part of a policy.

Packages can, of course, be installed or deleted by other BMC Marimba Client Automation tools, such as Tuner Administrator. However, when the Policy Service next updates, it will attempt to apply all state transitions specified in the policy to the package.

If you want a package to revert to its as-packaged schedules, do not specify schedules in the policy you create for it in Policy Manager.

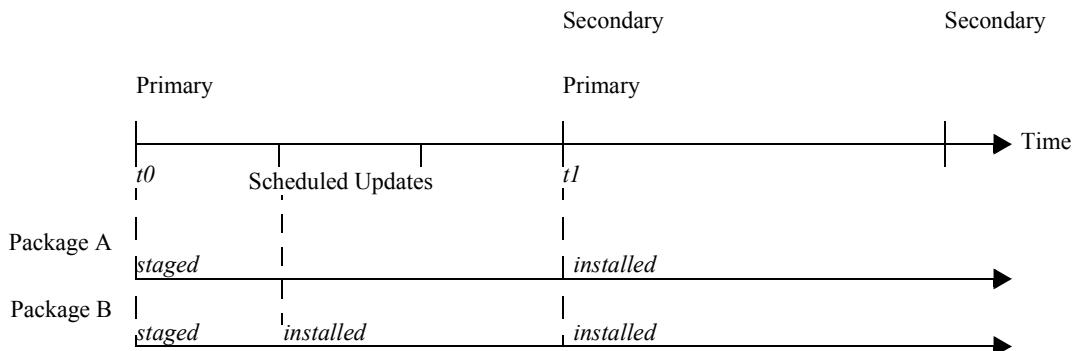
## Updating staged packages

You can set an update schedule for a package that has been staged on an endpoint to keep it current before it is installed. The behavior of a staged package during an update depends on the setting of the `adapter.updateinstall.pending` property, which is set to `true` when you create the package in Application Packager.

The behavior is best understood through an example. Assume that Package A has been packaged without setting the `adapter.updateinstall.pending` property. Then, in Policy Manager, you assign Package A to a target with a primary state of stage, activating at time  $t0$  and a secondary state of install, activated at time  $t1$ . During the period when the primary schedule is in effect, you set Package A to update every hour.

Package B has been packaged with `adapter.updateinstall.pending=true` and is assigned to the same target as Package A, with identical schedules.

Figure 8-5: Effect of updates on two staged packages



At time  $t_0$ , both packages are staged at the endpoint. After an hour, both packages update and the update causes Package B to install, while Package A remains staged. Both packages update until time  $t_1$ , with Package A in a staged state and Package B in an installed state. After time  $t_1$ , Package A, too, installs. The updates of Package B do not cause it to return to a staged state because of the endpoint state transition rules discussed in “Endpoint state transitions” on page 131.

Therefore, if you have not specifically allowed a package to update during staging, do not update it while it is staged unless you want to install it at the same time it updates.

## Roaming users and multiple-user machines

Before version 6.0.2.1, Policy Service determined the installation status of channels based on the channel state for the machine. It did not take into account any local user channel states. For example, if a channel is already installed for the machine but not for the currently logged-in user, Policy Service will not attempt to install the channel.

To address this issue, Policy Service versions 6.0.2.1 and higher checks for the currently logged-in user’s local channel state, rather than the channel state for the machine, for all channels packaged using Application Packager with the parameter `local.user.settings` set to true. See the section on user-specific files in the *BMC Marimba Client Automation Application Packager User Guide* available on the BMC Customer Support website.

If the channel state for the machine is installed, Policy Service verifies whether the currently logged-in user’s local channel state is also installed. If the machine’s state and the logged-in user’s state do not match, then Policy Service triggers installation of the channel. Because of the complexity of maintaining local channel states for users, only one primary state—installed—is supported at this time.

## User-controlled packages

A package is considered a *user-controlled package* when you allow the endpoint user to control when the package is installed. This feature is only available on Windows operating systems.

When a user-controlled package is ready for installation, the tuner icon in the system tray changes to include an exclamation point (as a

notification indicator) and a popup balloon appears on the user's desktop stating that updates are available. When the user right-clicks the tuner icon with the notification indicator and chooses **View Software Updates**, the BMC Software Installation window is displayed on the user's desktop. This window displays the name of the packages available for installation, the scheduled install time if installation is postponed, and the **Install Now** and **Snooze** buttons. When the BMC Software Installation window is displayed, the user can:

- Click **Install Now** to install the packages immediately.
- Choose a snooze interval and click **Snooze** to postpone the installation for the chosen amount of time.

When the snooze interval ends, the BMC Software Installation window is displayed again. If installation is snoozed past the scheduled postpone time, the packages are automatically installed.

---

Note: Although the user can select the packages to be installed, when a package is selected then all the packages having higher priority than the selected package will also be installed with an appropriate information alert to the user.

---

## Creating user-controlled packages

You can create user-controlled packages that allow users to control when they want to install the packages. This feature is only available for packages on Windows targets.

To create user-controlled packages, you must set tuner properties, the installation state, and a postpone schedule.

### ► To create a user-controlled package

- 1 Set the following tuner properties:
  - a Set the `marimba.subscription.usercontrolled.enabled` tuner property to **true**.

Setting this property to true enables the postpone schedule which you must set to create a user-controlled package. Setting it to false disables this feature. The default is false.

- b (optional) Set the `marimba.subscription.usercontrolled.stage tuner` property.

When set to true, the user-controlled packages to install will be downloaded before showing the pop-up to the user with the list of packages. When set to false, the user-controlled packages will not be downloaded until the user clicks the **Install** button or until the postpone date arrives. The default value is false.

- c (optional) To create a user-controlled package for silent tuners, set the `marimba.tuner.display.nodisplay` tuner property to **false**.

Setting this property allows the tuner icon to be displayed when usercontrolled packages are available for installation. However, the tuner icon is removed once the packages are installed.

- 2 Ensure that either the primary or the secondary installation state is set to one of the following states: Install, Install-Start, Install-Persist, or Install-Start-Persist.
- 3 On the Packages tab of the Edit Policy page, locate the row of the package for which you want to set a postpone schedule.
- 4 Click the arrow in the column to the left of the Priority column to list the schedules that you can set, or click **Set Common Schedule**.
- 5 In the **Postpone Schedule** section, click **Edit**.
- 6 In the **User control expires on date** field of the Edit Postpone Schedule page, specify the date and time when the package will be installed if the user has not already installed it.  
Enter the date and time in the following format: [mm/dd/yy hh:mm AM | PM]
- 7 Click **OK**.

The postpone schedule is set, and the package installation can be postponed by the endpoint user until the ending postpone date is reached.

---

Note: For user-controlled software packages to be interactive, the endpoint tuner and Policy Service must be version 8.1.01 or later. With earlier versions of the tuner or Policy Service, the packages are installed without user interaction.

---

---

Note: User-controlled channel are subscribed to without user interaction.

---

---

Note: For user-controlled software packages to perform updateFrom operations, the installation time and postpone time of the second version of the package should be the same.

---

---

Note: If the first version of the user-controlled package has an expiration time, then this expiration time should be equal to the installation and postpone schedule of the second version of the package.

---

---

Note: When the user selects a lower priority package for install, all the packages having a higher priority than the selected package will be installed and explanatory messages will be displayed.

---

**Best Practice:** For updatefrom scenarios, the first version can be user-controlled package or non user-controlled package, but the second version should be a non user-controlled package.

► **To add a custom logo to the BMC Software Installation window that displays user-controlled packages**

- 1 Place your custom logo on a web server.
- 2 Set the `marimba.tuner.company.logo.url` tuner property to the URL where your logo is located on your web server, for example:  
`marimba.tuner.company.logo.url=http://myserver:80/logos/mycomplanylogo.gif`

Your custom logo must be a GIF file whose maximum size should not exceed 158 pixels wide and 62 pixels tall.

## Package deployment using custom segmentation

You can use the custom segmentation hierarchy option when you have to perform package deployments on a specific platform. You can specify the segmentation hierarchy in both the Application Packager and the Transmitter.

---

Note: For more information on the custom segmentation hierarchy option for Application Packager, refer Chapter 4 Using the Package Editor of BBCA Application Packager User Guide. For more information on the custom segmentation hierarchy option for a Transmitter, refer Section II Transmitter Administration-> Chapter 11 Advanced features-> Segmentation hierarchy of BBCA Transmitter Proxy Guide

---



Chapter

# 9

# Configuring Policy Manager

Before you start creating and assigning policies, you must configure Policy Manager by performing the tasks described in the following sections.

The following topics are provided:

- The Policy Service plug-in (page 146)
- Configuring and publishing the Policy Service plug-in (page 146)
- Plug-in configuration page: directory service fields (page 151)
- Policy Plug-in configuration for Matrix42 Empirum connection settings for OS migration (page 153)
- Setting up access control lists (page 153)
- Configuring policy compliance settings (page 155)
- Creating and managing profiles for Windows Power Options (page 156)
- Feature Properties (page 158)

# The Policy Service plug-in

The Policy Service plug-in is the server-side component of Policy Service. The plug-in resides on the transmitter that hosts the Policy Service channel. The plug-in connects to the directory service to determine which targets an endpoint belongs to—and it queries the directory service to determine what policies apply to an endpoint.

Only primary administrators are allowed access to the configuration settings for the Policy Service plug-in. Standard administrators and operators are not permitted to access or change the configuration settings for the plug-in.

You can set options for the Policy Service plug-in to:

- Specify the URL for the Policy Service channel where you want to publish the plug-in.
- Supply the publish user name and password required for publishing configuration changes to the transmitter that hosts the plug-in (if publish permission is required).
- Specify which directory service you want to use.
- Set the number of connections from the plug-in to the directory service.

---

Note: To change the configuration settings, you must be logged in as a primary administrator. Part of the initial installation and configuration is to specify which users are primary administrators and which users are standard administrators. Standard administrators are prevented from accessing and changing the configuration settings for Policy Manager.

To determine whether you are logged in as a primary administrator, place your mouse pointer over the Status button in the upper-right corner of the console window.

---

## Configuring and publishing the Policy Service plug-in

Typically, you must configure and publish the plug-in whenever you install or upgrade Policy Manager. However, you must also *republish* the plug-in during the following situations:

- If you change the settings on the Plug-in Configuration page.

- If you change the settings using the `-setpluginparam` command.
- If you modify the Policy Management object's attributes using the `-configSet` command.

---

Note: You do not need to publish the Policy Service plug-in each time you edit policies.

---

When possible, Policy Manager takes the following directory service information from the system settings (Applications > Console > System Settings > Data Source > Directory Services) and automatically enters them in the Plug-in Configuration page:

- Directory service host name and port number
- Base DN
- Bind DN
- Whether or not to use SSL

If this information is already on the page, you must confirm that these are the settings you want to use.

---

Note: If you had previously published the Policy Service plug-in to a transmitter with SSL enabled, restart Policy Manager before you publish the plug-in to a non-SSL transmitter.

---

## ► To configure and publish the Policy Service plug-in

- 1 Click the Configuration tab.
- 2 On the Configuration page, click the Plug-in link.

The Plug-in Configuration page appears. The upper left-hand side of this page shows the date and time when the plug-in was last published.

- 3 In the Master Transmitter section, specify information for publishing the plug-in:
  - a Specify the URL for the Policy Service channel where you want to publish the plug-in.

**Note:** The URL must end in `SubscriptionService`.

- b If publish permission is required, supply the publish user name and password required for publishing configuration changes to the transmitter that hosts the plug-in.

**Note:** The publish user name and password are only required if you (or the primary administrator for your enterprise) has configured transmitters to require user names and passwords so you can publish channels and packages to them.

- 4 Enter the information required for the Directory Services section.

Usually, the fields in this section are already filled out with information from the directory service specified for CMS. The only fields you might need to change are the ones for the bind DN and the bind DN password, if, for example, you want to use a bind DN that has different permissions from the one specified for CMS.

For information about the fields on this page, see “Plug-in configuration page: directory service fields” on page 151.

- 5 Depending on the type of directory service you are using, you might be able to verify the connection between Policy Manager and the directory service by clicking the Go button next to Test server(s).

Policy Manager makes a connection to the directory service you specified, verifying that the server and port accepts connections using the user name and password. Policy Manager then verifies the existence of the Subscription configuration object. The Subscription configuration object is located in the ConfigObjects container, a child of the BMC CM container. This object is created when you run the install or update LDIF scripts.

**Note:** The directory service need not be accessible from the computer where Policy Manager is installed. Therefore, even if this test fails, the Policy Service plug-in might still be able to make a successful connection.

- 6 If you want to use Policy Manager with newly provisioned machines, select the Allow the use of policies as models for newly provisioned machines check box. See the chapter about integrating tuner installation with OS provisioning in the *BMC Marimba Client Automation Deployment Manager Guide*, available on the BMC Customer Support website.

- 7 Click Preview.

The Preview Plug-in Changes page appears. This page highlights the changes you have made on the Plug-in Configuration page, so that you can review them before you publish the plug-in. By default, all settings are displayed with the changes highlighted. To display changes only, click the Show changes only link.

- 8 After reviewing your changes and modifying them if necessary, click Publish.

When you click the Publish button on this page, you publish the Policy Service plug-in for the Policy Service channel to the transmitter and you return to the Configuration home page.

If the plug-in fails to start, an SNMP trap is sent to Logger plug-in which logs the message in database. The number of update requests that have failed during a specified time period is also logged. This time period is 1800 seconds (30 minutes) by default. You can change this time period by setting the number of seconds in the `marimba.subscription.snmpintervalsec` property.

**Tip:** If you want to verify that the Policy Service plug-in has been published, use Transmitter Administrator to view the channels on your transmitter and expand the Policy Service channel to view its segments, as shown in Figure 9-1 on page 150. If you see the `.configurator` segment, you have published the plug-in to the transmitter.

Figure 9-1: Using transmitter administrator to view the Policy Management plug-in

The screenshot shows the BMC Configuration Management Transmitter Administration interface. At the top, there are tabs for Home, Tuner, Transmitter (which is selected), and Proxy. The main content area is titled "Manage Channels for <http://syslab164:7000>". It displays the status of the transmitter (On, Master, 7.0.0tp49 version, 4159 channels, up since 3/15/2006 4:19 PM GMT-08:00). Below this, there are links for Edit Settings, Clear Index Cache, Clear File Cache, View License, and Launch Console. A note says: "Use the tabs on this page to manage the channels on the transmitter. In the status section below, you can view information about the transmitter and perform actions on the transmitter. Certain actions require you to stop and restart the transmitter. For more information, click Help." The "Content" tab is selected, showing "Transmitter Content". A note says: "You can add, delete, and rename a channel, as well as create a CAR file from a channel. To view information about channel segments, click the arrow next to a channel. Be careful about changing content on a repeater or mirror. For guidelines, click Help." Below this is a table of channels:

Name	Last Updated	Version	Size
SubscriptionReporter	3/14/2006 4:47 PM GMT-08:00	7.0.0tp49	907K
SubscriptionService			
<a href="#">configurator</a>	3/14/2006 12:45 PM GMT-08:00	7.0.0tp49	1.7MB
<a href="#">Windows.x86/any</a>	3/14/2006 12:45 PM GMT-08:00	7.0.0tp49	129K
<a href="#">Windows CE,StrongARM/any</a>	3/14/2006 12:45 PM GMT-08:00	7.0.0tp49	

A red oval highlights the "configurator" row in the table.

The plug-in appears as the .configurator segment of the Policy Service channel on the transmitter.

## Starting or stopping the plug-in

By default, the plug-in is disabled. You can enable it from the Policy Manager Plug-in Configuration page by changing the Plug-in Status selection box to Enable. Enable this value before the Subscription Service channel update occurs on an endpoint.

If you need to prevent the plug-in from servicing update requests, such as during an infrastructure upgrade, you can disable the plug-in from the Policy Manager Plug-in Configuration page by changing the Plug-in Status selection box to Disable.

## Plug-in configuration page: directory service fields

Typically, the directory service fields on the Plug-in Configuration page are already filled out with information from the directory service specified for CMS. Depending on the directory service you are using, the list of fields you must fill out might be different. The Plug-in Configuration page includes the following directory service fields:

- **Host name and port**—The host name and port of the directory service, in the form `<hostname>:<port>`. The directory service you specify here is used to store policies. The entries default to those for the directory service specified for CMS, which is used for login authentication. (Usually, you are not required to fill out this field for Active Directory.)

If your organization provides replicated directory services for storing policies, use a directory service-to-repeater mapping. You can take advantage of replicated directory services by configuring Policy Management so that each repeater contacts a nearby directory service, eliminating the need to contact the one assigned to the master transmitter. Moreover, you can assign a list of directory services to each repeater. If one directory service in the list fails, the repeater attempts to contact the next one, eliminating single point-of-failure problems. See “Setting directory services for repeaters” on page 28. You should also use the `-ldapservers` command-line option described in “Using runchannel options” on page 389.

- **Use an SSL connection between the transmitter and directory service**—Check this box to enable Secure Sockets Layer (SSL) for secure communication between the transmitter and the directory service. See the section about security and SSL in the *BMC Marimba Client Automation CMS and Tuner Guide*, available on the BMC Customer Support website.
- **Base DN**—The base distinguished name for the directory service connection. The base DN defines the scope of the directory view as seen by the plug-in. In most cases, this is equivalent to the Sun Java System Directory Server (Sun One) Directory suffix. (Usually, you are not required to fill out this field for Active Directory).

For example, on ADAM / AD LDS or Sun ONE Directory, you can use `dc=company,dc=com`.

- **Bind DN**—The distinguished name of the user account that the Policy Service plug-in uses when it establishes a connection to the directory service. This user account should have read permissions in the scope of the directory defined by the Base DN entry. This scope typically maps to the entire directory tree.

For Active Directory, use one of the following formats:

- The full distinguished name (DN), such as  
`cn=Administrator,cn=Users,dc=company,dc=com`
- The user principal name (UPN), such as `Administrator@company.com`

For ADAM / AD LDS, use the full distinguished name (DN), such as  
`cn=Administrator,cn=Users,dc=company,dc=com`

For Sun Java System Directory Server (Sun One) Directory, use one of the following formats:

- The full distinguished name (DN), such as  
`uid=Administrator,ou=People,dc=company,dc=com`
- The common name (CN) for the directory administrator, such as  
`cn=Directory Manager`
- **Bind DN password**—The required password for directory service access. The plug-in uses this password for directory service connection.
- **Directory service connection pool size**—The Policy Service plug-in uses a pooling mechanism to establish and maintain connections to the directory service. This parameter sets the maximum number of connections established in the pool and can typically be left at the default value of 25.
- **The expiration time (in minutes) for the last successful host connection**—If you specify a list of host names for directory services failover, Policy Manager goes down the list until it successfully connects to a host. Policy Manager uses that successful host connection until the expiration time that you specify. Then, it attempts to make a new connection to the first host name in the list.

- **Allow the use of policies as models for newly provisioned machines**—If you are integrating tuner installation with OS provisioning, selecting this check box allows you to identify groups in the directory service whose policy you want to use as a model for installing applications and content to the newly provisioned machines. See the chapter about integrating tuner installation with OS provisioning in the *BMC Marimba Client Automation Server Deployment Manager Guide*, available on the BMC Customer Support website.

## Policy Plug-in configuration for Matrix42 Empirum connection settings for OS migration

In the Policy Manager Plug-in configuration page, the Matrix42 Empirum connection settings must be republished. These settings are used by Policy Service plugin to enable PXE activation for the endpoint. The PXE activation happens only when a personal backup is complete on the endpoint. This ensures that an accidental reboot of the computer will not start OS migration.

Select the **Allow repeater to connect Empirum** option to allow the repeater to connect to the Empirum Server. By default, only the Master Transmitter is allowed to connect to the Empirum Server.

## Setting up access control lists

*Access control lists (ACLs)* for Policy Manager identifies the administrators and groups (of administrators) who have permissions to view that target, to view policies for that target, and to assign policies to that target. You can set up access control lists for Policy Manager because, for security reasons, you want some administrators to view and change policies for certain machines and users only.

For example, if a system administrator in France is responsible for machines in France only, and if a system administrator in Germany is responsible for machines in Germany only, you can use access control lists for Policy Manager so that each system administrators can view and assign policies only to the machines for which they are responsible.

## Using the access control feature

As a best practice, decide whether or not you want to use the ACL feature during installation and setup. To determine whether the ACL feature is on or off, place your mouse pointer over the Status icon in the upper-right corner of the console window.

---

Note: BMC recommends assigning ACLs and permissions to administrators before turning on the ACL feature. Otherwise, administrators who log in to Policy Manager are not able to manage policies. As a best practice, assign ACLs and permissions to administrators, “stage” them, and then turn on ACL functionality in Policy Manager when you are ready to enforce permissions.

---

Before using Policy Manager to turn on the access control feature, use the console to set up the following:

- User authentication
- LDAP synchronization
- Access control lists
- Assign permissions for access control lists
- Assign permissions for applications

See the chapter on access control lists in the *BMC Marimba Client Automation CMS and Tuner User Guide*, available on the BMC Customer Support website.

### Enabling and disabling ACL functionality

After you set up access control lists, primary administrators can use Policy Manager to enable ACL functionality. ACLs are off by default.

You can enable or disable ACL functionality using the Configuration tab or the command-line interface.

#### ► **To enable ACL functionality using configuration options**

- Click the Configuration tab.
- Click ACL Options.

The Enable Access Control page appears.

- Select the Enable/Disable Access Control check box.
- Click OK.

#### ► To turn on the ACL feature using the command line

- On the command line, enter the following command:

```
runchannel <Policy_Manager_URL> -user <user_name> -password  
<password> -configSet marimba.subscription.acl true
```

You can turn off the ACL feature by using the same command and replacing true with false. You can also use the -preview option to preview the change you want to make.

Make sure you restart Policy Manager after you turn on the ACL feature.

## Configuring policy compliance settings

As you use the policy compliance feature of Policy Manager, you might want to change the policy compliance settings so that they suit the needs of your enterprise. For example, if you want information about compliance percentage or number of machines to persist longer in your queries, you can change the maximum cache time so you can view results for a longer time period. For more information about policy compliance, see “Viewing policy compliance” on page 317.

#### ► To change the policy compliance settings

- 1 Click the Configuration tab.
  - 2 On the Configuration page, click the Compliance Options link.
- The Compliance Options page appears.
- 3 Configure the following settings:

- **Enable the collection of compliance data?**

Choose Enable to turn on compliance reporting functionality.

- **Maximum cache time for list results (in seconds)**

Determines the amount of time Policy Manager caches compliance information for the list of items (such as packages or targets) in a group. After the specified time, you must rerun your query.

- **Maximum cache time for non-list results (in seconds)**

Determines the amount of time Policy Manager caches the compliant, non-compliant, and not checked-in numbers for a group. After the specified time, you must rerun your query.

■ **Time limit for checked-in status (in hours)**

Determines the amount of time Policy Manager considers a machine compliant or non-compliant before marking it as not checked-in.

Typically, this is a negative number. For example, if you specify -48, and the machine has not checked in for two days at the time you run a query, Policy Manager marks the machine as not checked-in. Policy Manager must recalculate the compliance information before determining compliance or non-compliance.

- 4 Click **Apply** to save your changes and return to the Configuration page.

## Creating and managing profiles for Windows Power Options

You can create profiles for Windows Power Options that you can apply to machines through a policy.

For both plugged in machines and those running on a battery, the available options are

- Idle Time to Turn Off Monitor
- Idle Time to Turn Off Hard Disk
- Idle Time for Standby
- Hibernate option
- Idle Time for Hibernate
- Prompt for Password when Computer Resumes Activity from Standby
- Apply Power Options Properties Every Time Policy Service Runs

### ► To create a Power Options profile

- 1 Click the Configuration tab in Policy Manager.
- 2 Click Power Setting Profile Configuration.
- 3 Click **Add**.
- 4 Type a name for the Profile in **Profile Name**.

Only alphanumeric characters, spaces, and underscores are allowed in profile names.

- 5 (optional) Type a description for the profile in **Profile Description**.
- 6 Select the Power Options properties for machines that are “plugged in” and for those that are running on a battery (such as a laptop).
- 7 Click **Apply Power Options Properties Every Time Policy Service Runs** if you want to enforce these power options during every policy update.
- 8 Click **OK**.
- 9 See “Setting Power Options for Windows targets” on page 273 to create a policy that uses a Power Options profile.

#### ► To delete a Power Options profile

- 1 Click the **Configuration** tab in Policy Manager.
- 2 Click **Power Setting Profile Configuration**.
- 3 In the table, click the check box next to the profile that you want to delete.
- 4 Click **Remove**.

The profile is deleted.

---

Note: Deleting a power options profile will not remove the profile from the endpoint.

---

#### ► To modify a Power Options profile

- 1 Click the **Configuration** tab in Policy Manager.
- 2 Click **Power Setting Profile Configuration**.
- 3 In the table, click the check box next to the profile that you want to modify.
- 4 Click **Edit**.
- 5 Change any of the settings as needed and click **OK** to save your changes.

---

Note: To enable or disable the hibernate option in Windows 7 endpoints, the logged on user must have Admin privileges.

---

# Feature Properties

- "marimba.subscription.autoscan.enabled"
- "marimba.subscription.ucd.enabled"
- "marimba.subscription.usercontrolled.enabled"

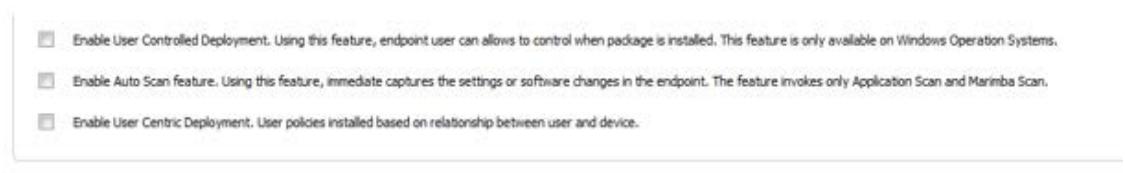
## Permission

All modules are supported.

## Location

Policy Manager -> Configuration->Advanced Options

## Screen Shot

- 
- Enable User Controlled Deployment. Using this feature, endpoint user can allows to control when package is installed. This feature is only available on Windows Operation Systems.
  - Enable Auto Scan feature. Using this feature, immediate captures the settings or software changes in the endpoint. The feature invokes only Application Scan and Marimba Scan.
  - Enable User Centric Deployment. User policies installed based on relationship between user and device.

# Chapter 10 Viewing targets and packages

When you use Policy Manager, you select packages and assign them to *targets*. A target encompasses one or more *endpoints*. An endpoint is a Policy Service process running on a *machine* (computer or other device).

The following topics are provided:

- Types of targets (page 160)
- Viewing targets (page 164)
- What is a package? (page 175)
- Using the Query Builder to search for custom packages (page 181)

# Types of targets

A target can be one of the following:

-  A user (an endpoint identified by a login name)
-  A machine
-  A user group
-  A machine group
-  A container or organizational unit (OU)
-  A domain in Active Directory
-  A collection, which is the result of a database or LDAP query. See “What is a collection?” on page 161.
-  *All Endpoints*, a special virtual group composed of all endpoints. See “What is the All Endpoints target?” on page 160.
- Site

After you assign a package to a target, you can assign an install state to each package, along with schedules and property settings that affect the installation at each endpoint. All information that you assign a target is called its *policy*.

**Hierarchical resolution of policies.** The resolution of policies for targets is hierarchical. For example, this is the endpoint:

`cn=user1,cn=computers,ou=finance,dc=company,dc=com`

The endpoint gets policies assigned to the following targets:

- `cn=computer,ou=finance,dc=company,dc=com`
- `ou=finance,dc=company,dc=com`
- `dc=company,dc=com`

## What is the All Endpoints target?

The All Endpoints target represents all targets (users or machines) that have a Policy Service configured to the LDAP in which it resides.

You can use the All Endpoints target to distribute software to everyone in your organization. It is also useful for setting target and package properties for all endpoints to provide a consistent end-user experience.

The All Endpoints icon  identifies the All Endpoints target in the Target View page. The All Endpoints target is a virtual group and does not actually exist in the underlying directory. Therefore, the group does not need to be maintained with a list of all the endpoints. When changes occur (endpoints are added or removed), updates to the All Endpoints target occur automatically.

If you want all endpoints to receive a package, assign it to the All Endpoints target. The All Endpoints target is also useful for enforcing target and package properties at all endpoints to provide a consistent end-user experience.

If you want to keep certain targets from receiving packages that are otherwise assigned to All Endpoints, you can assign them separately and specify the Exclude state. The Exclude state overrides all other states. Packages with that state are not distributed to their assigned targets.

## What is a collection?

A collection is list of user or machine names that results from running either a database or LDAP query.

**Based on an LDAP query.** To create a collection based on an LDAP query, you use Policy Manager command-line interface to create a query that returns a list of users or machines that meet your criteria and then refresh the query, either manually or according to a schedule. The resulting list of users or machines is available for selection as a target in Policy Manager.

**Based on a database query.** If you have the Inventory module, you can use Report Center to create collections that you can then use as targets in Policy Manager. To create a collection, you use Report Center to create a query that returns a list of machines that meet your criteria, save this query in the Collections folder (in Report Center), and then run the query, either manually or according to a schedule.

You can assign policies to the entire collection or to individual items in the collection. However, you cannot assign them to the result set groups. The result set groups are temporary and cannot be assigned policies.

For instructions about creating, running, and deleting database query collections, see the *BMC Marimba Client Automation Report Center User Guide*. For information about LDAP query collections, see “LDAP query collections” on page 82.

## What is an excluded target?

You identify an excluded target when you select the *exclude* state for a target-package pair. The Exclude state specifies that the target does not receive the associated package, even if it has been assigned elsewhere, such as a target group. This feature is useful when you want all but a few members of a large target group such as All Endpoints to receive a package. In this case, you assign the package to the target group, create separate policies to the targets you want to exclude, and set the exclude state for them.

## What is a directly assigned target?

A target is said to be in *direct assignment* if a package or property has been assigned to it explicitly as part of a policy. If a target has been assigned a package only because it is a member of a user or machine group that has been assigned the package, the target is in *indirect assignment*.

When browsing targets, directly assigned targets are identified by a policy icon  in the browser-based interface.

When you delete all packages and properties that are directly assigned to a target, the target is no longer identified by a policy icon. The policy icon still appears if there are any tuner or package properties directly assigned to the target (even if all packages have been deleted).

## What is a site?

A site is a subnet which contains a list of computer network addresses specific to a location. Microsoft Active Directory users can configure policy for site and services. However, ADAM and SunOne users cannot assign policies based on sites. To overcome all these drawbacks, you can use the site based deployment feature of the Policy Manager to configure policies for sites or subnets irrespective of and type of LDAP implemented. You can also use this feature to recently or newly added computers in the network. You can use the site based policy deployment to assign specific policies to computers that are in a particular site. The list of sites is retrieved from the Transmitter where the sites are configured. You can select a single site or multiple sites and assign policy such as packages, patch groups, tuner or package properties to the selected site. While updating Policy Service on endpoints, Policy plug-in brings down policies assigned for the respective site along with policies assigned for user, machine, groups, and collections.

---

Note: if a machine belongs to multiple sites, Policy Manager resolves the conflict at the endpoint based on channel state and priority set for different sites.

---

## Prerequisite

Prior to using the site based deployment feature, you must enable this feature in the Advanced Options page of Configuration page of Policy Manager.

When you enable the site based deployment feature in Policy Manager the `subscriptionmanager.sitebaseddeployment.enabled` property is set and enabled. Once this option is enabled, the list of sites is retrieved from the Master Transmitter. You must specify the URL of the master transmitter where the SBRP settings are configured when you enable this feature. The `subscriptionmanager.sitebaseddeployment.mastertxurl` property stores the value of the Master Transmitter URL which you specify. Policy Manager retrieves the list of sites information from the Master Transmitter or from the database which is configured in CMS. The Policy plugin retrieves policies assigned for sites based on this property and Policy Service on the endpoints sends site information based on this property. By default, the `subscriptionmanager.sitebaseddeployment.enabled` property is disabled.

Ensure to configure SBRP sites in the SBRP configuration page for the master transmitter, and specify the Transmitter in Policy Manager Configuration page.

## Enabling site based deployment

You must enable the site based deployment feature before you can assign policies to computers based on site.

### ► To enable or disable site based deployment

- 1 Click the Configuration tab.
- 2 On the Configuration page, click the Advanced Options link.  
The Advanced Options page appears.
- 3 To enable this feature, select the **Enable site based deployment** option.
- 4 In the **Master Transmitter URL** text box, enter the URL of the Master Transmitter.

## Selecting targets based on sites

You can assign site based policies using the following two techniques:

- Single

You can select only a single target like user, machine, collection, or site.

- Multi-mode

If you select Multi-mode policy assignment, then you can select multiple targets like user, machine, collection, and site and assign common policies.

### ► To select targets based on site

- 1 On the left side of the Target View page, select **Single** or **Multiple** link.
- 2 If you have enabled site based deployment feature, you can view the **Sites** icon. Click the **plus** icon to view the list of sites available.

Policy Manager displays the list of sites available and which is retrieved from the master transmitter.

- 3 To edit the policy for a site, click on the required site.

Policy Manager displays the packages assigned to site. You can click on Edit to edit the policy for the site.

You can perform various actions on the policy assigned to a site like edit, copy, update, or compliance operations.

## Viewing targets

This section includes the following topics:

- “Browsing targets” on page 165
- “Viewing members of a target” on page 166
- “Searching for targets” on page 167
- “Viewing target details” on page 171
- “Viewing packages assigned to a target” on page 173

## Browsing targets

The left side of the Target View page shows the targets to which you can assign packages. When you select a target, you can view the packages assigned to it, directly and indirectly, on the right side of the page.

You can view all the available targets by navigating through the left side of the page. The left side of the page shows a list of targets—individual users and machines, groups of users and machines, and collections resulting from database and LDAP queries.

When you first view this page, you see the target's Home, which consists of the high-level containers in the directory service where target information is stored. For example, in Active Directory, you usually see the root domains in the forest environment. You can expand the root domains to display the objects that can be used as targets in that domain. See "Targeting Active Directory groups" on page 45.

The target icons in the Target View page indicate how Policy Manager is obtaining its list of targets:

- **Domains**—An expandable domain icon  indicates that targets are being obtained from an Active Directory forest environment. Root domains and subordinate domains are represented by this icon.
- **Containers**—An expandable folder icon  indicates that targets are obtained from the directory service. Organizational units and containers are represented by this icon.
- **Users (sourced from transmitter)**—An expandable transmitter icon  indicates that users and user groups are obtained from the transmitter where you have published the Policy Service plug-in (using the Plug-in Configuration page).

---

Note: If you (or your primary administrator) have not yet published the Policy Service plug-in to the transmitter, Policy Manager does not know which transmitter to use. Policy Manager cannot obtain user and user group information. Additionally, if the transmitter has restricted access, you must set up the user name and password to use for authentication against the transmitter by running the following command-line option:

---

```
-txadminacess <username> <password>
```

The user name and password that you specify with this option are the ones required by the transmitter. You still need to specify the user name and password required for authentication by Policy Manager. See the `-user` and `-password` command-line options in “Command-line reference” on page 385.

---

In addition, the All Endpoints target is displayed. This is a virtual target that contains all of the endpoints at your site that have Policy Service installed.

If you are browsing a multidomain forest environment (Active Directory), you can browse across trees and domains.

## Viewing members of a target

This topic describes how you can view the contents of an expandable target (such as the users in a user group) from the Target View page.

► **To view the contents of an expandable target from the Target View page**

- On the Target View page, click + next to the expandable target’s name.

---

Tip: Holding your mouse over the targets shows their distinguished names (DNs) or organizational units (OUs) in the directory service.

---

When a target expands to show its members, you see the following changes to the left side of the Target View page:

- If you expand a container, the right pane displays the container name. The title persists until you navigate to another container.
- If you expand a group, the members of the group appear in the left pane, and the name of the group is added to the breadcrumb trail at the top of the left pane to orient you in the target hierarchy.
- Search options are displayed in the left pane that enable you to search for a target within the current container. See “Searching for targets” on page 167.
- A link is displayed in the lower left that enable you to switch between single and multiple selection modes.

- A breadcrumb trail is displayed in the left pane that shows your location in the container and group hierarchy so you can navigate easily. The link following the Home link shows the top level, usually the root domain.
- If the expanded list contains more than a predetermined number of items, the **next** link activates so you can access other items in the list. Use the **next** and **previous** links or the numbered drop-down list to navigate through long lists.

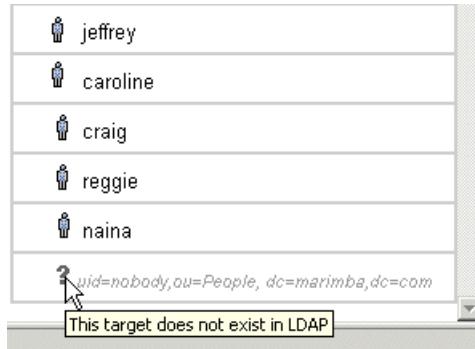
---

Note: Active Directory and ADAM / AD LDS do not display the total number of entries until you page through all search results. The drop-down list displays “...” for the total number of entries until you manually navigate to the last page.

---

- If a group contains any invalid members, the invalid member appears with a question mark, and its DN appear dimmed and in italics. If you hover the cursor over the question mark, information about the invalid member appears, as shown in Figure 10-1 on page 167.

Figure 10-1: An invalid member of a group



## Searching for targets

There are two options available when searching for targets:

- **Basic search.** This option allows searching for targets using the common name (CN). You can also specify a type to narrow your search. See “Basic search for targets” on page 168.

- **Advanced search.** For advanced users who are familiar with LDAP queries and search filters, this option allows searching for targets with more flexibility. See “Advanced search for targets” on page 169.

You can search for targets in the current expanded target group or container. This feature is useful when dealing with containers that contain large numbers of sub-containers or organizational units that contain large numbers of members.

---

Note: Active Directory and ADAM / AD LDS do not get the total number of entries until you have paged through all of the search results. Therefore, the drop-down list displays “...” for the total number of entries until you reach the last page.

---

## Basic search for targets

### ► To perform a basic search for targets

- 1 On the left side of the Target View page, click the container in which you want to perform a search.
- 2 On the left side of the Target View page, click the Basic Search link.

The Basic Search area appears.

Figure 10-2: The basic search area for targets



- 3 Provide the following information:
  - In the Search for text box, enter the common name (CN) or part of the CN for a target.

You are not required to know the complete name of the target. You can use the asterisk (\*) as a wildcard character to substitute for zero or more characters in the name. If an asterisk is part of the name, you must escape it so that it is not considered a wildcard character. See “Special characters in search strings” on page 171.

**Note:** The search is *not* case-sensitive.

- In the Limit to list, choose the type of target for which you are searching, or choose All types.
- To further limit your search, you can select the Do not search sub-containers check box.

4 Click Go.

Search results are displayed below the search area. When results are displayed in the list, the breadcrumb trail label changes to indicate that you are viewing search results.

To return to the top-level target view, click the Home link.

To return to the view where you originally started the search, click the Target View tab.

## Advanced search for targets

Use advanced search if you are familiar with LDAP queries and search filters for the directory service you are using. Because Policy Manager passes the query directly to the directory service, the search is faster than basic search. You can enter simple queries, such as searching for targets with users in the CN:

`cn=users`

You can enter more complex queries that use conditions, such as searching for targets with either users or groups in the CN by using the or notation (|):

`( | (cn=users) (cn=groups))`

For more information about LDAP search filters, see <http://www.ietf.org/rfc/rfc2254.txt>.

---

Note: You cannot use advanced search to look for targets using the complete distinguished name (DN).

---

## ► To perform an advanced search for targets

- 1 On the left side of the Target View page, click the container in which you want to perform a search.
- 2 On the left side of the Target View page, click the Advanced Search link.  
The Advanced Search area appears.

Figure 10-3: The advanced search area for targets



- 3 In the LDAP query text box, enter an LDAP search filter.

You can enter simple or complex queries, which Policy Manager passes directly to the directory service. If your query includes any special characters (such as, \*, (, ), \), see “Special characters in search strings” on page 171.

**Note:** The search is *not* case-sensitive.

- 4 To further limit your search, you can select the Do not search sub-containers check box.
- 5 Click Go.

Search results are displayed below the search area. When results are displayed in the list, the breadcrumb trail label changes to indicate that you are viewing search results.

To return to the top-level target view, click the Home link.

To return to the view where you originally started the search, click the Target View tab.

## Special characters in search strings

Certain characters require special representation when you enter them in search strings, either from the browser-based interface or the command line. To enter these characters in search strings, you must use the hexadecimal ASCII code for the character, preceded by a backslash. The special characters are summarized in Table 10-1 on page 171

Table 10-1: Special characters in search strings

Special characters	ASCII value (Hexadecimal)	Search string representation
--------------------	---------------------------	------------------------------

*	0x2a	\2a
(	0x28	\28
)	0x29	\29
\	0x5c	\5c

Some examples of using special characters in search strings are shown in Table 10-2 on page 171

Table 10-2: Examples of searching for special characters

To search for...	Enter the following...	Comment
Any string beginning with “abc(“	abc\28*	Shows how to represent a parenthesis character.
Any string containing an asterisk (“*”).	*\2A*	Shows how to represent an asterisk in a value, preventing it from being interpreted as a substring indicator.
The string “C:\MyTarg”	C:\5cMyTarg	Shows how to escape the backslash character.

## Viewing target details

The Target Details page shows all of the packages assigned to the selected target or targets, along with the installation priority, primary and secondary states, and schedules for each target.

### ► To view target details

- 1 On the Target View page, select a target, or several targets, if you are in multiple selection mode.
- 2 Click the Details View link in the right pane.

The Target Details page shows the packages assigned to the selected target or targets, along with the installation priority, primary and secondary states, and schedules for each target.

View the details of selected packages or edit their assignment to the targets. Click the column name to sort by column content. Page through long lists by clicking the previous and next links.

- 3 To return to the Target View page, click the Basic View link.

## Sorting the list of packages

On the Target Details page, you can sort the list of packages based on the following:

- Schedules
- Name
- Install priority (only available in single selection mode)
- Target to which the package is directly assigned (only available in single selection mode)
- Type (as indicated by the icon; for example,  for a package and  for a patch group)

### ► To sort the list of packages

On the Target Details page, click the column name to sort the packages by that column (either in ascending or descending order).

- If you sort by one of the schedule columns, the packages are arranged according to the activation date and time for that schedule. For example, if you sort by the primary schedule, the packages are arranged according to the activation date and time set for the primary schedule.
- If you sort by the name of the package, the packages are arranged alphabetically.
- If you sort by the install priority, the packages are arranged according to the priority in which they are installed. If you want to change the priority in which applications are installed, see “Setting the install priority for packages in a policy” on page 248.
- If you sort by the target to which the package is directly assigned, the packages that have been directly assigned are at the top, followed by indirectly assigned packages.

## Viewing packages assigned to a target

The right side of the Target View page shows the packages assigned to the target(s) you select in the Target List. The display differs somewhat depending on the mode you have selected. To switch between modes, click on the Single or Multiple link at the top of the Targets List. The following sections describe the two modes:

- “Single selection mode” on page 173
- “Multiple selection mode” on page 174

### Single selection mode

In single selection mode, you can choose a single target and view the packages that have been assigned to it, either directly or indirectly.

#### ► **To show the packages that have been assigned to a single target**

- 1 On the left side of the Target View page, make sure that you are in single selection mode. If you are in multiple selection mode, click the Single link to switch to single selection mode.
- 2 On the left side of the Target View page, locate and select the target for which you want to show packages.

After you click a target, any packages that have already been assigned appear on the right side of the page. The selected target might or might not have already received the packages shown.

**Note:** Resolution of state conflicts is not done, so it is possible for two packages with the same name (or URL) but different states to appear.

On the right side of the page, the name of the selected target appears at the top. (This is usually the `cn` attribute of the entry in the directory service.) Below the target name is a table that lists all the packages that are assigned (both directly and indirectly) to the selected target.

In single selection mode, you have the following options:

**Details View**—Click to view the states and schedules for each package assigned to the target you selected.

**Edit**—Click to create or edit the policy for the target. See “Creating and editing policies” on page 207.

Note: When you edit a policy, you see packages that are directly assigned to a target only. To edit policies that include indirectly assigned packages, you select the target to which they are directly assigned. To find out which target the packages are directly assigned, look in the Directly Assigned To column. For more information about the concept of direct assignments, see “What is a directly assigned target?” on page 162.

---

The following columns appear in the table on the right side of the page:

**Packages**—Shows the names of the packages assigned to the selected target. Place your mouse pointer over the package name to see its URL.

**Primary and Secondary States**—Shows the state or states that determine the way the package is to be distributed to each endpoint. For a list of installation states, see “Overview of installation states” on page 215.

**Directly Assigned To**—Shows the target to which the package was directly assigned. The package appears in the list because the selected target is a member of a group (or the target itself) to which the package was directly assigned.

## Multiple selection mode

In multiple selection mode, you can choose two or more targets and view the packages that they have in common.

### ► **To show the packages that have been assigned to multiple targets**

- 1 On the left side of the Target View page, if you are not already in multiple selection mode, switch to multiple selection mode by clicking the Multiple link.
- 2 On the left side of the Target View page, locate and select the targets for which you want to show packages.

Each time you select a target, the ***n* Target(s) Selected** link, at the top of the right pane, increments to indicate the number of selected targets. Click the link to display a list of the target names. To remove a target from that list, select the check box and click the Remove button.

The right pane contains a list of the packages *directly* assigned to *all* the selected targets (the *intersection* of the policies for the selected targets). If a package is not assigned to *all* selected targets, it does not appear in the list. The selected target might or might not have already received the packages shown.

In multiple selection mode, you have the following options:

**Details View**—Click to view the states and schedules for each package assigned to the targets you selected.

**Edit**—Click to create or edit the policy for the targets. See “Creating and editing policies” on page 207. When you edit the policies for multiple targets, you can add, edit, and remove packages that are directly assigned to all targets only.

The following columns appear in the table on the right side of the page:

**Packages**—Shows the names of the packages assigned to the selected targets. Place your mouse pointer over the package name to see its URL.

**Primary and Secondary States**—Shows the state or states that determine the way the package is to be distributed to each endpoint. For a list of installation states, see “Overview of installation states” on page 215.

## What is a package?

When you use Policy Manager, you select *packages* and assign them to targets. A package is typically an application that you want to install on endpoints, but it can be composed of any data that you want to distribute. You typically use Application Packager to create a package.

After you assign a package to a target, you can assign a state to each package-target pair, along with schedules and property settings that affect the installation at each endpoint.

The package icon  identifies packages.

## Viewing packages

This section includes the following topics:

- “Browsing packages” on page 176
- “Searching for packages” on page 177

- “Viewing targets that have been assigned a package” on page 177
- “Viewing package details” on page 180

## Browsing packages

To view packages that have been assigned to one or more targets, click the Package View tab. The Package View page shows all the packages that have been assigned to one or more targets. (Packages are added to this list whenever you add them to a policy.) When you select a package, you can view its details and the targets to which it is assigned.

---

Tip: Holding your mouse pointer over a package displays its URL.

---

The left side of the Package View page shows a list of packages that have been assigned to targets. The list shows packages and the URLs from which they are obtained. To enter a new package to the list, you must identify it and assign it to a package by creating or editing a policy. See “Creating and editing policies” on page 207.

- By default, the packages are listed alphabetically by package name. Click Show: URL to view package URLs instead of package names.
- A search text box appears near the top of the left side of the page to let you search for a package. You are not required to know the complete name of the package. You can use the asterisk (\*) as a wildcard character to substitute for zero or more characters in the name.

The search is *not* case-sensitive.

- A link appears below the search text box that you can use to change between single and multiple selection modes. You can view the details of one package or several packages simultaneously. Details include all the targets to which a selected package is assigned. To view the details of many packages at the same time, switch to multiple selection mode by clicking on the Multiple link.
- If the list contains more than a predetermined number of items, the **next** link becomes active to let you access other items in the list. Use the **next** and **previous** links to move back and forth through large lists. You can use the numbered drop-down list to speed your navigation through very long lists.

## Searching for packages

You can search for packages that have been assigned to one or more targets. This feature is useful when dealing with large numbers of packages.

### ► To search for a package

- 1 On the left side of the Package View page, enter the name of a package in the Search text box.

You are not required to know the complete name of the package. You can use the asterisk (\*) as a wildcard character to substitute for zero or more characters in the name. If an asterisk is part of the name, you must escape it so that it is not considered a wildcard character. See “Special characters in search strings” on page 171.

The search is *not* case-sensitive.

- 2 Click Go or press the Enter key on your keyboard.

The search results are displayed in the packages list. Click the Reset link to show all assigned packages.

If the list contains more than a predetermined number of items, the **next** link becomes active to let you access other items in the list. Use the **next** and **previous** links to move back and forth through large lists. You can use the numbered drop-down list to speed your navigation through very long lists.

---

Note: Active Directory and ADAM / AD LDS do not display the total number of entries until you have paged through all of the search results. Therefore, the drop-down list displays “...” for the total number of entries until you reach the last page.

---

## Viewing targets that have been assigned a package

The right side of the Package View page shows the targets that have been assigned the packages you select in the packages list. The display differs somewhat depending on the mode you have selected. To switch between modes, click on the Single or Multiple link at the top of the packages list. The following sections describe the two modes:

- “Single selection mode” on page 173
- “Multiple selection mode” on page 174

## Single selection mode

In single selection mode, you can choose a single package and view the targets to which the package has been assigned.

### ► To show the targets that have been assigned a single package

- 1 On the left side of the Package View page, make sure that you are in single selection mode. If you are in multiple selection mode, click the Single link to switch to single selection mode.
- 2 On the left side of the Package View page, locate and select the package for which you want to show targets.

After you click a package, any targets to which the package has already been assigned appear on the right side of the page. The targets that appear might or might not have already received the package shown.

On the right side of the page, the name of the selected package appears at the top. Below the package name is a table that lists all the targets that have been assigned the selected package.

In single selection mode, you have the following options:

**Details View**—Click to view the states and schedules for each package assigned to the target you selected.

**Edit**—Click after selecting check boxes corresponding to the targets for which you want to set states and schedules. See “Creating and editing policies” on page 207.

The following columns appear in the table on the right side of the page:

**Targets**—Shows the names of the targets that have been assigned the selected package. (This is usually the `cn` attribute of the entry in the directory service.)

---

**Tip:** Holding your mouse over a target displays its organizational unit in the directory service.

---

**Primary and Secondary States**—Shows the state or states that determine the way the package is to be distributed to each endpoint. For a list of installation states, see “Overview of installation states” on page 215.

## Multiple selection mode

In multiple selection mode, you can choose two or more packages and view the targets to which they have been assigned.

### ► To show the targets that have been assigned multiple packages

- 1 On the left side of the Package View page, if you are not already in multiple selection mode, switch to multiple selection mode by clicking the Multiple link.
- 2 On the left side of the Package View page, locate and select the packages for which you want to show targets.

Each time you select a package, the *n Package(s) Selected* link, at the top of the right pane, increments to indicate the number of selected packages. Click the link to display a list of the package names. To remove a package from that list, select the check box and click the Remove button.

The right pane contains the list of targets that have been *directly* assigned *all* selected packages (the *intersection* of the policies for the selected packages). If a target has not been assigned *all* selected packages, it does not appear in the list. The targets that appear might or might not have already received the packages shown.

In multiple selection mode, you have the following options:

**Details View**—Click to view the states and schedules for each package assigned to the target you selected.

**Edit**—Click after selecting check boxes corresponding to the targets for which you want to set states and schedules. On the Edit Policy page, you can set states and schedules for the targets that you have selected.

The following columns appear on the right side of the page:

**Targets**—Shows the names of the targets that have been assigned the selected package. (This is usually the `cn` attribute of the entry in the directory service.)

---

Tip: Holding your mouse over a target shows its organizational unit in the directory service.

---

**Primary and Secondary States**—Shows the state or states that determine the way the package is to be distributed to each endpoint. For a list of installation states, see “Overview of installation states” on page 215.

## Viewing package details

The Package Details page shows all targets that have been assigned the selected package or packages, along with the installation priority, primary and secondary states, and schedules for each target.

### ► To view details of a package (or several packages, if you are in multiple selection mode)

- 1 On the Package View page, select a package, or several packages if you are in multiple selection mode.
- 2 Click the Details View link to see the package details.

The Package Details page appears. This page shows the targets to which the package or packages have been assigned, along with the installation priority, primary and secondary states, and schedules for each target.

You can view the details of selected packages or edit their assignment to the targets. Click on the column name to sort by column content. Page through long lists by clicking the previous and next links.

- 3 To return to the Package View page, click the Basic View link.

## Sorting the list of targets

On the Package Details page, you can sort the list of targets based on the following:

- Schedules
- Target name

### ► To sort the list of targets

On the Package Details page, click the column name to sort the targets by that column (either in ascending or descending order).

- If you sort by one of the schedule columns, the targets are arranged according to the activation date and time for that schedule. For example, if you sort by the primary schedule, the targets are arranged according to the activation date and time set for the primary schedule.
- If you sort by the name of the target, the targets are arranged alphabetically.

# Using the Query Builder to search for custom packages

This section includes the following topics:

- “The Query Builder feature” on page 181
- “Configuring the Query Builder” on page 181

## The Query Builder feature

The Policy Manager features a Query Builder option that allows you to search for policies based on criteria such as policy creation and modification time, and installation and expiration time of primary, secondary, update, and verify-repair schedules of packages. You can use this feature to remove obsolete packages or packages that have not been modified for a long time.

## Configuring the Query Builder

You can use the Query Builder to search for policies and packages using the Target View or the Package View.

### Target View

#### ► To search for policies using the Target View

- 1 Click the Advanced Search tab.
- 2 Click the EDIT button on the new Policy Search link.
- 3 On the Advanced Policy Search window that pops up, provide parameters such as the target name, created date, and modified date, and click the OK button.

The policies are filtered the policies based on the provided parameters.

- 4 From the search results, select the policies to remove, for example, packages that are obsolete or packages have not been modified for a long time, and remove them.

You can specify the policy date parameters using one of the following date options:

**Before :** Searches polices created or modified before a specified date.

**After :** Searches polices created or modified after a specified date.

**Between :** Searches polices created or modified between the specified dates.

You can also search for orphan policies and remove them from LDAP. If a machine object is created in LDAP with assigned policies and its LDAP machine entry is deleted without removing the associated policies, the policy objects continue to exist under the subscription container but will not be delivered to the endpoints, because the endpoints no longer exist. Such policies are called orphan policies.

#### ► To search for orphan policies using the Target View

- 1 Click the Advanced Search tab.
- 2 Click the Policy Search link.
- 3 Select the **Orphan Policies Only** check-box to find the orphan policies associated with the target.
- 4 From the search results, select the orphan policies and remove them.

### Package View

The package view includes a link, Advanced Search. On clicking the EDIT button on the Advanced Search link, the Advanced Package Search window pops up, which displays a variety of package, patch group and remediation group search options.

#### ► To search for policies using the Package View

- 1 Click the EDIT button on the Advanced Search link.
- 2 On the **Advanced Package Search** window that pops up, provide parameters such as package, patch group and remediation group search options, and click the OK button.

You can specify one or a combination of the search attributes and filter packages, patch groups, and remediation groups by selecting the respective search attributes from the Select Attribute drop-down box. Specifically, you can specify one or a combination of the following policy attributes:

- Activation and expiration time range of primary, secondary, update and verifyrepair schedules.
- Search for Package Primary and Secondary States
- Search for channel name starts with or ends with or contains of or equals to

- Type can be patch, package, remediation or all
- WoW-enabled or disabled channels
- Exempt from blackout channels or not exempt from blackout channels
- Searching for individual types like patch, package, and remediation separately is recommended as a best practice.



Chapter

# 11 User Centric Deployment

When you use Policy Manager, you can use

The following topics are provided:

- Introduction to User Centric Deployment (page 186)
- Advantages of UCD (page 186)
- A typical scenario where user centric deployment is used (page 187)
- Workflow of UCD (page 187)
- Enabling UCD on an endpoint (page 188)
- Types of user centric deployments (page 188)
- Using a template for UCD (page 191)
- Processing UCD templates at endpoint (page 194)

# Introduction to User Centric Deployment

You can deploy software package deployments on specific endpoints for users when they logon to a computer which is not their originally allotted computer. It is a common scenario where users log on to different computers in different locations for the purpose of work. In this scenario, if a user logs on to different endpoints all channels are subscribed in all the endpoints which has disadvantages like more licenses being used per person and the amount of bandwidth used for deployments. In this scenario, all the packages are subscribed on computers which are not frequently used by the user, which is disadvantageous. However, if you want to specify particular computers to which software packages can be deployed, whenever the user logs on, then you can use the User Centric Deployment (UCD) feature of Policy Manager for deployment. The UCD feature extends the user policies of Policy Manager by classifying the endpoints into different device levels. You can use UCD to specify the deployments for different levels of devices for different users. When a package is deployed on an endpoint for a user, Policy Manager checks the level of the device to ascertain whether the endpoint has the required device level, and performs the deployment.

---

Note: To perform package deployment on an endpoint where the user has logged on, you must ensure that the endpoint is UCD enabled.

---

## Advantages of UCD

The advantages of using UCD feature are:

- The most appropriate software is installed for a user.
- Availability of secondary device for usage.
- Saves network bandwidth.
- Ensures the policy of each user having only one license of a software.
- Floating licenses are efficiently used.

## A typical scenario where user centric deployment is used

User uses one primary device and can access multiple secondary devices and multiple tertiary devices. The software usage behavior of the user indicates the following observations:

- Uses Microsoft Office, Microsoft Outlook and Microsoft Notepad applications.
- Uses Microsoft Office on the primary device.
- Uses Microsoft Outlook on any secondary device which the user accesses.
- Uses Microsoft Notepad on all the tertiary devices.

The BBCA Administrator creates a policy in Policy Manager where the rules are as follows:

- When the user logs on to a primary device, install all the required applications.
- When the user accesses a secondary device, install only Microsoft Outlook.
- When the user accesses a tertiary device, install only Microsoft Notepad.

The above policy ensures that costly Microsoft licenses are not wasted, saves network bandwidth, and ensures that the user can access Microsoft Outlook on all secondary machines accessed.

This type of policy can be implemented only when the all the endpoints are classified into different usage levels like primary device, secondary device, etc. You can use UCD feature to implement this type of policy.

## Workflow of UCD

When a user logs on to any device, the following operations are performed:

- Identify the level of the device which the user has accessed.
- Assign software based on the user profile and the level of the device.
- As per the user's policy, install the required software based on the level of the device.

## Prerequisites

UCD must be enabled on the endpoint.

## Enabling UCD on an endpoint

To enable UCD on an endpoint, ensure to set the *marimba.subscription.ucd.enabled* property to true on the endpoint. By default this property is set to false.

## Types of user centric deployments

The various types of user centric deployments are:

- Static

The static type of deployment ensures that the user and the device level is mapped and set at the endpoint. You can set the property in the Advanced tab of Edit Policy page of Policy Manager. When the policy updates, this property is implemented on the endpoint. For example:

```
ucd.usernamexyz.devicelevel2,service=machineB,machineC
```

This property classifies machine B and machine C as device level 2 for the usernamexyz user.

---

Note: Static mapping takes higher precedence over any other methods of identifying the device level. For example, if this static property is assigned to an endpoint, and if the dynamic method of finding the device level is also enabled, then the static property holds precedence. If you want the dynamic method of identifying device level to hold precedence, then you must remove the static property from the policy.

---

You can also use the bulk or file upload option to upload a CSV file which contains the mappings between the user, device and the level of the device. The CSV format should have the following format:

user, machine name, device level

For example:

```
user_name_xyz, machineA, 1
```

This property means that user\_name\_xyz is assigned machine A which has a device level of 1.

- Dynamic or Template

To implement UCD using a template, you can define the required templates in Policy Manager, publish it to the Policy plug-in and publish it to all endpoints when the Policy Service updates. You can configure templates for each channel in the Edit Policy page for each user or user group. The channel to templates mapping is many-to-many mapping. A channel can have any number of templates, and similarly a template can be added to more than one channel. This method of implementation has the least priority. For example, if no static mapping is performed and no custom channel URL is provided to find a device level for that endpoint, then the template method is implemented.

- Custom channel

You can write a custom Marimba channel to find the device level for the currently or last logged-on user for a computer. When the device level information is retrieved, it is written in the channel property. Policy Service reads this property and uses the device level as the device level for that user and installs the required packages or channels.

Once you implement a custom channel to retrieve the device level value and update the `custom.ucd.devicelevel` property in the `channel.txt` file, Policy Service reads the value of this property. If Policy Service does not find the value of this property, it returns -1 as the default value. When Policy Service returns -1, all channels applicable for the logged-on user are installed in the computer. Custom channel has higher precedence than the template which is specified. You can only assign a numeric value for this property.

For example: `custom.ucd.devicelevel=1`

---

Note: When the device level for a computer and its level is obtained, the device level information is stored in the `device_level` column of `machineperson` table. You can use the `inv_person` view to generate device level reports for a computer.

---

## Prerequisites for endpoints

Ensure the following on the endpoint:

- The `marimba.subscription.ucd.enabled` property is set to true.
- Audit logon events security setting is enabled for success.
- Logged on user has administrator rights.

- User Account Control (UAC) is disabled.
- Endpoint can run VB-Script.

## Configure device level

You can specify the maximum device levels for identification level type. Based on this device level value, you can set the level for static and dynamic device level identification type.

### ► To configure device level

- 1 In the Configuration page of Policy Manager, click the **User Centric Deployment** link. The User Centric Deployment Options page appears.
- 2 Select **Using file upload** option.  
The File Upload Identification Type section appears.
- 3 Select the required device level from the list box.  
The minimum level value is 2. The maximum level is 5.
- 4 Click **Save** to save the device level.

## Using the file upload feature of UCD

When you have very few UCD properties to set for endpoints, for each user you can set the properties in the Advanced tab of Edit Policy page of Policy Manager. However if you have many users for whom you have to set the UCD properties, then you can use the file upload feature of UCD.

### ► To upload UCD mappings in bulk:

- 1 In the Configuration page of Policy Manager, click the **User Centric Deployment** link. The User Centric Deployment Options page appears.
- 2 Select **Using file upload** option. The File Upload Identification Type section appears.
- 3 Click **Browse** and navigate to select the CSV file which contains the mappings between the users and device levels.
- 4 Click **Update Policy**.

The mappings are applied in the policies. Once the mappings are applied, the User Centric Deployment Options page displays a status section.

- 
- 5 If you want to view the status, click View Status.

The File Upload Status dialog appears which shows the status of the applied mappings. This dialog displays appropriate messages for the following scenarios:

- Invalid mappings
- Invalid users
- Invalid device levels
- Duplicate device levels for the same user
- Failure to update the policy for a user

---

Note: You can check whether the UCD properties have been applied or not by checking the tuner properties in the Advanced tab of Edit Policy page for a user.

---

## Using a template for UCD

If you want to define a template where you can identify and map the device levels to users, then you can use the template feature of UCD where you can custom define the settings for identifying the mappings between device levels and users. Once you create the template, it is distributed to the endpoints on policy service update.

When you provide a name for the template, ensure that the name does not contain any special characters.

### ► To create a template to dynamically custom-define the mappings

- 1 In the Configuration page of Policy Manager, click the User Centric Deployment link. The User Centric Deployment Options page appears.
- 2 Select **Using Template** option. Type section appears where you can add, edit or remove templates.
- 3 To create a new template, click **Add**. The Add UCD Template page appears.
- 4 In the **Template Name** text box, type the name of the template.
- 5 In the **Template Description** text box, type the description of the template.
- 6 In the **Device level** list box, select the device number.

7 Specify or select the following parameters:

- Device level
- Consider Login history
- Total number of hours logged on
- Number of days to be considered
- Consider device lock and unlock time
- Consider remote login mode
- Consider device configuration
- Ram Size
- Disk Space
- OS Name
- Processor Name

8 Click Publish.

Once the template is published, the published templates are stored as part of the configurator segment of Policy Service on Master Transmitter. The Policy Service URL is retrieved from Policy plug-in publish UI. If Policy plug-in is not yet published, then the UCD templates are not published and an error message displays which indicates failure to publish policy plug-in. You must republish the plugin whenever you add or modify the template.

Once the templates are published, these templates are downloaded to the endpoints and applied on the endpoint profile. The applicable templates are identified and the device levels of applicable templates are sent to plugin for filtering channels.

## Assigning templates to channels

You can assign the UCD template to the channels on the Edit Policy page. Once you assign a UCD template to a channel, the device levels are stored in LDAP on newly created Subscription attribute for the UCD template. The last column on the Edit Policy page displays the Device Level which are assigned for each channel. You can click on Set Device Level button to modify, add, or delete device level assigned to the channels. Once you click Set Device Level button, the Select Device Level dialogue appears and you can choose using template option which shows the list of UCD templates. You can choose the required template and click on Save button to assign the template for the selected channels. You can assign UCD template to more than one channel by selecting the check boxes of the required channels.

---

Note: The Set Device Level button is enabled only for the targets types user and user groups. The other targets like container, collection, machine are not supported. It is recommended to have any user group, targets should contain only users as its members. If machines are also members, then the filtering channels based on device level will be applied for the machine targets also even though which is not expected.

---

## Assigning static device level to channels

You can assign the static device level to the channels on the Edit Policy page. Once you assign a static device level to a channel, the device levels are stored in LDAP on newly created Subscription attribute for the UCD template. The last column on the Edit Policy page displays the Device Level which are assigned for each channel. You can click on Set Device Level button to modify, add, or delete device level assigned to the channels. Once you click Set Device Level button, the Select Device Level dialogue appears and you can choose using static device level option which shows the list of device level values. You can choose the required device level and click on Save button to assign the device level for the selected channels. You can assign device level to more than one channel by selecting the check boxes of the required channels.

## Processing UCD templates at endpoint

The UCD templates are downloaded to endpoint and applied against the endpoint data and the applicable templates are identified. The device levels of applicable templates are sent to plugin for filtering channels.

To update policy service on user logon event, set the following properties:

- logon.notify=true
- logon.action=update

The applicable templates are identified in the following way:

The conditional operator applied is AND. When all the configured parameters are satisfied with the given values, then that template is applicable for that endpoint for that logged-on user.

The Login History and Device Configuration options are used to find the device level for that machine. Any one of these options or a combination of these options can be used for identifying the machine level. This template is applicable for those endpoints only when all the following conditions are true:

- Login History Calculation:

The total number of hours that the user logged on to that endpoint for the configured number of days should be greater than the configured number of hours. Then this condition is evaluated as true, otherwise it is evaluated as false and quits.

(total no\_of\_hours logged in) > configured no\_of\_hours

Users' logon history could be taken from the System Events in case of Windows endpoints. For XP, Event codes 528, 551 are for logon and logoff code respectively. For Windows Vista and 7, the respective codes would be 4624, 4647. And logon type 2 is for interactive mode (direct machine login) and 10 is for remote interactive (through mstsc or so). The parameter 10 would be useful to filtering out the logon events through remote login.

- Consider Device Lock and unlock time

This option could be enabled, only when the ‘Login History’ option is enabled. If this check box is enabled, then the machine’s locked period for that duration should be calculated and that should be subtracted from the total login period. This option is used to calculate the effective login period for that endpoint. And, the login history calculation would become TRUE, when the following condition for configured number of days data is considered,

(total no\_of\_hours logged in – machine locked period) > configured no\_of\_hours

This information is found and calculated by Policy Service from the moment, it is subscribed to endpoint tuner. This will be calculated even though this feature is not enabled, since these data are required once the UCD is planned for the environment. These data are stored in a file named “machlockdb.txt” in data directory of policy service channel. The values are stored in comma separated values.

This file contains the values lock id, username, lock time, unlock time and locked duration by calculating the difference between lock time to unlock time.

Note: Lock and Unlock events could be captured only when tuner runs in service mode. The following properties should be enabled at endpoint

To capture endpoint machine lock/unlock details, the following properties should be set with the same values.

```
mclock.notify=true  
lock.action=start  
mcunlock.notify=true  
unlock.action=start
```

All above properties should be set in channel.txt of Policy Service. Also, these properties could be set to through Service property under Advanced Tab of Policy Edit Page.

- Consider Remote Login Mode

In some infrastructure, non-primary machines could be present on Data Center and accessed through remote login mode. If a user logged in to these machines in remote login mode, then these machines need not to be identified as primary machine. And, in some other organizations, the primary machines could be given from Cloud and a dump terminal could be given for remote access of the actual machines. Here, in this case, the remote login should be considered for primary machine identification. This option would help us to calculate the login history more effectively according to the different environments. This remote login mode, option could be enabled only when the 'Login History' option is enabled.

### How to enable settings to capture Logon events on success cases

- 1 Open Run from start menu and issue command gpedit.msc
- 2 Navigate to the path "Computer Configuration -> Windows Settings -> Security Settings -> Local Policies -> Audit Policy"
- 3 Double click on "Audit Logon Events"
- 4 In the "Local Security Setting" tab, enable "Success" check box
- 5 Save the settings.
  - Consider Device Configuration

At some infrastructure, primary device could be identified with the device configurations. Here, the parameters used are RAM Size, Disk space are greater than the given values and the OS and processor are equal to the chosen value, then we could identify this particular template is valid for that end-point. These parameters could be identified in different ways for different flavors of desktop machines.

## Troubleshooting

### Login History Calculation

For finding logon history for the user, we use windows Event Logger service. This service, logs every logon and logoff info with time and type of logon in the event's list. The implementation is to issue a SQL query to get a list of events logged in for the specified time frame. This time frame would be given as a command line argument to cscript command and file name. Following is the actual command executed on Windows desktop boxes to collect the logon history of all users. The start date and end date should be given in the format 'yyyy/mm/dd' and the remote login flag take the value True or False.

Syntax:

```
cscript //nologo <vb script file with extension> <start date> <end date>  
<remote login>
```

Example:

```
cscript //nologo UserLogonDuration.vbs 2013/09/01 2013/09/21 True
```

The output of above command will be in the format of comma separated values and stored as a csv format file inside the data directory of Policy Service. This csv file would be saved in the name of the template name. And, then this file is parsed for the respective user and the logon history is calculated as per the template values. This command could be invoked directly on a command prompt at the debugging front.

The script files used here are moved inside data directory of Policy Service by Policy Service channel itself. A new folder by name “scripts” would be created inside data directory and the script files would be moved inside it. The script file “UserLogonDuration.vbs” for Windows Vista and Windows 7 boxes and file “UserLogonDurationXP.vbs” for Windows XP boxes.

#### ■ Device Configuration

The following commands for the supported desktops are listed below for debugging purposes. The following are the set of commands used to get device configuration for windows machines,

- OS Name: wmic OS get name
- Processor Name: wmic cpu get name
- Disk Capacity: wmic diskdrive get size
- RAM Size (Non-XP): wmic memorychip get capacity
- RAM Size (XP): wmic computersystem get TotalPhysicalMemory

These commands could be used at the time debugging. If any of the templates gets failed, these set of commands could be issued at the respective endpoint to collect the data. We can manually evaluate the template values along with the result of these commands’ output.



# Chapter 12 Blackout Period

When you use Policy Manager, you can use

The following topics are provided:

- Setting a blackout period for a target (page 200)
- Exempting packages from the blackout period (page 204)
- Configuring blackout priorities (page 205)
- Setting blackout priorities (page 204)

# Setting a blackout period for a target

This section includes the following topics:

- “What is a blackout period?” on page 200
- “Setting the blackout period for a target” on page 201

## What is a blackout period?

A *blackout period* defines a range of time when you want to prevent Policy Management from changing the state of packages on one or more targets. The blackout period prevents all downloads, installations, updates, and repairs from occurring during the specified period of time. You can also specify no blackout period if you want activities to take place at any time. See “Setting the blackout period for a target” on page 201.

---

Note: During the specified blackout period, no activities will take place for all packages on the endpoint tuner. This is true even for packages that are not managed using Policy Manager. Take this information into account when you set the blackout period because there may be emergency situations when you need to update a package on endpoints during a blackout period. You will either need to wait until the blackout period is over, or manually perform updates at each endpoint.

---

## Setting the blackout period for a target

Follow these steps to specify a blackout period that is used by default to prevent policy state changes on one or more targets. The blackout period prevents downloads, installations, updates, and repairs from occurring during the specified period of time of the day. Usually, you'll want to set a blackout period to avoid possible interruptions to workday activities at the endpoints or to minimize network traffic during critical periods. Activities that are scheduled during a blackout period are performed immediately after the blackout period is over. The blackout period is relative to the endpoint's time zone.

**Global blackout period.** To set a global blackout period, select the All Endpoints target before you specify the blackout period. If two blackout periods are set for an endpoint (as a result of two policies resolving to the same endpoint), the endpoint's blackout period is a union of the two blackout periods. For example, user Alan belongs to the Engineering user group and a blackout period is set for both. The blackout period for Alan is 7 AM to 11 AM, while the blackout period for Engineering is 9 AM to 5 PM. The resulting blackout period for Alan is 7 AM to 5 PM.

**Exempting Policy Service from the blackout.** When you set a blackout period, you can exempt Policy Service so that it can still update and run during the blackout period. This exemption is useful if you anticipate emergency situations when you want to update or install packages during the blackout period.

**Exempting packages from the blackout.** On the Packages tab of the Edit Policy page, you can indicate which packages you want to exempt from the blackout period. This exemption is useful if you anticipate emergency situations when you want to update or install packages during the blackout period. See “Exempting packages from the blackout period” on page 204.

When multiple users (using the same Policy Manager from two different machines) edit the Blackout Period page for the same target, the most recent edits to be saved are applied.

---

Note: You can set target-level settings, such as the blackout period, for one target at a time only. You cannot set target-level settings if you are in multiple-selection mode and are editing policies for multiple targets at one time. However, you can set target-level settings for all targets by selecting the All Endpoints target.

---

This procedure assumes you are on the Packages tab of the Edit Policy page. See “General directions for creating and editing policies” on page 209.

## ► To set the blackout period

- 1 On the Edit Policy page, click the Blackout Period tab.

The Blackout Period page appears.

- 2 Choose one of the following options in the Blackout period area:

- If you want to set a blackout period, choose Set the following blackout period, and, in the Disable activities section, specify the starting and ending times for the blackout period.

Note: You can set a blackout period that spans one day (such as from 9 AM to 5 PM), but not one that spans midnight (such as from 6 PM to 5 AM). You can also select the days of the week for the blackout period.

- If you do not want to set a blackout period, choose None. Allow activities to take place any time.

- 3 If you specified a blackout period for the target, choose one of the following options in the Policy Service behavior section:

- Enable Policy Service to update during the blackout period and allow operations for packages that are exempt from the blackout period.

- Disable Policy Service from updating during the blackout period. No operations will be performed for packages, even those that are exempt from the blackout period.

- 4 When you are finished creating or editing the policy, click Preview.

The Preview page allows you to review changes to the policy before you save them and apply them to the target. See “Previewing and saving policy changes” on page 211.

- 5 After reviewing your changes to the policy, click Save to confirm your changes and save them.

## Details about the blackout period

When you specify a blackout period, Policy Service appends the time period you specify to the tuner property `marimba.schedule.filter` in the endpoint tuners' `prefs.txt` file. For example, if an endpoint tuner's original update schedule for channel is anytime on any day of the week, and then you use Policy Management to set a blackout period from 9 AM to 5 PM, the tuner property is set to the following value:

```
marimba.schedule.filter=ANYTIME on mon+tue+wed+thu+fri+sat+sun  
BLACKOUT 9:00AM-5:00PM
```

If you later select the No blackout period option, the original channel update schedule set at the endpoint tuner is used. In the example given, only `BLACKOUT 9:00AM-5:00PM` is removed from the value for the `marimba.schedule.filter`. The tuner property that affects the blackout period (`marimba.schedule.filter`) is not deleted from the endpoint. You can delete that tuner property explicitly using the Tuner and Package Properties page. See “Setting tuner and package properties for a target” on page 275.

## Exempting packages from the blackout period

When you add packages to a policy, you can indicate which packages you want to exempt from the blackout period. This exemption is useful if you anticipate emergency situations when you want to update or install packages during the blackout period.

This procedure assumes you are on the Packages tab of the Edit Policy page. See “General directions for creating and editing policies” on page 209.

### ► To exempt packages from the blackout period

- 1 On the Packages tab of the Edit Policy page, locate the rows of the packages that you want to exempt from the blackout period.
- 2 In the rows of the corresponding packages, select the check boxes under the Exempt from Blackout column.

In order for these exemptions to take effect, you must also exempt Policy Service from the blackout period, so that it can obtain and apply policy updates. You set the option for exempting Policy Service when you set the blackout period. See “Setting the blackout period for a target” on page 201.

When you are finished editing the policy, preview and save the policy. See “Previewing and saving policy changes” on page 211.

## Setting blackout priorities

This section includes the following topics:

- “What are blackout priorities?” on page 204
- “Configuring blackout priorities” on page 205

## What are blackout priorities?

In earlier versions of Policy Management, if an endpoint received more than one blackout schedule via different group memberships, you could not control which blackout schedule to apply to that endpoint. One of the blackout schedules was randomly assigned, and you did not know which schedule was assigned.

With version 8.2.00, you can prioritize the blackout schedule(s) that are

applied to an endpoint. When Policy Manager encounters an endpoint that has more than one blackout schedule, it scans the blackout schedule priorities specified for that endpoint, and applies the blackout schedule with the highest priority to that endpoint.

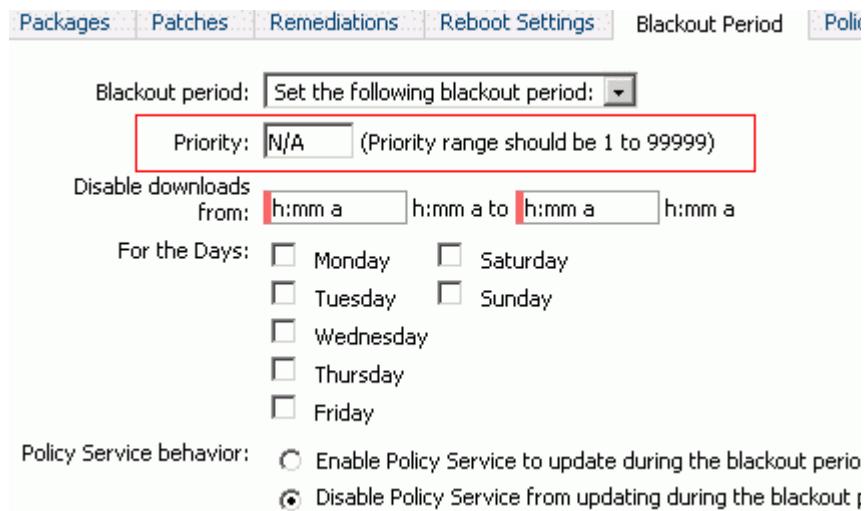
## Configuring blackout priorities

On the Blackout Period tab in Policy Manager, as shown in Figure 12-1, specify a priority value from 1 to 99999, with the lower value indicating a higher priority.

If you do not specify a priority, the default priority of 99999 is used.

The Policy Service sorts the blackout schedules (on LDAP) for the endpoint, in order of decreasing priority, and applies the highest priority blackout schedule to that endpoint. If blackout schedules have the same priority, a blackout schedule is randomly assigned, similar to earlier versions of Policy Management.

Figure 12-1: Specifying blackout priorities





# Chapter 13 Creating and editing policies

A policy is the association between one target, one or more packages, and the scheduled delivery of the packages to the targets.

The following topics are provided:

- What is a policy? (page 208)
- General directions for creating and editing policies (page 209)
- Adding and removing packages from a policy (page 213)
- Specifying states and schedules for packages in a policy (page 215)
- Creating WoW deployments (page 219)
- Scheduling wake-up start and interval times using the PC Alarm Clock feature (page 247)
- Setting the install priority for packages in a policy (page 248)
- Copying policies (page 253)
- Deleting policies (page 254)
- Peer Approval Policy (page 257)
- Managing software, data, and updates (page 260)
- Editing policies from Package View (page 264)
- Specifying policies for OS migration (page 266)
- Specifying personal backup settings for OS migration (page 267)
- Setting a blackout period for a target (page 195)
- Specifying the Policy Service schedule for a target (page 270)
- Setting tuner and package properties for a target (page 275)

- Specifying transmitter permissions for a target (page 308)
  - Specifying the profile for a target (page 310)
  - Provisioning unprovisioned Intel AMT vPro computers (page 313)
- 

**WARNING:** Each time you log in to the console and start using Policy Manager, you begin a *browser session*. Many of the Web pages associated with a session depend on your having visited previous pages that set variables in the session. For this reason, you should not specify Web pages directly by typing in a URL, following bookmarks, or using the browser's Back and Forward buttons during a session. This behavior is typical of browser-based applications.

---

## What is a policy?

A policy specifies which packages are delivered to which endpoints, at what time. It also specifies a primary state (and optionally, a secondary state) for each package. You can set schedules for the primary state and secondary state, as well as schedules for updates, verification, and repairs.

A policy can also include the following information:

- Tuner and package properties
- Policy Service update schedule
- Blackout period and exceptions to the blackout period
- The profile associated with the target
- Transmitter permissions associated with the target

All information in a policy is stored in a directory service. It is applied to each endpoint when the endpoint contacts the Policy Service plug-in on the transmitter to check for policy updates.

# General directions for creating and editing policies

You can create a policy by choosing a target and specifying the packages that the target should have. For each package, you can specify the following information:

- Primary and secondary states
- Primary, secondary, update, and repair schedules
- Installation priority
- Blackout exemption
- The states and schedules for the packages.
- The blackout period for the target and exempting Policy Service from the blackout period.
- The update schedule for Policy Service on the target.
- Tuner and package properties that you want to set on the target.
- Transmitter permissions required for the target.
- Profile information for the target.

---

Note: If you are in multiple selection mode and are editing the policies for multiple targets at one time, you can only edit the packages and package information, such as states, schedules, install priority, and blackout exemptions. The exemptions apply to all specified targets. The tabs for other information, such as blackout schedule and Policy Service update schedule, appear only when you are editing the policy for a single target.

---

The procedure that follows describes how you can create or edit a policy starting from the Target View, Target Details, Package View, and Package Details pages. The first step differs, depending on which page you use, but the remainder of the steps are the same.

## ► To create or edit a policy

- 1 Choose one of the following options:
  - Click the Target View page.
  - Click the Target Details page.

- Click the Package View page.
  - Click the Package Details page.
- 2 On the Policy Manager Target View page, locate and select the target for which you want to create or edit a policy and then click Edit.
- The Edit Policy page appears, and in Packages tab shows any packages assigned to the target.
- 3 To add or remove packages from the policy click Edit Package List.
  - a To add a package, see “Adding packages to a policy” on page 213.
  - b To remove a package, see “Removing packages from a policy” on page 214.
  - c When you are finished adding and removing packages, click OK.
- 4 On the Edit Policy page on the Packages tab, specify states, schedules, and whether you want to create a Wake On Wan (WOW) deployment for the packages that you added to the policy. See “Specifying states and schedules for packages in a policy” on page 215 and “Creating WoW deployments” on page 219.
- 5 To specify the install priority for the packages in the policy, see “Setting the install priority for packages in a policy” on page 248.
- 6 To specify the packages that you want to exempt from the blackout period, see “Exempting packages from the blackout period” on page 199.
- 7 To specify reboot settings for the policy, see “Specifying reboot settings for Windows targets” on page 273.
- 8 If you are creating or editing the policy for a single target, you can also complete the following tasks:
  - “Setting a blackout period for a target” on page 195
  - “Specifying the Policy Service schedule for a target” on page 270
  - “Setting tuner and package properties for a target” on page 275
  - “Specifying transmitter permissions for a target” on page 308
  - “Specifying the profile for a target” on page 310
- 9 When you are finished creating or editing the policy, click Preview.

The Preview page allows you review changes to the policy before you save them and apply them to the target. See “Previewing and saving policy changes” on page 211.

- 10 After reviewing the policy, click Save to confirm your changes and save them.

The next time that Policy Service updates and runs on the endpoint, it downloads and applies the policy that you created or edited.

## Editing a policy for multiple packages (Edit All)

You can change package states, select or deselect the WoW Deployment option, and select or deselect the Exempt from Blackout option for multiple packages using the Edit All option.

### ► To edit a policy for mutiple packages

- 1 From the Packages tab of the Edit Policy page, click the Edit All button.  
A pop-up window displays listing the packages an options that you can edit.
- 2 On the left, select the packages that you want to change.
- 3 Make your changes for each package.

You can change:

- Primary State
- Secondary State
- Exempt from Blackout
- WoW Deployment

- 4 Click Save.
- 5 When you are finished creating or editing the policy, click Preview.
- 6 After reviewing the policy, click Save to confirm your changes and save them.

The next time that Policy Service updates and runs on the endpoint, it downloads and applies the policy that you created or edited.

## Previewing and saving policy changes

Policy Manager requires you to preview policy changes before you can save them. This enables you to make sure that the correct settings are applied to the target.

Note that when you save a policy, you are saving edits across all tabs. Make sure you review your changes in all the tabs before you save.

The Preview page parallels the Edit Policy page. The Preview page contains the following four tabs, each corresponding to a tab in the Edit Settings page:

- **Packages**—Shows the packages in the policy, including information for each package, such as states, schedules, and install priority.
- **Blackout Period**—Shows the blackout period for the target.
- **Policy Service Schedule**—Shows the schedule for updating Policy Service.
- **Power Options**—Shows the selected power options for the endpoints.
- **Advanced**—Shows advanced settings, such as the tuner and package properties, transmitter permissions, and profile information for the target.

## ► To preview your changes

- 1 On the packages tab of the Edit Policy page, click Preview after editing a policy.

Note: When you click Preview on the Edit Policy page, the Preview page appears. Make sure you click on each tab to preview the changes made to those sections.

- 2 Preview each tab. the Save button applies all tabs, not just the open tab.
- 3 Choose one of the following options:
  - If you want to save the policy, click Save to confirm your edits and save them.
  - If you do not want to save your changes and want to continue editing the policy, click Back to Edit.
  - At any point, if you want to discard your changes and return to the Target View page, click Cancel.

## Notes about saving policies

You cannot save policies if you do not have write permissions to the child container on the directory service. The child container was created when you (or your primary administrator) set up Policy Manager.

If you are using Active Directory and the policy you saved was for a target located in a site different from where the policies are stored, the policy takes effect when policies are replicated to the Global Catalog nearest the plug-in.

## Adding and removing packages from a policy

This section includes the following topics:

- “Adding packages to a policy” on page 213
- “Removing packages from a policy” on page 214

### Adding packages to a policy

This procedure assumes you are on the Edit Package List page. To get to that page, click the Edit Package List button on the Packages tab of the Edit Policy page. See “General directions for creating and editing policies” on page 209.

#### ► To add packages to a policy

- 1 On the left side of the Edit Package List page, under Select Package, choose one of the following options:
  - Click the Currently Deployed tab if you want to add a package that has already been assigned to a target.
  - Click the Transmitters tab if you want to choose a package that has been published to a transmitter, but not yet assigned to any target. You can browse the list of packages on a transmitter by entering the transmitter’s URL and clicking Go.

When you select a folder containing packages from the Transmitter tab, Policy Manager displays a pop-up dialog asking you if you want to add all of the packages under the selected folder to the Package List.

Each selected package appears on the Packages List in the right pane. To remove a package from the list, see “Removing packages from a policy” on page 214.

- 2 Click OK to save your changes and return to the Edit Policy page.
- 3 When you are finished editing the policy, preview and save the policy. See “Previewing and saving policy changes” on page 211.

## Removing packages from a policy

This procedure assumes you are on the Edit Package List page. To get to that page, click the Edit Package List button on the Packages tab of the Edit Policy page. See “General directions for creating and editing policies” on page 209.

### Removing packages from a policy and from targets

When you delete a package from the policy, the package is deleted from the target the next time that Policy Service updates, with the following exceptions:

- The state of the package is set to `install-persist` or `install-start-persist`.
- The tuner property `No Delete (marimba.subscription.nodelete)` is set to `true`.

For either exception, the package remains on the targets, but the package is no longer managed through Policy Manager. To actually delete these packages from the targets, you must explicitly set the package’s primary state to `uninstall`.

---

Note: When you remove a channel from a policy in the Packages tab, the channel properties shown in the Advanced tab are automatically removed from the channel.

---

#### ► To remove packages from a policy

- 1 In the right pane of the Edit Package List page under the Package List pane, select the packages you want to remove from the Package List.
- 2 Click Remove.
- 3 Click OK to save your changes and return to the Edit Policy page.

When you are finished editing the policy, preview and save the policy. See “Previewing and saving policy changes” on page 211.

# Specifying states and schedules for packages in a policy

This section describes how you can set states and different types of schedules for packages in Policy Manager. It includes the following topics:

- “Overview of installation states” on page 215
- “Setting the primary and secondary states” on page 218
- “Overview of schedules” on page 231
- “Setting the primary and secondary schedule for packages” on page 237
- “Setting the update schedule for packages” on page 238
- “Setting the repair schedule for packages” on page 241
- “Conflict resolution: states and schedules in policies” on page 244

## Overview of installation states

Installation states determine how, and if, a package is installed on a target. For each package assigned to a target, you can set a primary state. Using Policy Manager, you can assign installation states to a package as part of the policy for one or more endpoint targets.

You can control how a package is installed by setting a secondary state to some of the primary states, such as *stage* and *advertise*.

You can associate an activation schedule with the secondary state to specify a window of time in the future when package installation should occur across your organization, even for endpoint targets that connect to the network infrequently. The default secondary state is *install*.

Some states take precedence over others. See “State precedence” on page 217.

Table 13-1 describes the states that are available for packages, and the secondary states that can be associated:

Table 13-1: Installation states for packages

Primary state	Description	Secondary state
Stage	Downloads a package but does not install it. The user at each endpoint can perform an interactive installation, or, if you set a secondary state, you can schedule a time when the downloaded package is installed automatically. (This state might appear as <i>install-pending</i> when the tuner is viewed using Tuner Administrator. This state was called <i>subscribe and no install</i> in a previous release.) <sup>a</sup>	Install Install-Start Install-Persist Install-Start-Persist
Advertise	Describes the package as available to be downloaded. If you set a secondary state, you can schedule a time when the download and installation actually takes place. (This state might appear as <i>available</i> when the tuner is viewed using Tuner Administrator.)	Install Install-Start Install-Persist Install-Start-Persist
Install	Downloads and silently installs a package. (This state was called <i>subscribe</i> in a previous release.)	Uninstall
Install-Start	Installs a package, and starts the installed software. The package starts every time Policy Service runs. If the package is already running, Policy Service does not restart it. (This state was called <i>subscribe and start</i> in previous releases.)	Uninstall
Install-Persist	Installs a package and does not permit it to be deleted even if the endpoint is removed from a target group. This state is useful for installing system software upgrades. The package can still be deleted with the uninstall state. (This state was called <i>subscribe-persist</i> in a previous release.)	N/A
Install-Start-Persist	Installs a package, starts the installed software, and does not permit it to be deleted even if the endpoint is removed from a target group, as with the install-persist state. The package starts every time Policy Service runs. If the package is already running, Policy Service does not restart it.	N/A

Table 13-1: Installation states for packages (Continued)

Primary state	Description	Secondary state
Exclude	<p>Excludes a package from an endpoint even if it is part of a subscribed group. This state is useful when you want to subscribe a large group but exclude a small subset. This state has the highest precedence. (See “State precedence” on page 217.)</p> <p><b>Note:</b> To use the exclude state, you must create a policy that specifies the endpoints to exclude from receiving the package, as well as identifying the larger group to the package.</p>	N/A
Uninstall	Deletes the package. If the installed software is running under control of the tuner, it is stopped without warning and then deleted. Note that this installation state deletes packages that were installed with persistence. (This state was called <i>delete</i> in previous releases.)	N/A
Primary	Downloads the package and designates it as the tuner’s primary channel—one that is started each time the tuner starts and often serves as the GUI for the tuner.	N/A

<sup>a</sup>. To use the stage state to download entire files without installing them, set both the `preload` and `delayfiledownload` parameters to false in the package. For more information see the *BMC Marimba Client Automation Application Packager User Guide*, available on the BMC Customer Support website.

## State precedence

Conflicts between package states can occur when an endpoint is present in two or more groups, or when a policy has been assigned to both a single endpoint and the members of a group to which the endpoint belongs. Policy Manager enforces a precedence to each state and enforces the state with the highest precedence. In such cases, the order of precedence is as follows:

- |                        |                       |
|------------------------|-----------------------|
| 1 (highest precedence) | Exclude               |
| 2                      | Uninstall             |
| 3                      | Primary               |
| 4                      | Install-Start-Persist |
| 5                      | Install-Start         |
| 6                      | Install-Persist       |
| 7                      | Install               |

8	Stage
9 (lowest precedence)	Advertise

---

Note: Conflicts can occur between package states that are not caused by the way a policy is targeted. Therefore, conflict resolution is not based on whether an endpoint is targeted directly or indirectly as a member of a group. See “Conflict resolution: states and schedules in policies” on page 244.

---

## Setting the primary and secondary states

The *stage* and *advertise* primary states let you set a secondary state to control how a package is installed. You can associate an activation schedule with the secondary state to specify a window of time in the future when package installation should occur across your organization, even for endpoints that connect to the network infrequently. The default secondary state is *install*.

For information about primary and secondary states, see “Overview of installation states” on page 215.

### ► To specify the primary and secondary states for a package

- 1 On the Packages tab of the Edit Policy page, locate the row of the package for which you want to set primary and secondary states.

See also “General directions for creating and editing policies” on page 209.

- 2 In the package row under the Primary State column, choose a state from the drop-down list.
- 3 If a secondary state is applicable, select a state from the drop-down list under the Secondary State column. The default secondary state is *install*.
- 4 Set the schedule if a secondary state is specified.

See “Overview of schedules” on page 231.

## Creating WoW deployments

As an initiative of Green IT, endpoint users shutdown their PCs during non-working hours. Because of this, all the deployments scheduled while the PCs are powered off resume when the users turn on their PCs at the beginning of a working day. This causes high usage of network bandwidth due to downloading of channels as specified in the policy.

To avoid this network congestion, you can use the Policy Management Wake-on-Wan (WoW) feature to wake up machines during non-peak network periods so that deployments can be made reliably and without network disruption. The WoW feature runs in CMS as a service and for each WoW task in CMS, the WoW feature creates a unique Task ID. You can execute WoW deployments using the following channels:

- Subscription Manager
- Report Center
- Infrastructure Administrator

You can also use the WoW feature in Infrastructure Administrator and Report Center to wake up endpoints. However, you can use Report Center to wake up only a single endpoint.

### WoW prerequisites

- The network adapter of WoW target machines must support the Wake-On-LAN (WoL) feature.
- The WoW seed tuners must be of version 7.5 or later.
- Ensure that the Schema Manager is updated to 8.2.01 or later, because the 8.2.01 schema contains WoW specific tables and views.
- After the schema is updated, ensure that you perform the inventory scan. The inventory scan populates the database with the data required (details of WoW target computers and seed tuner computers) for a WoW deployment.

---

Note: The WoW feature cannot wake-up computers that shutdown abnormally. For example, when you perform force shutdown on a computer, you cannot wake-up the machine using the WoW feature.

---

## Architecture of Hazelcast framework for WoW

CMS uses the Hazelcast framework to implement the distributed WoW feature. The Hazelcast framework allows you to distribute the CMS WoW task load to other tuners for execution, and thus avoids Out of Memory errors in the CMS tuner. BBCA achieves this by using the Worker channel called the Task Executor Service. The communication between worker channels takes place either by multicast or TCP/IP communication protocols. When workers are unavailable, CMS executes the WoW task.

### Worker channel - Task Executor Service

You can use the Worker channel to execute distributed WoW tasks. Once the CMS channel uploads the WoW tasks in the Hazelcast's distributed queue, CMS notifies about the queued task. The WoW task is then distributed across the worker channels in first-in-first-out (FIFO) order. When the task is distributed to worker channels, based on the availability of the threads, WoW execution (finding a seed tuner and sending magic packets for that subnet) starts and the status is sent back to CMS.

The Task Executor Service channel executes WoW task in another tuner.

Note:

- It is not recommended to run the Task Executor Service in the tuner where CMS is running.
- Ensure that the worker tuner is 8.2.01 or above, and the Task Executor Service channel is subscribed. Task Executor Service is an auto-start channel and must be running to execute WoW task.

Once the WoW task starts, the subnet information is collected and added in the Hazelcast distributed task queue. The Distributed Task Queue stores WoW tasks information like subnets, subnet's status, cache machines and subnet cancelling status. The queue distributes the tasks among all the Worker channels. The Worker channels perform the WoW task allocated to them and then sends back the WoW status reports to the CMS.

### Configuring distributed WoW task properties

To configure distributed WoW task properties, you can set the following properties in the tuner's prefs.txt file in CMS and in the Worker tuner where the Task Executor Service is run:

- marimba.task.distribution.enabled

To enable the CMS to upload WoW tasks to the distributed task queue, set this property to true. If no workers (Task Executor Service) are available in the network to run the WoW task, CMS runs the WoW task.

**Note:** You must set this property for all tuners which run the Task Executor Service channel.

■ `marimba.task.distribution.groupname`

You can set this property to create a logical cluster group with a customized name. BBCA uses this name to create a cluster between CMS and Worker tuners for communication. By using this property, you can create multiple groups across the network.

Default value: `Marimba_task_distributor`

■ `marimba.task.distribution.password`

You can use this property to set the password which will be used by the workers to join the cluster group.

**Note:** The password is stored in a tokenized or encrypted format.

■ `marimba.task.distribution.port`

You can use this property to specify the port to be used for communication between CMS and Task Executor Service. This port is used by Hazelcast framework for communication.

Default value: 5701

■ `marimba.task.distribution.port.autoincrement`

You can use this property to enable or disable auto-increment of the port number, so that you can run multiple workers in a single computer.

Default value: False

■ `marimba.task.distribution.multicast.address`

You can use this property to specify the multicast address used by the CMS and the workers to join in a cluster group for communication.

Default value: 224.2.2.3

■ `marimba.task.distribution.multicast.port`

You can use this property to specify the multicast port number used by the workers to join a cluster group.

Default value: 54327

- `marimba.task.distribution.tcpip.enabled`

By default, the Hazelcast framework uses the multicast for discovery. You can also configure it to use only TCP/IP for environments where multicast is not available or preferred.

Default value: false

- `marimba.task.distribution.tcpip.members`

You can use this property to specify the list of other hostnames where Task Executor Service and CMS is running. You must specify the list in comma separated value format.

---

Note: In TCP/IP mode of the distributed WoW task, to accept a join request from another worker, you must have at least one running worker in a cluster. This is a pre-requisite for Hazelcast framework.

---

For distributed WoW task, the following WoW properties are not changed:

- `marimba.wow.max.threads`
- `marimba.wow.filter.subnets`
- `marimba.wow.wakeup.strategy`
- `marimba.wow.ping.retry`
- `marimba.wow.ping.timeout`
- `tuner.lms.status`

When you use the distributed WoW task, ensure that you specify the `marimba.wow.filter.subnets` property in CMS tuner, and specify the remaining properties in the Worker tuner.

## How does the WoW feature work?

WoW deployments search for seed tuner on the subnet. A seed tuner is the first available tuner on the subnet which is used to distribute a magic packet to wake up the WoW targets.

To start a WoW deployment, the WoW feature must identify the list of machine IDs or targetDN of the endpoints which have to be woken up. For a WoW deployment task, Policy Manager uses the targetDN data, while Report Center and Infrastructure Administrator use the machine ID data.

For each WoW deployment task, the WoW feature generates a unique Job Id which is a combination of the collection name and the current time in milliseconds. The WoW feature identifies the list of machine IDs for the endpoints that have to be woken up. After the endpoints are identified, the WoW feature performs the following tasks:

- Identify the subnet of the identified endpoints
- Populate the subnet.
- Within the subnet, identify the endpoints which will act as a seed tuner.

The subnet contains the details of the computers to be woken up and the computer which acts as the seed tuner. While identifying endpoints for the subnet, the WoW feature also identifies the LMS, seed machine, vPro, and cache machines.

When the endpoints to be woken up are located in different subnets, the WoW feature parallelly executes WoW deployments for multiple subnets. Once the subnets are identified and loaded with the required data, the WoW feature registers the details of the subnets and starts the process of waking up the required endpoints. Prior to waking up the endpoints, the WoW feature identifies the seed tuner in each subnet. To identify the seed tuner in each subnet, the WoW feature executes the required number of threads in parallel.

The subsequent WoW task will be on queue until the current wow task is completed to utilize the available threads.

When you use Report Center to wake up a machine, the WoW feature immediately performs the WoW deployment without queuing. However, the performance of the WoW feature degrades when you try to wake-up more computers using the Report Center.

---

Note: The WoW feature does not rely on the LDAP Sync schedule to find the machine IDs. The WoW feature performs a direct LDAP lookup operation.

---

## Execution of threads in parallel

When the WoW feature performs a WoW deployment tasks for multiple subnets, the WoW feature initially identifies the seed tuner in each subnet. To identify the seed tuners in the subnets, the WoW feature executes the threads in parallel.

The WoW feature uses two parallel processes to perform the task of finding seed tuners. The WoW feature uses one process to find the subnets and another process to find the seed tuner in each subnet. In both the processes, the WoW feature processes multiple threads in parallel.

For each subnet, the WoW feature searches for a LMS computer or a vPro-enabled computer. If any subnet contains a LMS or a vPro-enabled computer, then that computer acts as a seed tuner. When the WoW feature finds a seed tuner in a subnet, then the WoW feature does not search for any more seed tuners in that subnet.

### Configuring the maximum number of threads

The WoW feature parallelly performs WoW deployments in multiple subnets. After the subnet is loaded with the required details, the WoW feature registers the subnet. The WoW feature processes the registered subnets and wakes up the computers that are marked for wake-up. To identify a seed tuner in each subnet, the WoW feature executes the threads in parallel. The threads identify the seed tuner in each subnet.

When you schedule a wake-up job, the WoW feature segregates the subnets and the computers in each subnet.

You can use the `marimba.wow.thread.max` tuner property to configure the maximum number of threads that the WoW feature can use for parallel processing. For example, `marimba.wow.max.threads=50`. You must restart the tuner after you set this property.

The WoW feature allocates 5 threads to process each subnet. For example, if you set the `marimba.wow.max.threads` tuner property to 50, depending on the number of free threads available, the WoW feature executes the subnets in parallel.

For example, if the WoW feature has to process a WoW task scheduled for 50 subnets, and the maximum number of threads configured is 50, then the WoW feature executes 10 subnets in parallel for this wow task. The WoW feature queues the remaining 40 subnets. If any of the subnet completes its WoW task, then the WoW feature processes the subnet in the queued state.

## Filtering machines and subnets from WoW deployments

When you schedule a WoW deployment task, the WoW feature segregates the subnets, and then segregates the computers in each subnet. The WoW feature provides you with the ability to exclude individual computers or subnets during WoW deployments. You can use the `marimba.wow.filter.subnets` property in the tuner's `prefs.txt` file, to exclude specific computers or subnets during WoW deployment tasks.

For example, you can set the following property:

```
marimba.wow.filter.subnets=10.10.51.52,10.10.52.52,192.168.1.40-192.168.1.54,192.168.2.0/255.255.255.0.
```

You must restart the tuner after you set this property.

## Configuring the seed finding technique

For each WoW task, the WoW feature searches and validates the seed tuners based on the seed tuner finding strategy configured in the `marimba.wow.wakeup.strategy` property. Based on this property, the WoW feature implements the strategy to find the seed tuners. By default, the seed finding strategy is set to LMS, vPro, SeedFinder. In the default strategy, the WoW feature searches and validates the seed tuners in the following order:

- LMS machines
- vPro machines
- Seed Finder machines

You can use the `tuner.lms.status` property to specify that the machine is configured as an LMS. Inventory Service captures the value of this property. The valid values are `True` or `False`.

For a ping operation, you can specify the `marimba.wow.ping.timeout` property to specify the timeout in seconds. To retry the ping operation, you can set the `marimba.wow.ping.retry` property to `true`. You must restart the tuner after you set this property.

## Best practices

You can implement the following best practices:

- Configure at least one LMS machine per subnet to improve the performance of the WoW deployment task.
- The WoW feature supports the following strategies to find seed tuners:

- LMS
- vPro
- Seed Finder
- Cache

It is recommended to configure the seed finding strategies in the following order:

LMS, cache, vPro, SeedFinder

---

Note: The LMS computer is a predefined seed computer, and by default has the highest priority. The cache contains the list of machines which were used as the seed computers during the previous WoW deployment, and therefore the cache strategy is placed prior to vPro because there is a possibility that the vPro-enabled computer is in power off state. To use the vPro-enabled computer which is in a power off state, the WoW deployment task has to wait till the vPro-enabled computer is powered on. The vPro-enabled computer can be woken up even from a powered off state and used as a seed tuner. The SeedFinder strategy has the least priority because this strategy causes multiple threads to run in parallel to find the seed tuners.

---

- Ensure only a very minimal number of WoW deployment tasks using the Infrastructure Administrator or Report Center. Using the Report Center or Infrastructure Administrator for WoW deployment tasks degrades the performance of the WoW feature.
- Ensure that the interval for Update schedule of a WoW deployment task has enough time to avoid overlap of other WoW deployment tasks.
- If the machines to be woken up are located in different subnets, then CMS takes more time to wake-up these machines, because prior to waking up the computers in a subnet, for each subnet, CMS has to find a powered on computer which will act as a seed tuner for the subnet. For example, CMS takes less time to wake up 100 computers in a subnet than to wake up 10 computers located in 10 subnets.

## Limitations of WoW deployments

WoW deployments have the following limitations:

- If you configure the vPro strategy for a WoW deployment, the WoW feature tries to wake up the vPro machine and use that machine as a seed tuner. But during the vPro-enabled computer wake-up process, if the computer takes a long time to initialize the network, then the WoW feature excludes the VPro-enabled computer from being used as a seed computer.
- The Tuner Administrator page does not display the individual computer's power on status, but displays the status as success or failure based on the status of the magic packet. The Tuner Administrator page displays the status as success if the magic packet is successfully received by the target machine.
- When the WoW wakeup task fails to wake up a target, the Administrator cannot perform a retry operation on the target.

**Workaround:** You can create a new WoW deployment task for the computers which could not be woken up.

- The WoW feature does not wake-up computers which do not have a hostname.
- When the Administrator performs a wake-up operation on a single target, the WoW feature does not display the status of the magic packet, but displays the Wakeup job triggered on target machine message as the status.

**Workaround:** To perform a WoW deployment task on a single computer, you can use the machine details page or wakeup link in Report Center.

## Troubleshooting WoW deployments

You can use the following troubleshooting techniques for WoW deployments:

- To view the log entries for a specific WoW deployment task, filter the CMS logs based on the task ID. You can view the logs for a specific task. For example, 1328518620087\_collection\_win7\_Asia\_Katmandu, which has the format of a combination of current time and collection name.
- If the CMS logs report the No active database configured in CMS message, then ensure that the database is active and the CMS computer is accessible to the computer which contains the database.
- If the CMS logs report the Total Number of machines found: 0 or Total number of subnets loaded: 0 message, then it means that there is no data in the database for the specified machine ID.

- If the number of computers loaded is less than the actual number of computers woken up, then check if the WOL status is disabled for the missing computers in the `machine_wow_details` view in `invdb`. For SQL Server, the default database name is `invdb`, unless you edited the necessary database setup batch scripts to change this value during the install, reinstall, or upgrade process.
- From BBCA 8202 version, the WOW module uses Intel AMT WS-MAN protocol to wake remote Intel vPro machines. When WS-MAN does not work for older version of AMT vPro machines, vPro wake up can fall back to SOAP API method by using following tuner property “`marimba.wow.vpro.usesoapapi=true`”.

Note: This property change needs to be made at CMS tuner and it affects all WOW deployments that have vPro machines.

- On some Intel AMT vPro machines, the OS boot up may take long time hence tuner on this machine cannot be used as seed tuner. In order to make the WOW module wait for vPro machine to become available for use, a tuner property “`marimba.wow.vpro.waittimeout`” can be used to extend the wait timeout.

Note: The value must be integer and in seconds. The default value is 60.

- Windows 8 introduced a feature called Fast Startup ( hybrid boot or hybrid Shutdown) to help your machine start up faster after shutting down. Due to this new feature, WoW is not supported for Windows 8 machine from shut down. Please refer kb article <http://support.microsoft.com/kb/2776718>

**Workaround:** User need to clear the option “Turn On Fast Startup” in Power Option Settings to wakeup windows 8 endpoint. However, this is not recommended by Microsoft.

- While provisioning vPro machine through Policy Service with the option ‘Use the same IP as the host (for static IP only)’ on Static Environment, the Gateway address field will be empty on AMT network settings of vPro machine and it returns the vPro machine power state as -1. This causes the issue, when user tries to wake up the machine using vPro AMT and wakeup will be failed.

**Workaround:** User needs to set gateway address in AMT network settings manually on each time policy update with vPro provision settings.

## WoW-specific tables and views

The 8.2.01 schema contains the following WoW-specific tables:

- machine\_wow\_state

This table stores the machine name, LMS Status, wol status and the subnet range for the machine.

- seed\_tuner\_machine\_cache

This table stores the cache data for each subnet.

The 8.2.01 schema contains the following WoW-specific view:

- machine\_wow\_details

This view retrieves the data from various tables which stores the entire data for WoW execution. This view gathers data related to vPro Status, vPro AMT username and password, LMS status, and subnet range.

## Creating a WoW deployment in the Subscription Manager

You can create a WoW deployment in the Subscription Manager. You can create the WoW enabled deployment package for a policy in the following two ways:

- Immediate deployment

In the Edit Policy page, if you select the **WoW Deployment** option for a package, CMS creates the WoW task for the selected target and schedules the start of the execution of the WoW task after four minutes from the CMS time. CMS executes the WoW task irrespective of endpoint time zones. This is known as immediate deployment.

- Deployment based on the endpoint's timezone

You can create a WoW task based on the endpoint's timezone. You can specify the schedule of a WoW deployment for a package in the following locations:

- Edit Primary Schedule page
- Edit Secondary Schedule page
- Edit Repair Schedule page
- Edit Update Schedule page

Once you enable the WoW deployment for a package in any of the preceding pages, the WoW feature creates an individual WoW task for each schedule with the appropriate timezone information.

The WoW feature creates a WoW task for each time zone. For example, if you schedule an WoW deployment on a primary schedule that is scheduled to start at 1:00 PM for the collection\_win7 collection which has machines located in 4 different time zones, then the WoW feature creates 4 different WoW tasks with the following names:

- collection\_win7\_Asia\_Katmandu
- collection\_win7\_Asia\_Calcutta
- collection\_win7\_America\_New\_York
- collection\_win7\_America\_Los\_Angeles

The WoW feature executes these tasks at 1:00 PM based on the respective time zones.

If you create a WoW deployment for immediate deployment and a WoW deployment based on the endpoint's timezone, then the WoW feature creates an individual WoW task for each type of deployment and each timezone.

---

**Important:** To enable the WoW feature, you must select the Enable WoW feature option in the Advanced Options section of the Policy Manager Configuration page. If you do not select the **Enable WoW feature** option, but enable the WoW deployment for a policy, then the WoW task will be created but will not be executed.

---

## ► **To create a WoW deployment**

- 1 Choose a target in Policy Manager and add the required channels.
- 2 Select the **WoW Deployment** option.
- 3 Set the primary and secondary schedules for the policy.
- 4 Save the policy.

If the endpoint is powered off, the WoW deployment wakes up the computer based on the set schedule.

---

Note: Before disabling the overall WoW option in the Advanced options section of the Policy Manager Configuration page, if you create a policy with a WoW-enabled package , then the WoW check box for that package is checked but disabled in the user interface. CMS does not execute the WoW task.

---

## Cancelling a WoW task

You can use Policy Manager to cancel the current running WoW tasks that are created for package deployments.

### ► To cancel a WoW task:

- 1 Once the task is scheduled, type the following link in your Web browser:

```
http://CMSPHOST:PORT/shell/settings/  
list_tasks.do?action=list&group=policy
```

CMS displays the list of all the WoW tasks.

Replace CMSPHOST:PORT with your CMS host address and port number.

- 2 Select a WoW task that you want to cancel.
- 3 Click the Stop button to stop the execution of the WoW task.

---

Note: You cannot cancel a WoW task scheduled from Infrastructure Administrator or the Report Center.

---

## Overview of schedules

When you create or edit a policy using Policy Manager, you have the option of specifying various schedules. The schedules define when packages are distributed, updated, verified, and repaired on targets.

---

Note: Scheduling package updates can impact the state of the software on your endpoints, and requires attention to best practices. See “Managing software, data, and updates” on page 260.

---

This section includes the following topics:

- “What schedules can you set for packages?” on page 232
- “What are primary and secondary schedules?” on page 232
- “What are activation and expiration?” on page 233
- “What is an update schedule?” on page 234
- “What is a repair schedule?” on page 235
- “What is recurrence?” on page 236

## What schedules can you set for packages?

You can set the following schedules for packages in a policy:

- Primary schedule
- Secondary schedule
- Update schedule
- Repair schedule

The four types of schedules are independent from each other. For each schedule, you can specify activation and expiration dates and times that define the window of time when you want to make packages available for certain activities or states. Schedules are relative to the endpoint’s time zone.

If you do not want to specify activation dates and times, you can also specify that activities take place the next time Policy Service updates. If you choose this option, activities take place immediately the next time that Policy Service updates. For more information about Policy Service updates, see “Specifying the Policy Service schedule for a target” on page 270.

You can also set a blackout period for dates and times when you don’t want any activities to take place. See “Setting a blackout period for a target” on page 195. You can also exempt specific packages from the blackout period. See “Exempting packages from the blackout period” on page 199.

---

Note: Scheduled activities do not take place if the tuner property for the blackout period (`marimba.schedule.filter`) is set to never at the endpoints.

---

## What are primary and secondary schedules?

The *primary* and *secondary schedules* define when you want to apply a package's primary and secondary states to the targets in a policy. The schedules include activation and expiration dates and times for applying the package's primary and secondary states to the targets. You must specify a secondary state and schedule only if the primary state is *stage* or *advertise*. For a list of the primary and secondary states, see “Overview of installation states” on page 215.

For example, you can set primary and secondary schedules so that a package is downloaded on the targets during a specified period of time, and then later installed during another specified period of time. First, you can set the primary state of a package to *stage*, so that it is downloaded (but not yet installed) on the targets starting January 1 at 5 PM (activation) and ending January 14 at 5 PM (expiration). Then, you can set the secondary state to *install*, so that the package is installed on the targets starting January 14 at 5 PM (activation) and ending January 28 at 5 PM (expiration).

If the policy includes many packages, you can only set a secondary schedule if a secondary state is specified for one or more packages. The secondary schedule is assigned to packages with secondary states only.

For instructions on setting the primary and secondary schedules, see “Setting the primary and secondary schedule for packages” on page 237.

## What are activation and expiration?

Activation and expiration define the window of time when you want to make packages available for certain activities or states.

*Activation* defines the date and time after which installation states, updates, or repair activities for a package can be applied to the targets in a policy. For example, if you want to distribute a certain package to targets after January 31 at 5 PM, you can specify that as the activation date and time in the policy. You can also specify that activation takes place when Policy Service updates. If you choose this option, activities take place immediately the next time that Policy Service updates. For more information about Policy Service updates, see “Specifying the Policy Service schedule for a target” on page 270.

*Expiration* defines the date and time after which installation states, updates, or repair activities for a package can no longer be applied to the targets in a policy. This means that if any of these activities have not taken place and the expiration date and time passes, the activities no longer take place. For example, if you do not want a package to be available for downloads after February 28 at 5 PM, you can specify that as the expiration date and time in the policy. Endpoints can no longer download that package after that date and time. You can also specify that there be no expiration for particular packages.

---

Note: Expiration does not remove packages from the endpoints on which they are already installed. It only affects endpoints that have not yet performed a certain activity, such as installation, by the expiration date and time.

---

You can also specify activation and expiration for activities on a recurring schedule, such as updates and repair activities. In these cases, activation defines the date and time after which you want updates or repair activities to start taking place regularly, while expiration defines the date and time after which you no longer want them to take place.

## What is an update schedule?

The update schedule defines when a package gets updated and, if necessary, has the updates installed automatically. The update schedule is useful if applications and content that you package and distribute to users change frequently. You can use the update schedule to regularly check for and download new or changed files. The update schedule only takes effect on the endpoint if the package has already been subscribed or installed.

---

Note: Scheduling package updates can impact the state of the software on your endpoints, and requires attention to best practices. See “Managing software, data, and updates” on page 260.

---

Policy Manager allows you to override the update schedule that was set during packaging and publishing. You can specify a recurring update schedule that occurs daily, weekly, or monthly, at specific days and times. You can also specify the update schedule to *never*. Optionally, you can set activation and expiration dates and times so that you can control when updates start and stop taking place.

For instructions on setting the update schedule, see “Setting the update schedule for packages” on page 238.

---

Note: The package update schedule appears as never when viewed using Tuner Administrator, even when a different schedule has been set using Policy Manager. Policy Service sets the package update schedule to never so that the package does not update on its own and updates according to the schedule set through Policy Manager.

---

## What is a repair schedule?

The repair schedule defines when a package is verified and, if necessary, repaired automatically. This is useful if applications and content that you package and distribute to users become unusable or do not run properly. You can use the repair capabilities to automatically check and fix problems, such as missing or corrupted files. The repair schedule only takes effect on the endpoint if the package has already been subscribed or installed. Also, repair operations only work for packages created using Application Packager.

Policy Manager allows you to override the repair schedule that was set during packaging and publishing. You can specify a recurring repair schedule that occurs daily, weekly, or monthly, at specific days and times. You can also specify the repair schedule to *never*. Optionally, you can set activation and expiration dates and times so that you can control when repair activities start and stop taking place.

For instructions on setting the repair schedule, see “Setting the repair schedule for packages” on page 241.

**What does repair do?** When you repair a package, the file and other objects in the package are first compared with the file information for the package when it was originally installed. After verification is complete, one of the following occurs:

- There are no object mismatches. No repairs are needed.

- There are object mismatches. The target tuner re-installs the affected files or objects, if they are available from the tuner storage. If the affected files are not available, the target tuner contacts the transmitter to download the files or objects needed to complete the repair.

---

Note: If the transmitter cannot be contacted or the package on the transmitter has been deleted or modified so that the needed objects are no longer available, the log files include a record of the objects that cannot be repaired.

---

## What is recurrence?

Recurrence refers to activities that occur repeatedly, usually according to a regular schedule. In Policy Manager, you can specify recurring schedules for updates and repair activities. Such activities take place repeatedly, according to the schedule between the activation and expiration dates and times.

For example, you can schedule a package to update daily, but no update requests are honored before the activation time. After the activation time and before the expiration time, Policy Manager honors update requests. After the expiration time, the application's update schedule is set to never, and no further update requests are honored.

## Auto Inventory Scan

When you use BBCA to make any settings or software changes to the endpoint, the changes in the endpoint are captured only when the next inventory scan runs according to a schedule. However, if you want to immediately capture the changes made at the endpoint, you can use the auto inventory scan feature to immediately capture the inventory changes at the endpoint. The advantage of this feature is that the software and policy settings changes are immediately captured and you need not use regular policy schedules which consume resources. The feature invokes only Application Scan and Marimba Scan.

To enable this feature, set the `marimba.subscription.autoscan.enabled` property to true.

**Note:**

- The auto inventory scan feature immediately captures the settings or software changes in the endpoint only when you use BBCA to deploy the changes.
- When you use remote package installation through Deployment Manager or through Remote Tuner admin, Inventory scan does not immediately run to capture the policy or inventory changes at the endpoint.
- If you use Policy Manager to distribute the Power Setting or Power Scheme policy, the changes are captured only in regular Hardware scan.

When you use BBCA to add or remove a package in an endpoint, the auto inventory scan starts the Application scan only. However, when you use BBCA to make changes to a Tuner or a Channel, the auto inventory scan starts the Marimba scan only. When you use BBCA to make changes to both packages, and Tuner or Channel properties, then the auto inventory scan starts both the marimba scan and Application scan.

## Setting the primary and secondary schedule for packages

You can set the schedule for the primary and secondary states of packages. When an installation is scheduled, the target tuner downloads the package from the transmitter and performs the installation activity that you have specified. If you specify both a primary and a secondary state, the target tuner performs the installation activities at the scheduled time. Schedules are relative to the endpoint's time zone.

This procedure assumes you are on the Packages tab of the Edit Policy page. See “General directions for creating and editing policies” on page 209.

### ► To set the schedule for the primary and secondary state of a package

- 1 On the Packages tab of the Edit Policy page, locate the row of the package for which you want to set schedules.
- 2 In the package row, click the arrow in the first column on the left and click Set.
- 3 To set a schedule for the primary state:
  - a In the Schedule for Primary State section, click Edit.  
The Edit Primary Schedule page appears.
  - b In the Activation and Expiration areas, specify the dates and times to define the window of time when you want to make packages available for the primary state.

See “What are activation and expiration?” on page 233.

- c Click OK to save the schedule and return to the Edit Policy page.

Note: The schedule now appears in the Schedule for Primary State section.

- 4 If you’ve specified a secondary state for the package, set a schedule for the secondary state:

- a In the Schedule for Secondary State section, click Edit.

The Edit Secondary Schedule page appears.

- b In the Activation and Expiration areas, specify the dates and times to define the window of time when you want to make packages available for the secondary state.

See “What are activation and expiration?” on page 233.

- c Click OK to save the schedule and return to the Edit Policy page.

Notice that the schedule now appears in the Schedule for Secondary State section.

When you are finished editing the policy, preview and save the policy. See “Previewing and saving policy changes” on page 211.

---

Note: You cannot set a primary schedule and a secondary schedule where the secondary schedule is earlier than the primary schedule. For example, the following schedules are *not* valid:

---

Primary schedule: Activate at 2/1/04 1:30 AM

Secondary schedule: Activate at 1/1/04 12:00 AM

## Setting the update schedule for packages

You can set the update schedule for packages. When an update is scheduled, the target tuner checks if the package has been updated on the transmitter from which the package gets updates. If the package has already been updated, it downloads any new or modified files and installs them on the target. The update schedule only takes effect on the endpoint if the package has already been subscribed or installed.

---

Note: The package update schedule appears as *never* when viewed using Tuner Administrator, even when a different schedule has been set using Policy Manager. Policy Service sets the package update schedule to *never* so that the package does not update on its own and updates according to the schedule set using Policy Manager.

---

Packages might already have an update schedule associated with them, so you have the option of using that update schedule or overriding it, using Policy Manager:

- Different users might be doing the software packaging and the software distribution. If you are responsible for using Policy Manager to distribute software, you might want to override the schedules set by those doing software packaging.
- Also, you might choose to set a default schedule during packaging and then override the default schedule when you distribute software to specific groups.
- After you have set the schedule during packaging, you cannot easily change it. It is easier to change schedules using Policy Manager.

This procedure assumes you are on the Packages tab of the Edit Policy page. See “General directions for creating and editing policies” on page 209.

#### ► To set the update schedule for a package

- 1 On the Packages tab of the Edit Policy page, locate the row of the package for which you want to set schedules.
- 2 In the row of the package, click the arrow in the first column on the left side of the page to display the schedule information for the package.
- 3 In the Update Schedule section, click **Edit**.

The Edit Update Schedule page appears.

- 4 Use the drop-down list to specify the schedule that you want the package to follow:

- **Follow the schedule specified when the package was created and published.** Choose this option if you specified a schedule for the package when it was created and published and you want to follow that schedule.
- **Use Policy Manager to set the update schedule.** Choose this option if you want to use Policy Manager to manage the update schedule.

- 5 If you chose to use Policy Manager to manage the update schedule, specify the following information:
  - a In the Recurrence area, specify how often you want to update the package:
    - **Never**—The package is never automatically updated. You'll have to manually update the package, when necessary.
    - **Daily**—The package can be updated every day, only on weekdays (Monday through Friday), or every 1 to 365 days, depending on the number you set.
    - **Weekly**—The package can be updated every 1 to 52 weeks, depending on the number you set. During the week when an update is scheduled, an update occurs only on the days you select. You must select at least one day of the week.
    - **Monthly**—The package can be updated every 1 to 12 months on the day you set. If you choose day 30 or 31 and the month when an update occurs doesn't have that day, such as February, the update occurs on the first day of the next month.
  - b In the Update Time area, specify the time of the day when you want to update the package:
    - **Update at**—Select to choose one time of the day when updates occur.
    - **Update every**—Select to specify multiple times of the day when updates occur. You can set updates to happen every specified number of minutes or hours during the day and you can limit updates to a single range of hours during the day. For example, you can have updates occur every hour between 9 AM and 5 PM, or every 30 minutes from 5 PM to 9 AM.
  - c In the Activation and Expiration areas, specify the dates and times when you want to start and stop following this update schedule.

See “What are activation and expiration?” on page 233.
- 6 Click **Enable WoW Deployment** to “wake up” any target machines that is powered down during the update schedule.

For more information on the WoW feature, see “Creating WoW deployments” on page 219.
- 7 Click **OK** to save the schedule and return to the Edit Policy page.

---

Note: The schedule now appears in the Update Schedule section.

---

When you are finished editing the policy, preview and save the policy. See “Previewing and saving policy changes” on page 211.

## Setting the repair schedule for packages

Sometimes applications and content that you package and distribute to users might become unusable or not run properly. You can use the repair capabilities to automatically check and fix problems, such as missing or corrupted files.

You can set the repair schedule for packages. When repair is scheduled, files that have changed since the package was installed are re-installed. Before repairing the package, the target tuner connects to the transmitter from which the package gets updates.

The repair schedule only takes effect on the endpoint if the package has already been subscribed or installed. Also, repair operations only work for packages created using Application Packager. Packages might already have a repair schedule associated with them, so you have the option of using that repair schedule or overriding it using Policy Manager. Here are some situations where you might want to use Policy Manager to set a repair schedule:

- Different users might be doing the software packaging and the software distribution. If you are responsible for using Policy Manager to distribute software, you might want to override the schedules set by those doing software packaging.
- Also, you might choose to set a default schedule during packaging and then override the default schedule when you distribute software to specific groups.
- After you set the schedule during packaging, you cannot easily change it. It is easier to change schedules using Policy Manager.

This procedure assumes you are on the Packages tab of the Edit Policy page. See “General directions for creating and editing policies” on page 209.

## ► To set the repair schedule for a package

1 On the Packages tab of the Edit Policy page, locate the row of the package for which you want to set schedules.

2 In the row of the package, click the arrow in the first column on the left side of the page to display the schedule information for the package.

3 In the Repair Schedule section, click Edit.

The Edit Repair Schedule page appears.

4 Use the drop-down list to specify the schedule that you want the package to follow:

- **Follow the schedule specified when the package was created and published.** Choose this option if you specified a schedule for the package when it was created and published and you want to follow that schedule.

- **Use Policy Manager to set the repair schedule.** Choose this option if you want to use Policy Manager to manage the repair schedule.

5 If you chose to use Policy Manager to manage the repair schedule, specify the following information:

a In the Recurrence area, specify how often you want to repair the package:

- **Never**—The package is never automatically verified and repaired. You'll have to manually repair the package when necessary.

- **Daily**—The package can be verified and repaired every day, only on weekdays (Monday through Friday), or every 1 to 365 days, depending on the number you set.

- **Weekly**—The package can be verified and repaired every 1 to 52 weeks, depending on the number you set. During the week when a repair is scheduled, it occurs only on the days you select. You must select at least one day of the week.

- **Monthly**—The package can be verified and repaired every 1 to 12 months on the day you set. If you choose day 30 or 31 and the month when a repair occurs doesn't have that day (such as February), it occurs on the first day of the next month.

b In the Repair Time area, specify the time of the day when you want to repair the package:

- **Repair at**—Select to choose one time of the day when repair occurs.

- **Repair every**—Select to specify multiple times of the day when repair occurs. You can set it to happen every specified number of minutes or hours during the day, and you can limit it to a single range of hours during the day. For example, you can have it occur every hour between 9 AM and 5 PM, or every 30 minutes from 5 PM to 9 AM.
  - c In the Activation and Expiration areas, specify the dates and times when you want to start and stop following this repair schedule.  
See “What are activation and expiration?” on page 233.
- 6 Click OK to save the schedule and return to the Edit Policy page.
- Notice that the schedule now appears in the Repair Schedule section.
- When you are finished editing the policy, preview and save the policy. See “Previewing and saving policy changes” on page 211.

## Setting common schedules for multiple packages

You can set common schedules for multiple packages so that two or more packages in a policy have the same schedules. When you set common schedules for multiple packages, you set all schedules: primary, secondary, update, and repair. Even if you only edit one type of schedule, the other schedules that appear on the Set Common Schedule page are applied to all the packages.

This procedure assumes you are on the Packages tab of the Edit Policy page. See “General directions for creating and editing policies” on page 209.

### ► To set common schedules for multiple packages

- 1 On the Packages tab of the Edit Policy page, select the check boxes that correspond to the packages for which you want to set common schedules.
- 2 Click Set Common Schedule.
- 3 On the Set Common Schedule page, set the primary, secondary, update, and repair schedules for the packages.

For example, to set the schedule for the primary state:

- a In the Schedule for Primary State section, click Edit.

The Edit Primary Schedule page appears.

- b In the Activation and Expiration areas, specify the dates and times to define the window of time when you want to make packages available for the primary state.
- c Click OK to save the schedule and return to the Edit Policy page.

Notice that the schedule now appears in the Schedule for Primary State section.

See the following sections:

- “Setting the primary and secondary schedule for packages” on page 237
  - “Setting the update schedule for packages” on page 238
  - “Setting the repair schedule for packages” on page 241
- 4 When you have finished setting schedules, review the schedules that appear on the Set Common Schedule page.

**Note:** The schedules that appear on the Set Common Schedule page are applied to all packages that you selected. This behavior applies even if you have not edited or changed one type of schedule (such as if you changed the primary schedule, but not the update schedule).

- 5 After reviewing the schedules, click OK to save the schedule and return to the Edit Policy page.

When you are finished editing the policy, preview and save the policy. See “Previewing and saving policy changes” on page 211.

## Conflict resolution: states and schedules in policies

The same package can be assigned to an endpoint with conflicting states and schedules. For example, an endpoint belongs to two different groups and gets the same package with different states and schedules. To resolve the conflict, Policy Service attaches a number to the conflicting states, as shown in Table 13-2 on page 245 and Table 13-3 on page 245. These numbers represent the priority for the final state and schedule state of the package. The final state of the package is defined as the secondary state, if it exists for the packager; otherwise, the final state is the primary state. For more information about how states and schedules can affect updates, see “Managing software, data, and updates” on page 260. For more information about primary and secondary states, see “Overview of installation states” on page 215.

The schedule state for a package is defined as one of 12 valid schedule configurations represented by the presence or absence of the primary, secondary, update, and repair schedules, as shown in Table 13-3 on page 245. The final state has precedence over the schedule in all cases, which means that the schedule state is only relevant if the final state is not the same between any two conflicts. To account for this, the final state conflict priority numbers are incremented by 100, as shown in Table 13-2 on page 245.

Policy Service compares the conflict priority sums (by adding the final state and schedule state conflict priority numbers) for all instances of the same package. The instance with the lowest sum wins. If more than one instance has the same sum, the winner is chosen at random.

Table 13-2: Final state conflict priority

State	Conflict priority
Exclude	100
Uninstall	200
Primary	300
Install-Start-Persist	400
Install-Start	500
Install-Persist	600
Install	700
Stage	800
Advertise	900

Table 13-3: Schedule state conflict priority

Primary	Secondary	Update	Repair	Conflict priority
Yes	Yes	Yes	Yes	1
Yes	Yes	No	Yes	2
Yes	Yes	No	No	4
Yes	Yes	Yes	No	3
Yes	No	Yes	Yes	5
Yes	No	No	Yes	6
Yes	No	Yes	No	7

Table 13-3: Schedule state conflict priority (Continued)

Primary	Secondary	Update	Repair	Conflict priority
Yes	No	No	No	8
No	Yes	Yes	Yes	These schedule configurations are not valid because you cannot have a secondary state without a primary state.
No	Yes	Yes	No	
No	Yes	No	Yes	
No	Yes	No	No	
No	No	Yes	Yes	9
No	No	No	Yes	10
No	No	Yes	No	11
No	No	No	No	12

## Resolving differences in package states or schedules

In cases where the states or schedules specified for packages are different for the targets in a policy, you must specify whether you want to maintain the difference or use one setting for all targets.

If there are any differences that need to be resolved, you see the following message on the Edit Policy page:

Packages and schedules marked with  have different states and schedules across the selected targets. You can maintain these differences or select a state or schedule to apply across all selected targets.

### ► To resolve differences for a package states and schedules

- From the Edit Policy page, locate the packages that have different states or schedules for the selected targets.
- In the Packages tab, under the Primary State or Secondary State column, select one of these options:
  - Maintain difference for targets**—Select this option if you want to maintain different package states for the targets, such as if you want the state to be *stage* for one target and *install* for another.
  - <State>**—Select this option if you want to specify one package state that applies to all the targets.

- 3 Under the Schedule section, for schedules that are different for the selected targets, select one of these options:
  - **Maintain <primary, secondary, update, repair> schedule difference for targets**—Select this option if you want to maintain different package schedules for the targets, such as if you want a package to be available for installation at different dates for each target.
  - **Apply the edited <primary, secondary, update, repair> schedule to all selected targets**—Select this option if you want to specify one schedule for the package that applies to all targets. You must specify the schedule that you want to apply. See “Overview of schedules” on page 231.

When you are finished editing the policy, preview and save the policy. See “Previewing and saving policy changes” on page 211.

## Scheduling wake-up start and interval times using the PC Alarm Clock feature

Version 8.2 of the Policy Manager includes a PC Alarm Clock feature that you can use to specify wake-up start and interval times for computers enabled with Intel AMT or vPro. Use the PC Alarm Clock feature to ensure timely delivery of software and patches.

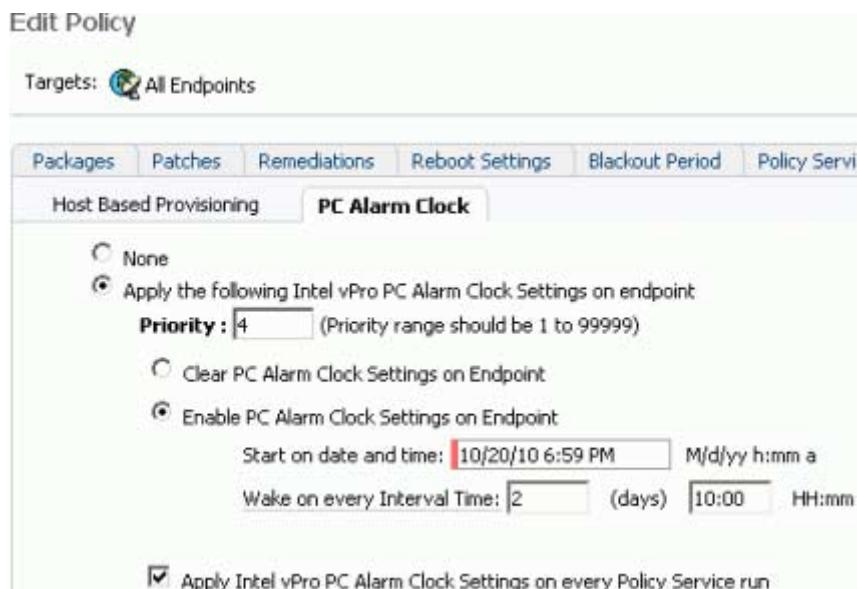
When the computer starts, the service channels are started. Using the PC Alarm Clock feature, you can ensure that the latest updates reach the endpoint.

### ► To schedule wake-up start and interval times

- 1 On the Edit Policy page in Policy Manager, click the PC Alarm Clock tab.
- 2 Select **Apply the following Intel vPro PC Alarm Clock Settings on endpoint**, as shown in Figure 13-1.
- 3 Select **Enable PC Alarm Clock Setting on Endpoint**, and specify the following values:
  - a **Start on date and time:** Enter a date and time for the host to start, using the specified date and time format.
  - b **Wake on every Interval Time:** Enter the time period between wake-up times after the start time. The ranges of values are:
    - Days: 0 to 65535

- Hours: 0 to 23
  - Minutes: 0 to 59
- 4 To prevent random assignment of vPro settings from more than two different group policies, specify a value for **Priority**.
  - 5 To force apply the PC Alarm Clock settings to the endpoint, select the **Apply Intel vPro PC Alarm Clock Settings on every Policy Service run** check box.
- After updating the Policy Service on the endpoint, the settings are applied to the vPro computers.

Figure 13-1: Specifying PC Alarm Clock settings



## Setting the install priority for packages in a policy

This section includes the following topics:

- “What is install priority?” on page 249
- “Install priority concepts” on page 249
- “Conflict resolution: Packages with the same install priority” on page 252

- “Conflict resolution: when multiple users edit the same policy” on page 252

## What is install priority?

The *install priority* determines which package is installed first when you schedule two or more packages for installation on a target at the same time. For instructions on changing install priority, see “Install priority considerations” on page 250.

The install priority rules are guidelines that do not guarantee a specific installation order. See “Exceptions to install priority” on page 249.

You can specify prerequisite packages that other packages require before installing the latter. But you must observe best practices relating to scheduling updates. See “Managing software, data, and updates” on page 260.

### Install priority concepts

When you assign a package to a target, the default numeric value assigned to the install priority of the package is 99999. You can view this value by looking at the package’s `mrbachannelorder` attribute in the policy object for the target using an LDAP browser/editor tool, as shown in Figure 13-2 on page 249.

When you set install priority (as described in “Setting install priority for packages” on page 251), the numeric value for the `mrbachannelorder` attribute changes corresponding to the install priority you set. The numeric value for the `mrbachannelorder` attribute matches the number that appears in the Install Priority column on the Target Details or Edit Policy page.

Figure 13-2: Example values for the `mrbachannelorder` attribute

	http://maramba1:5282/472test=3
<code>mrbachannelorder</code>	http://maramba1:5282/Subnet_Repeater_Policy=2
	http://maramba1:5282/BrowserIntegrationModule=1

The package with the lowest value is installed first. In this example, the package with the value 1 is installed first, followed by 2 and 3.

## Exceptions to install priority

This section describes situations where Policy Manager doesn't follow your specified install priority.

**Timeouts and post-install scripts.** Typically, Policy Service waits for packages to complete installation (including any post-install scripts) before installing the next package specified by the install priority. Policy Manager ignores install priority under the following conditions:

- Download and installation of the package takes longer than the specified timeout (see “`marimba.subscription.timeout`” on page 281). The default timeout is 60 minutes. When the timeout is reached, Policy Service begins installing the next package.
- You have configured the package to run the post-install script as a detached process. In this case, Policy Service considers the package installed when the post-install script starts running, and begins installation of the next package.

**Dependencies.** You can use Application Packager to set dependencies for packages, such as available hard disk space, installed RAM, or the presence of other packages prior to installation. If you have set dependencies for a package, the dependencies take precedence over install priority.

## Install priority considerations

When you set install priority for packages, consider the following Target View settings on the Packages tab of the Edit Policy page:

- **Starting priority.** The number you specify as the starting priority determines the package installation sequence in comparison to packages in other policies. The starting priority is important if certain endpoints belong to multiple targets or groups, and you want to control whether packages assigned to one target have a higher priority than those assigned to another target. For example, if you want packages assigned to the All Endpoints target to have a higher priority than those assigned to other targets and groups, you can reserve install priority numbers 1 to 100 for the All Endpoints target. Then you can assign all other targets a starting priority of 101 and higher.
- **Priority.** The order in which packages appear in the Packages tab of the Edit Policy page determines their install priority. The default priority displays as N/A, which is equivalent to a `mrbachannelorder` attribute value of 99999.



- **Update Priority.** Use this button to change the installation order of a package to a new installation order. When the installation order is changed from one position to the other, if there is another package with the same priority, the installation order of that other package and the remaining packages will be rearranged to maintain the sequence.

To change the priority of a package to the default (N/A or 99999), delete the contents of its Priority text box and click Update Priority. You must always click the Update Priority button before clicking Preview to save your changes.

- **Revert Priority.** Use this button to undo the most recent priority change you made. After you save changes, you cannot revert priority.

See “General directions for creating and editing policies” on page 209.

## Setting install priority for packages

### ► To set the install priority for packages

- 1 On the Packages tab of the Edit Policy page, set the starting priority for the packages in this policy:

- a In the Starting priority text box, enter a numeric value indicating this policy’s priority compared with other policies. See “Install priority concepts” on page 249.
  - b Click Apply.

Notice that the first package in the policy gets the number that you specified in the Starting priority text box, such as 101. The following packages get numbers corresponding to their order in the policy (the second package gets 102, the third gets 103, and so on).

- 2 If you want to promote or demote the install priority of a package, locate the row of that package.
- 3 To assign a new install priority, type a different number into the Priority field.

---

Note: As an example, if you assign 5 in the Priority field—and you already have a package with an install priority of 5—Policy Manager increments the existing install priority and those that follow by one, until the application resolves all numbering conflicts.

---

When you are finished editing the policy, preview and save the policy. See “Previewing and saving policy changes” on page 211.

## Conflict resolution: Packages with the same install priority

It is possible to have the same install priority for different packages when viewing the Target Details page. For example, you set the following install priorities for user Alan:

- Package A, install priority 1
- Package B, install priority 2
- Package C, install priority 3

You also set the following install priorities for the user group Engineering:

- Package D, install priority 1
- Package E, install priority 2
- Package F, install priority 3

If Alan is a member of Engineering, the Target Details page shows overlapping install priorities (two 1s, two 2s, and two 3s). In this case, all packages with install priority 1 have priority over those with install priority 2, and so on. You can resolve further install priority collisions by using the Starting Priority value to assign different blocks of priority numbers to packages with overlapping targets. See “Setting install priority for packages” on page 251.

## Conflict resolution: when multiple users edit the same policy

This section shows the result when multiple users edit a policy at the same time.

► **The following table assumes the following sequence of events**

- 1 User 1 goes to the Edit Policy page and begins editing.
- 2 User 2 goes to the Edit Policy page and also begins editing.
- 3 User 2 finishes editing and saves the Edit Policy page.

#### 4 User 1 finishes editing and saves the Edit Policy page.

Table 13-4: When multiple users edit the same policy

User 1:	User 2:	Final result
Edits packages A and B	Adds package C	Policy contains packages A, B, and C.
Edits package A	Adds or edits package A	User 1 wins. User 1's edits overwrite User 2's edits.
Adds or edits package A	Edits package B	User 1 wins. Even though User 1 did not edit package B, the previous states and schedules set for package B overwrite User 2's edit.

For information about conflict resolution when multiple users edit tuner and package properties, see “Conflict resolution: When multiple users edit properties” on page 292.

## Copying policies

If a policy exists for a target and you want other targets to have that policy, you can copy that policy to the other targets. For more information about policies, see “What is a policy?” on page 208. Copying policies is available in single selection mode only. You can copy the policy from one *source* target to one or more *destination* targets.

You can copy the policy directly assigned to the source target only. The Directly Assigned To column shows the target to which specific packages are directly assigned. Any packages or properties that are not part of the directly assigned policy for the source target (that is, packages that are indirectly assigned through inheritance from another target) are not included in the policy copied to the destination targets.

---

**WARNING:** If the destination target already has a directly assigned policy, that policy is overwritten by the policy that you copy from the source target.

---

After copying a policy, the policies assigned to the source and destination targets exist independently from each other. That is, editing the policy for the source target does not affect the policy for the destination target, and vice-versa.

## ► To copy a policy from one target to another

- 1 From the Target View or Target Details page, select the source target from which you want to copy the policy.
- 2 Click Copy (  ).
- 3 From the Copy Policy page that appears, select one or more destination targets to which you want to copy the policy.
- 4 Click Preview.
- 5 From the Copy Preview page, review the policy that you are copying to the destination targets and choose one of the following options:
  - If you want to copy the policy, click Copy to confirm your changes and save them.
  - If you do not want to save your changes and want to change the destination target or targets, click Back to Edit.
  - At any point, if you want to discard your changes and return to the Target View page or Target Details page, click Cancel.

The destination targets are now assigned the policy that you copied.

## Deleting policies

To delete a policy, you must use one of the following pages:

- Target View page
- Target Details page

When you delete the policy using this method, the packages (and other information) included in the policy are deleted from the targets the next time that Policy Service updates. However, packages are not deleted from the targets if any of the following is true:

- The packages' state is set to `install-persist` or `install-start-persist`.
- The tuner property `No Delete (marimba.subscription.nodelete)` is set to `true`.

If one of these is true, packages remain on the targets and these packages are no longer managed through Policy Manager. To actually delete these packages from the targets, you must explicitly set the packages' primary state to `uninstall`.

---

Note: If you are in multiple selection mode and are deleting the policies for multiple targets, you are deleting the policies (including all the packages and package-related information, such as states and schedules) for all targets, even though only the packages that are common to all targets are shown.

---

## ► To delete a policy from the Target View or Target Details page

- 1 From the Target View or Target Details page, select one or more targets.

**Note:** You can delete a policy from a target only if the policy is *directly assigned* to the target. The Directly Assigned To column shows the target to which a package is directly assigned.

- 2 Click Delete.

The Delete Preview page appears. This page allows you to review the policy that you are deleting from the targets. It also allows you to specify whether you want to delete the entire policy or packages only.

If you are deleting all packages that are directly assigned to the targets, you are given the option of deleting all additional properties and settings for the targets. These include tuner and package properties, blackout period, transmitter permissions, and Policy Service update schedule.

- 3 Choose one of the following options:

- Click Delete the whole policy if you want to delete all packages in the policy, as well as all additional properties and settings in the policy. These include tuner and package properties, blackout period, transmitter permissions, and the Policy Service update schedule.
- Click Delete the packages only if you want to delete all packages in the policy, but you want to keep tuner and package properties, blackout period, transmitter permissions, and the Policy Service update schedule.
- Click Cancel if you decide not to delete the policy.

---

Note: If you delete the Policy Service update schedule as a result of deleting the policy for a target, Policy Service still updates according to the schedule that was previously set. The default Policy Service update schedule is not restored. Policy Service continues to use the previously set update schedule until you set a new Policy Service update schedule.

---

You return to the Target View or Target Details page and the policy you deleted no longer appears. The packages (and, if you specified, other information in the policy) are deleted from the target the next time that Policy Service updates. If you chose to delete packages only, the directly assigned policy icon  still appears for the target. If you chose to delete the whole policy, the policy icon no longer appears for the target.

# Peer Approval Policy

In an enterprise, the software distributed through policy has to undergo an approval process before it is implemented and enforced. This feature tracks the policy changes which are deployed to an endpoint and enforce approval process on policy change. The Peer Approval policy of BBCA captures the policy change and stores it in the database till the policy is approved. When a policy change is made, an automatic e-mail is triggered to the approvers who are already specified.

In the Policy Configuration page the primary administrator can specify the approval groups from the LDAP. When a policy is created, modified or deleted for a target in LDAP, an e-mail is triggered automatically to the specified distribution group in LDAP. The e-mail notification contains details of the policy created information, and the added, modified, or deleted packages and properties.

When a policy is saved, the details of the policy are stored in the database till the policy is approved. Once the policy is approved, the policy from the database is merged with the live policy in LDAP.

When an approver logs on to Policy Manager, the approval pending policies are displayed and the approver can see details about the policy changes under the “Policy Approval” tab. the approver can either approve or reject the policy. The approver can also view the list of rejected and approved policies.

## ► To enable peer approval policy

- 1 In the Configuration page of Policy Manager, click **Peer Approval**.  
The Peer Approval Settings page appears.
- 2 Select **Enable peer approval policy** check box.
- 3 In the **Peer LDAP Groups** text box, type the required LDAP groups who will be the approvers.
- 4 To enable e-mail notifications, select **Enable E-mail notification** check box.

---

Note: BBCA sends e-mails on all the policy changes only if you select the “Enable E-Mail notification” option in Policy Manager configuration.

---

- 5 In the **To** text box, type the required e-mail addresses and distribution lists.
- 6 Click **Save**.

---

Note: After enabling peer approval policy, the policy manager channel needs to be restarted once to make sure change is reflected.

---

► **To view the list of pending policies**

- 1 In the Policy Manager, click Target View tab .
- 2 Click **Policy Approval** tab.

The Policy Approval Status page appears, where you can view the list of pending, approved and rejected policies.

- 3 To view the details of pending policies, click **Pending Policies**.

The list of pending policies appears. You can also search for a particular policy by using the **Search** text box. Type the required string in the **Search** text box to search for the required policy. The search feature matches the string with the policy name and displays the matching records. For example, if you search using the abc string, the Search feature displays all the policies where the policy name contains the abc string.

- 4 To view the details of approved policies, click **Approved Policies**.

You can view the list of approved policies.

- 5 To view the details of rejected policies, click **Rejected Policies**.

You can view the list of rejected policies.

► **To approve or reject a policy**

- 1 In the Policy Manager, click Target View tab .
- 2 Click **Policy Approval** tab.

The Policy Approval Status page appears, where you can view the list of pending, approved and rejected policies.

- 3 To view the details of pending policies, click **Pending Policies**.

- 4 To view the details of the original policy, click **View Original Policy**.

- 5 To reject or approve a policy, select the required policy and click **View Details**.

The Approve/Reject Policy Change Details dialog box appears and displays the following details:

- **Policy Name**

- Type
  - Created by
  - Created on
  - Review Comments
  - Details of added or modified or deleted channels and tuner properties
  - Details of channel configuration is displayed on tooltip text
- 6 If you want to enter any review comments, type the review comment in the **Review Comments** text box.
- 7 To approve the policy, click **Approve**.  
Policy Manager applies the policy change.
- 8 To reject the policy, click **Reject**.  
Policy Manager rejects the policy change.

---

Note: To approve the policy change, the logged in user should be part of approval group. The secondary admin users should have policy ACL write permission to approve pending policies.

---

---

Note: The logged in user cannot approve his own policy even though he is part of approver group.

---

---

Note: Operator group configured in CMS cannot be configured as peer approver group since they don't have policy read/write permission.

---

# Managing software, data, and updates

This section includes the following topics:

- “Endpoint environment management concepts” on page 260
- “Use cases” on page 260

## Endpoint environment management concepts

Policy management gives you a broad range of features for managing the state of your endpoints. You determine which packages to distribute to which targets, and when and how to install them. Policy’s function is to define the desired state of an endpoint, compare that with the actual state, and implement the policies of your enterprise. As you have seen in the preceding sections, robust configuration options are available that enable you to:

- Manage software, data, and their updates
- Assign installation states to packages (such as *Stage*, *Advertise*, or *Install*)
- Stage updates at targets and install them at a later date
- Specify if, when, and how you deliver packages by setting Primary and Secondary states
- Specify a priority that determines the order in which Policy applies changes to packages

The power and flexibility of Policy Management’s features provide multiple options, and using them together effectively relies on best practices.

The following section presents use cases that describe some of BMC’s recommended best practices for using Policy Management to control the state of software, data, and updates on your endpoints.

## Use cases

This section includes the following topics:

- “Overview of scheduling and staging updates” on page 261
- “Scheduling future update events” on page 261
- “Staging future update events” on page 262

- “Managing packages and states using install priority” on page 263
- “Flexible configuration options” on page 264

## Overview of scheduling and staging updates

Policy Manager replaces a channel automatically when the final file name segment of an existing channel’s URL matches the same segment for the new channel in the policy. Policy Manager performs the update as the first operation of the service. In addition to matching the URLs, the updateFrom function removes the replaced channel. The function optimizes file replacement to minimize bandwidth consumption, replacing only files and data that have changed with the new version. UpdateFrom follows the install order and schedule specified by the policy.



The only exception occurs when you have set the tuner property `marimba.subscription.nodelete=True`. In this case, Policy cannot remove the replaced channel because the configuration option forbids it. Best practice here is to include the replaced channel in the policy with a state of Uninstall.

## Scheduling future update events

- To schedule an installation on a future date that replaces one version of a package (*url-1*) with an updated version of the same package (*url-2*)

- 1 Create a policy that installs *url-1* for a group of targets.

Policy installs *url-1* on the specified targets.

- 2 In the policy, set a future expiration date *d* for *url-1* and add the updated version of the package (*url-2*), specifying *d* as the future installation date.

On future date *d*, Policy Service updates from *url-1* to *url-2* on each targeted endpoint.

Best practice in this case is to specify the same date for expiration of *url-1* and installation of *url-2*. The expiration of *url-1* is an *implied delete*, which (together with the addition of *url-2*) kicks off an updateFrom on date *d*. The updateFrom function optimizes the process by replacing only the files that have changed.

---

Note: Timing is key to updating applications across URLs. If the expiration of *url-1* does not coincide exactly with the installation of *url-2*, data conflicts can occur. In the above example, if *url-1* expires after the installation of *url-2*, then both channels will exist in an *Installed* state during the overlap period. If *url-1* expires before the installation of *url-2*, and you explicitly set the state of *url-1* to Uninstall, the target does not have the package at all during the interim period.

---

## Staging future update events

► **To stage a future update that replaces one version of a package (*url-1*) with an updated version of the same package (*url-2*)**

- 1 Create a policy that installs *url-1* for a group of targets.

Policy installs *url-1* on the specified targets.

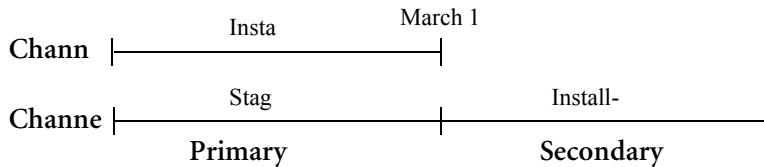
- 2 In the policy, set a future expiration date *d*, and add *url-2* with an installation state of *Stage*, specifying *d* as the installation date.

At date *d*, Policy performs an *updateFrom* (*url-1* to *url-2*).

Staging future updates is similar to scheduling — specify the same date *d* for both expiration of the first URL and installation of the second.

Policy enables you to set a Primary and Secondary state for a package, so you can set independent scheduling options for each state. In the following example, you stage Channel 2 (an updated application version) to replace Channel 1 (a current application version) on March 1. The older version expires on March 1, and the *updateFrom* function optimizes the installation of the new version. Note the states you assign to Channel 2 before and after the update.

Figure 13-3: Using primary and secondary states to stage a future update.



## Managing packages and states using install priority

### ► To specify prerequisite channels for installation prior to updating an existing channel

- 1 Create a policy that installs, say, channels 2 through 5 (in that order) for a group of targets.

Policy installs channels 2 through 5 on the specified targets.
- 2 Modify the policy to install a prerequisite channel before updating an existing channel in the policy.
  - a Add Channel 1 with a state of *Install* in position #1.
  - b Remove Channel 3 (*url-1*) and replace it with the updated Channel 3 (*url-2*), still in position #3. Follow the guidelines found in the previous use cases for scheduling and staging updates.

Policy processes the channels in the following order:

- a Install Channel 1
- b Install Channel 2
- c Update Channel 3 (*url-1* to *url-2*)
- d Install Channel 4
- e Install Channel 5

Policy supports the previous scheduling and staging scenarios when using install priority features to specify the package installation sequence.

## Flexible configuration options

The powerful configuration options available in Policy Manager enable you to implement a wide range of creative deployment scenarios. For example, say you want to distribute an application as a “one-time” offer for a restricted time period, such as the month of January. If a target doesn’t check in during the month of January, you don’t want that target to have the application. If you want users who update in January to have the software on February 1, you must specify an installation state of *Install-Persist*.

### ► To distribute an application as a one-time offer in January

- 1 Add the application’s URL to a policy.
- 2 Set the policy to expire on January 31.
- 3 Set the installation state to *Install-Persist*.

## Editing policies from Package View

The Package View page provides a convenient alternative to the Target View page. You can use it when you want to view the targets that are assigned one or more common packages. While on the Package View page, you might want to add targets, remove targets, or edit policies.

### ► To add or remove targets on the Package View page

- 1 Click the Package View tab.
- 2 On the left side of the page, select one or more packages. See the following topics:
  - “Browsing packages” on page 176
  - “Searching for packages” on page 177
  - “Viewing targets that have been assigned a package” on page 177

Each time you select a package, it appears in the list of selected packages on the top of the right side of the page. To remove a package from that list, click the Remove icon  next to it.

Below the list of selected packages is a list of targets that have been *directly* assigned *all* selected packages (the *intersection* of the policies for the selected packages). If a target has not been assigned *all* selected packages, it does not appear in the list. The targets that appear might or might not have already received the packages shown.

- 3 To add targets:
  - a Click Add.
  - b Use the left side of the page to locate and choose a target. See “Viewing targets” on page 164.

Each time you select a target, it appears in the list of selected targets on the right side of the page.
  - c For each target, specify a package state and schedule:
    - For the package state, choose a state under the Primary and Secondary State columns.
    - For the schedule, select the targets for which you want the schedule to apply and click Set Common Schedule.

See “Specifying states and schedules for packages in a policy” on page 215.
  - d When you are finished adding targets, click Add.
- 4 To remove targets:
  - a Select one or more targets from which you want to remove the common packages.
  - b Click Remove.
  - c On the confirmation page, click Remove.

After adding or removing targets, you return to the Package View page.

## ► To edit policies from the Package View page

- 1 Click the Package View tab.
- 2 On the left side of the page, select one or more packages. See the following topics:
  - “Browsing packages” on page 176
  - “Searching for packages” on page 177
  - “Viewing targets that have been assigned a package” on page 177

Each time you select a package, it appears in the list of selected packages on the top of the right side of the page. To remove a package from that list, click the Remove icon  next to it.

Below the list of selected packages is a list of targets that have been *directly* assigned *all* selected packages (the *intersection* of the policies for the selected packages). If a target has not been assigned *all* selected packages, it does not appear in the list. The targets that appear might or might not have already received the packages shown.

- 3 Click Edit.
- 4 Make your changes to the policy. See “General directions for creating and editing policies” on page 209.

After editing the policy, you return to the Package View page.

## Specifying policies for OS migration

You can use the Target View page to specify and edit policies that are applied during migration to the Windows 7 operating system.

### ► To specify policies for OS migration

- 1 Click the Target View tab.
- 2 Click Edit.
- 3 Click the OS Migration tab.

The OS Migration page is used to assign the OS migration policy to a selected group or collection.

- 4 Select **Apply the following OS Template settings on endpoint**.
  - a Specify the activation time (that is, the time when the OS migration is planned).
  - b Specify the expiration time beyond which the OS migration must not happen.
  - c Specify the Template variables specific to your region or locale.
  - d Select the OS Template that will be applied after migration commences.
  - e Specify the Personal Backup settings if you require a backup to be made before migration commences.

---

Note: The OS Template and Personal Backup Templates can be created and modified in the Empirum Console. For more information, see the [BMC Marimba Client Automation OS Migration with Matrix42 User Guide](#).

---

## Specifying personal backup settings for OS migration

The personal backup feature allows you to restore user settings and data on the clients after an OS migration activity. Based on the migration schedule, the Policy Service sends a request to the Policy plug-in to identify the local depot for the endpoint. The Policy plug-in finds the local depot configured for the endpoint from the Subnet-based Repeater Policy (SBRP) configuration. The personal backup feature uses these local depot access details.

After the backup operation is complete, the endpoint sends a request to the Empirum Server to enable PXE activation for the endpoint. The PXE activation enables the endpoint for OS migration.

If backup is not selected for a particular endpoint, the endpoint sends a request to the Empirum Server after the OS migration activation time has been reached.

### ► To specify personal backup settings for OS Migration

- 1 Click the Target View tab.
- 2 Click Edit.
- 3 Select the OS Migration tab.

The OS Migration page is used to assign the OS migration policy to a selected group or collection.

- 4 Select the **Personal Backup Settings** check box, and select the Personal Backup template.
- 5 Save the policy.

---

Note: The Personal Backup templates are loaded from the Empirum Server.

Ensure that you have configured the Empirum connection settings in the CMS Console. You can use the Empirum Console to create and edit the templates. You must ensure that the templates have been created before the policy is assigned.

---

---

Note: You must set the following tuner property on the endpoint to enable personal backup: `marimba.subscription.osm.backup.enabled=true`.

---

The Wake on WAN Schedule can be enabled for OS migration policy. When you enable this option, the machines are waked onthe network when the migration is activated.

---

Note: The OS Migration Service channel is responsible for executing personal backup, activating PXE boot and rebooting the machine on schedule.

Ensure that this channel is also published at the same location as the Policy Service on the Master Transmitter.

---

---

Note: For details about the SBRP configuration, see the *BMC Marimba Client Automation Transmitter and Proxy Guide*.

---

## Scheduling a personal backup through a policy

The personal backup for an end-point can be scheduled through a policy. On policy update at endpoint, the OS Migration service channel will be subscribed and personal backup schedule will be assigned as its update schedule. On update schedule, the data at endpoint is backed-up using the specified Personal Backup template.

---

Note: When OS Migration is initiated for an end-point, the backup can be configured to use previously used Backup Template using the **Use last applied template for personal backup option** in OS Migration tab.

---

You can navigate to the Personal Backup tab under OS deployment tab in Policy Manager's Edit Policy page.

# Specifying the Policy Service schedule for a target

This section describes how you set the schedule for Policy Service updates. It includes the following topics:

- “What is the Policy Service?” on page 270
- “Setting the schedule for Policy Service updates” on page 271

## What is the Policy Service?

Policy Service resides on each target managed by Policy Manager. The service is implemented as a channel and has no graphical user interface (GUI). The Policy Service channel is responsible for applying the policies assigned to the target tuner on which it is running.

On each endpoint where it is running, Policy Service contacts the transmitter where the Policy Service plug-in has been published, identifies its endpoint, and receives a policy from the transmitter. Policy Service then applies the policy—downloading and installing packages as required.

Policy Service starts at scheduled intervals and stops after it applies all installation states and properties required by the policy.

## Setting the schedule for Policy Service updates

This topic describes how to set the update schedule for Policy Service. If you do not set a schedule, the default is to update every 90 minutes.

If you have set a blackout period for the target, you can exempt Policy Service from the blackout period. This exemption is useful if you anticipate emergency situations when you want to update or install packages during the blackout period. See “Setting the blackout period for a target” on page 196.

When multiple users edit the Policy Service Schedule page for the same target, the most recent edits to be saved are applied.

---

Note: You can set target-level settings, such as the update schedule for Policy Service, for one target at a time only. You cannot set target-level settings if you are in multiple-selection mode and are editing policies for multiple targets at one time. However, you can set target-level settings for all targets by selecting the All Endpoints target.

---

This procedure assumes you are on the Packages tab of the Edit Policy page. See “General directions for creating and editing policies” on page 209.

### ► To set the update schedule for Policy Service

- 1 On the Edit Policy page, click the Policy Service Schedule tab.
- 2 Choose Use Policy Manager to set the Policy Service update schedule.

If you do not choose this option, you cannot set a schedule. Policy Service follows the default update schedule (every 90 minutes).

- 3 In the Recurrence section, specify how often you want to update Policy Service:
  - **Daily**—Policy Service can be updated every day, only on weekdays (Monday through Friday), or every 1 to 365 days, depending on the number you set.
  - **Weekly**—Policy Service can be updated every 1 to 52 weeks, depending on the number you set. During the week when an update is scheduled, an update occurs only on the days you select. You must select at least one day of the week.

- **Monthly**—Policy Service can be updated every 1 to 12 months on the day you set. If you choose day 30 or 31 and the month when an update occurs doesn't have that day (such as February), the update occurs on the first day of the next month.
- 4 In the Update time section, specify the time of the day when you want to update the Policy Service:
- **Update at**—Specify one time of the day when updates occur.
  - **Update every**—Specify multiple times during the day when updates occur. You can set updates to happen every specified number of minutes or hours during the day, and you can limit updates to a single range of hours during the day. For example, you can have updates occur every hour between 9 AM and 5 PM or every 30 minutes from 5 PM to 9 AM.

When you are finished editing the policy, preview and save the policy. See “Previewing and saving policy changes” on page 211.

---

Note: If you delete the Policy Service update schedule as a result of deleting the policy for a target, the Policy Service still updates according to the schedule that was previously set. The default Policy Service update schedule is not restored. The Policy Service continues to use the previously set update schedule until you set a new Policy Service update schedule.

---

## Specifying reboot settings for Windows targets

The Reboot Settings tab on the Edit Policy page allows policy-based control of reboots on Windows endpoints. From this tab you can specify the reboot schedule after a policy is pushed to a Windows endpoint.

### ► To set the reboot schedule for Windows targets in a policy

- 1 From the Edit Policy page, click the Reboot Settings tab.
- 2 Click the Reboot Schedule drop-down and select one of the following:
  - Follow the reboot schedule at the endpoint—allows the Common Reboot Service settings on the endpoint to determine its own reboot schedule.
  - Change the reboot schedule at the endpoint—specifies a specific reboot schedule for the endpoint. You can set a daily, weekly, or monthly reboot schedule.

## Setting Power Options for Windows targets

The Power Options tab on the Edit Policy page allows policy-based control of the Windows Power Options on supported Windows endpoints in which the Advanced Configuration and Power Interface (ACPI) is supported.

Power Options are set through Power Options profiles as described in “Creating and managing profiles for Windows Power Options” on page 156.

User settings take precedence when a user changes the Power Options properties manually on the endpoint. The Power Options settings are reapplied in the following two cases:

- when the administrator changes a Power Options setting in Policy Manager
- when the **Apply Power Options Properties Every Time Policy Service Runs** option is selected in the Power Options profile

### ► To apply Power Options profiles for Windows targets in Policy Manager

- 1 From the Edit Policy page, select a target.
- 2 Click the Power Options tab.
- 3 Click Set power options applicable to the profile.

4 Select a profile.

During a policy update, these properties are pushed to the endpoint along with the other policy items. The Policy Service first subscribes to all the channels in the policy and then applies the tuner and channel properties.

---

Note: If an endpoint user sets System standby to “Never,” then this value on the endpoint is not overridden with any other values even if the value is updated in Policy Manager. The limitation is caused by the inability of ACPI (Advanced Configuration and Power Interface provided by Microsoft) to reset the standby time if the previous value is set to Never.

---

# Setting tuner and package properties for a target

The following topics describe tuner and package properties and how you can set them using Policy Manager:

- “Overview of tuner and package policies” on page 275
- “What are tuner properties?” on page 276
- “What tuner properties do you commonly set with Policy Manager?” on page 276
- “What are package properties?” on page 282
- “Properties that control reboots for packages (Windows only)” on page 283
- “Setting tuner properties” on page 285
- “Setting package properties” on page 286
- “Tuner and package properties format” on page 287
- “Deleting tuner and package properties” on page 289
- “Conflict resolution: property values” on page 291
- “Conflict resolution: When multiple users edit properties” on page 292

## Overview of tuner and package policies

When you create or edit a policy using Policy Manager, you have the option of specifying tuner and package properties for a target. Tuner and package properties allow you to control the behavior of the target or packages on the target endpoints, such as to control the number and frequency of retry attempts for a failed download or to control the reboot behavior of a particular package.

When you set properties using Policy Manager for a channel or package that has not been installed yet, the properties are applied when the channel is installed.

This section includes the following topics:

- “What are tuner properties?” on page 276
- “What tuner properties do you commonly set with Policy Manager?” on page 276

- “What are package properties?” on page 282
- “Properties that control reboots for packages (Windows only)” on page 283

## What are tuner properties?

Tuner properties configure many characteristics and settings of a tuner, including the URL of its primary channel and the URL for tuner updates, as well as licenses and proxy details. You can use Policy Manager to add, edit, and delete tuner properties (in the `prefs.txt` file) for the target tuner.

For a lists of the tuner properties that you usually set through Policy Manager, see “What tuner properties do you commonly set with Policy Manager?” on page 276. For a list of tuner properties, see the chapter about tuner properties in the *BMC Marimba Client Automation Reference Guide*, available on the BMC Customer Support website.

For instructions about setting tuner properties, see “Setting tuner properties” on page 285.

## What tuner properties do you commonly set with Policy Manager?

If you have Policy Service installed, the tuner has additional properties that become available. You can set the following tuner properties to change the tuner’s interaction with Policy Service through one of the following methods:

- When a tuner installer is created during setup and deployment
- Through a profile update
- Through Tuner Administrator
- Through Policy Manager

The first nine properties that appear in Table 13-1 on page 277 are the ones that appear in the list on the Policy Manager's Tuner and Package Properties page. They are listed in the order that they appear on the page. The rest of the properties in the table need to be entered by their complete property names; they are listed in alphabetical order.

Table 13-1: Commonly used tuner properties

Property name	Possible values	Description
<b>No Delete</b>	true or false	By default, packages are deleted when a user or machine is removed from a group in the underlying directory service. If you set this property to true, packages are not deleted, even if you delete packages from targets using the Target Details page. They remain on the targets and are no longer managed using Policy Manager. You must use the uninstall state to delete the packages from targets.
<b>Retry Time</b>	integer Default: 60 (60 seconds or 1 minute)	This property sets the delay, in seconds, before Policy Service tries to subscribe failed packages. See “Policy Management state verification and retry” on page 126.
<b>Retry Count</b>	integer Default: 5	This property sets the number of retries before Policy Service gives up trying to subscribe failed packages. See “Policy Management state verification and retry” on page 126.
<b>Use Shortcuts</b>	true or false Default: false	When this property is set to true, advertised packages are represented as desktop shortcuts (Windows platforms only).

Table 13-1: Commonly used tuner properties(Continued)

Property name	Possible values	Description
<b>Install Mode</b>  marimba.subscription.installmode	silent or aspackaged  Default: silent	This property defines the mode used by Policy Service while installing or uninstalling channels. You can set it to the following values: <ul style="list-style-type: none"><li>n <b>silent</b>—Policy Service installs all packages on the target tuner in silent mode. This is the default value.</li><li>n <b>aspackaged</b>—Policy Service uses the installation mode specified in the package.</li></ul> If you want a completely silent installation, you might want to set these tuner properties to prevent progress bars or error dialog boxes from appearing on endpoints: “marimba.tuner.display.noprogress” on page 281 and “marimba.tuner.display.noerrors” on page 281.
<b>Re-apply Config on Failure</b>  marimba.subscription.reapplyconfigonfail	true or false  Default: true	This property determines whether or not to reapply a cached policy if Policy Service cannot communicate with the plug-in to get an updated policy.  When set to true, Policy Service reapplies the cached policy (the last policy downloaded) if it cannot communicate with the plug-in to get an updated policy. When set to false, Policy Service does not attempt to reapply the cached policy.  Added in version 6.0.

Table 13-1: Commonly used tuner properties(Continued)

Property name	Possible values	Description
marimba.subscription.adminusers	<p>&lt;user1&gt;, &lt;user2&gt;,... which specifies a comma-delimited list of administrators. The list of administrators can contain the name of any user who can log in at the endpoint machines.</p> <p>Default: none</p>	<p>If the endpoint user's machine has Policy Service 5.0.1 or higher, you can set this tuner property to specify the administrators who you want to be able to log in to a user's machines temporarily when using user-based targeting (possibly for troubleshooting). The property prevents the user's channels from getting deleted when you log in as an administrator.</p> <p>When the administrator logs in, the following happens:</p> <ul style="list-style-type: none"> <li>▀ Channels subscribed for another user are not removed.</li> <li>▀ If there are channels that have been assigned to an administrator, these channels are delivered to the endpoint.</li> <li>▀ The properties that have been set for the administrator are set. Possibly, the properties set for the users are overwritten by those set for the administrator. However, when the users log back in and Policy Service updates, the users get their properties back.</li> </ul>

Table 13-1: Commonly used tuner properties(Continued)

Property name	Possible values	Description
subscription.autostart.channel	<p>&lt;channel_name&gt;, “&lt;argument1&gt;, &lt;argument2&gt;, &lt;argument3&gt;”, true or false</p> <p>Default: no value</p>	<p>Allows you to start the specified channel after Policy Service runs.</p> <ul style="list-style-type: none"> <li>▪ &lt;channel_name&gt; specifies the name of the channel that you want to start. It is required.</li> <li>▪ “&lt;argument1&gt;, &lt;argument2&gt;, &lt;argument3&gt;” are any arguments you want to pass to the channel. They are optional.</li> <li>▪ true or false specifies whether or not to update the channel before starting it. This last argument is optional.</li> </ul>
marimba.subscription.machinename	<p>a string representing the name of the machine</p> <p>Default: no value</p>	<p>Allows you to override the machine name returned by Policy Service. The name you enter here must match the name used in the machine's flat file. By default, this property has no value and the machine name is simply the DNS host name, without the domain information.</p>
marimba.subscription.retryintervalsec	<p>integer</p> <p>Default: 30 (30 seconds)</p>	<p>This property sets the interval to wait, in seconds, before retrying the connection to LDAP. You set this property on the tuner hosting the transmitter where you have published the Policy Service plug-in.</p>

**Example:**

```
subscription.autostart.channel=ChannelManager,"arg1,arg2,arg3",true
```

**Note:** Be careful when using this property with BMC Marimba Client Automation channels. Some channels might be configured to start Policy Service and cause recursion.

Table 13-1: Commonly used tuner properties(Continued)

Property name	Possible values	Description
marimba.subscription.timeout	integer  Default: 3600 (3600 seconds or 1 hour)	This property sets the maximum period of time, in seconds, that Policy Service waits for an operation to complete (such as a package sending a notification that it has achieved a specified state) before proceeding.
marimba.subscription.varytime	integer  Default: 10 (10 minutes)	This property sets the maximum period of time, in minutes, that scheduled events (such as downloading or updating a package) can be postponed. This improves transmitter performance by spreading out endpoint requests during periods of heavy load. For example, if this property is set to 10 minutes and an event is scheduled to occur at 10:00 AM, the event can occur any time between 10:00 and 10:10 AM.
marimba.tuner.display.noerrors	true or false  Default: false	This property indicates whether or not the tuner shows error and warning dialogs. To hide error and warning dialogs, set this property to true. The error and warning messages are printed out to a system console. Note that this property does not apply if you are running the tuner without a display.
marimba.tuner.display.noprogress	true or false  Default: false	This property indicates whether or not the tuner shows a progress bar when channels are being subscribed to or are being updated. To hide progress bars, set this property to true. Note that this property does not apply if you are running the tuner without a display.

For details about all available tuner properties, see the tuner properties chapter in the *BMC Marimba Client Automation Reference Guide*, available on the BMC Customer Support website.

## What are package properties?

*Package properties* provide information about the package and how it should run. These are also called *channel properties*. This information is typically used by the target tuner. Packages, which are usually applications that have been packaged into channels (using Application Packager), have additional properties that are referred to as channel parameters. *Channel parameters* provide information and settings used when installing and running packages.

Package properties and parameters are stored in the `properties.txt` file and the `parameters.txt` file, respectively. These files are usually created (in the channel directory) when the channel or package is created. You can edit the files manually or use Channel Copier or Publisher.

For a list of package properties and channel parameters, see the chapters about channel properties and parameters in the *BMC Marimba Client Automation Reference Guide*, available on the BMC Customer Support website.

Package properties and parameters usually appear in the following form:

`<property_name>=<property_value>`

For instructions about setting package properties, see “Setting package properties” on page 286.

## What is property priority?

If you have included an endpoint as a member of more than one group, you can set different values for a single tuner or package property in each group’s policy. When the transmitter sends different values for a single tuner or package property during a policy update, the Policy Service enforces the property value that has the highest priority value as set from the Tuner/ Package Properties tab on the Edit Policy page and described in “To set tuner properties for a target” on page 285 and “To set package properties for a target” on page 286.

## Properties that control reboots for packages (Windows only)

During the installation of packages created using Application Packager, some files cannot be written because they have been locked by the operating system. Most incidents of locked files occur because a file is open or a DLL is currently loaded. In such cases, the new file is saved to disk under a temporary name and marked to be renamed to the correct name and location upon system reboot.

Policy Service might install several packages during a single policy update. If several packages are being installed in one session, Policy Service postpones the system reboot until all packages have been updated and installed. The reboot can occur even if the installation fails, depending upon how far the installation progressed.

Policy Service can take into account the reboot properties of individual Application Packaged packages. These package properties are described in Table 13-2 on page 284. Two of these properties, `reboot.allow` and `reboot.force`, determine the reboot behavior, as shown in Table 13-3 on page 284. Two other package properties, `reboot.showdialog` and `reboot.allowcancel`, allow control over the dialog box displayed when a reboot is required.

For Tuner versions 8.0 and later:

- Policy Manager uses the `marimba.reboot.never` property set at the endpoint rather than the `marimba.subscription.reboot` option that was used prior to version 8.0.
- Policy Manager ignores the Allow Reboot and Reboot Time tuner properties. These options have also been removed from the Policy Manager user interface.

When the Reboot Schedule is set at the endpoint using Policy Manager or Infrastructure Administrator, the following reboot behavior exists when a package is distributed using the Policy Service:

- If the package does not require immediate reboot, the tuner follows the Reboot Schedule.
- If the package requires an immediate reboot, the tuner reboots immediately and ignores the Reboot Schedule.

Table 13-2: Reboot package properties

Property name	Possible values	Description
reboot.allow	true or false Default: true	This parameter indicates that the channel can reboot the machine if required. By default, it is set to true. Set the parameter to false if this behavior is undesirable. <b>Note:</b> Channels do <i>not</i> cause machines to reboot automatically in silent mode, unless the reboot.force parameter is set to true
reboot.allowcancel	true or false Default: true	If this parameter is set to true and a reboot is required on the endpoint, a reboot option allowing the user to cancel the reboot appears in the reboot dialog box. Even when the tuner is running in silent mode, the Restart Required window appears with the Cancel option.
reboot.force	true or false Default: false	When set to true, this parameter forces the machine to reboot after the end of installation. By default, it is set to false. The Reboot dialog appears with the Snooze option but without the Cancel option. <b>Note:</b> The appearance of the Reboot dialog depends on the tuner's Common Reboot Service settings. If the reboot schedule is configured, the reboot will be postponed to the scheduled time.
reboot.showdialog	Valid values: true or false Default: true	This parameter indicates that, in interactive mode (not silent or semisilent mode), the channel should display the reboot dialog box, if necessary. The appearance of the Reboot dialog depends on the tuner's Common Reboot Service settings. If reboot interaction is disabled in the Common Reboot Service, then no dialog is displayed.

Table 13-3: Effect of reboot.allow and reboot.force on system reboot

reboot.allow	reboot.force	System reboot?
true	true	Yes. Policy Service reboots the system whether or not a reboot is required.
true	false	Yes, if necessary. Policy Service checks if a reboot is required and reboots the system if necessary.

Table 13-3: Effect of reboot.allow and reboot.force on system reboot

reboot.allow	reboot.force	System reboot?
false	true	No
false	false	No

If both `reboot.allow` and `reboot.force` are set to true for any of the packages installed by Policy Service, Policy Service reboots the system whether or not a reboot is required. If all packages installed by Policy Service have `reboot.force` set to false but at least one package has `reboot.allow` set to true, Policy Service checks if a reboot is required and reboots the system, if necessary.

## Setting tuner properties

This topic describes how to set tuner properties for targets. Tuner properties configure many characteristics and settings of the tuner running on the endpoints. See “What are tuner properties?” on page 276.

---

Note: You can set target-level settings, such as the tuner and package properties, for one target at a time only. You cannot set target-level settings if you are in multiple-selection mode and are editing policies for multiple targets at one time. However, you can set target-level settings for all targets by selecting the All Endpoints target.

---

This procedure assumes you are on the Packages tab of the Edit Policy page. See “General directions for creating and editing policies” on page 209.

### ► To set tuner properties for a target

- 1 On the Edit Policy page, click the Advanced > Tuner/Package Properties tab. The Tuner and Package Properties page appears.
- 2 For tuner properties, you can select from the list of commonly used properties or you can enter the property directly in the Properties text box:
  - For the commonly-used tuner properties (for a list, see “What tuner properties do you commonly set with Policy Manager?” on page 276), select the property from the list, specify a value, and click Apply.

The property is added to the Properties text box.

- For any tuner property, you can enter the property name and value directly in the Properties text box. Follow the format described in “Tuner and package properties format” on page 287.
- 3 Review the properties listed in the Properties text box.

**Note:** Deleting a property from the Properties text box does not remove it from the target endpoints. This causes the property to remain as is on the target tuner (not managed by Policy Management). For instructions about deleting properties, see “Deleting tuner and package properties” on page 289.
  - 4 To set the priority for all the properties in the Properties text box, enter a number in the Priority field and click the Apply button.

Property priority is described in “What is property priority?” on page 282.
  - 5 When you are finished editing the policy, preview and save the policy. See “Previewing and saving policy changes” on page 211.

## Setting package properties

This topic describes how to set package properties for packages. *Package properties* provide information about the package and how it should run. See “What are package properties?” on page 282.

---

**Note:** You can set target-level settings, such as the tuner and package properties, for one target at a time only. You cannot set target-level settings if you are in multiple-selection mode and are editing policies for multiple targets at one time. However, you can set target-level settings for all targets by selecting the All Endpoints target.

---

This procedure assumes you are on the Packages tab of the Edit Policy page. See “General directions for creating and editing policies” on page 209.

### ► To set package properties for a target

- 1 On the Edit Policy page, click the Advanced > Tuner/Package Properties tab. The Tuner and Package Properties page appears.
- 2 For package properties, perform the following steps in the Package Properties table:
  - a From the Package list, select the package or group of packages for which you want to set properties. You can select one of the following:

- The URL for a specific package—Select this option if you want to set the property for one specific package only.
  - All packages—Select this option if you want to set the property for all packages on the endpoint tuners, except for the Policy Service channel.
  - All subscribed packages—Select this option if you want to set the property for all packages that are part of the policy, except those that have these states `uninstall`, `exclude`, `primary`, or `advertise`. Also, the Policy Service channel is not included.
  - Policy Service channel—Select this option if you want to set the property for the Policy Service channel only.
- b Enter the property name and value.
- c Click Apply.

The property is added to the Properties text box.

**Note:** You can also enter the package URL, package property name, and value directly in the Properties text box. Follow the format described in “Tuner and package properties format” on page 287.

- 3 Review the properties listed in the Properties text box.

---

Note: Deleting a property from the Properties text box does not remove it from the target endpoints. This causes the property to remain as is on the target tuner (not managed by Policy Management). For instructions about deleting properties, see “Deleting tuner and package properties” on page 289.

---

- 4 To set the priority for all the properties in the Properties text box, enter a number in the Priority field and click the Apply button.  
Property priority is described in “What is property priority?” on page 282.
- 5 When you are finished editing the policy, preview and save the policy. See “Previewing and saving policy changes” on page 211.

## Tuner and package properties format

After you add tuner or package properties, they appear in the Properties text box in the following format:

`<property>[ ,scope]=<value>`

This format is also used when entering tuner or package property key-value pairs directly in the text box or from the command line.

Your entries for *<property>* and *<scope>* can take one of the following forms:

- Tuner properties:

*<property>=<value>*

Example:

marimba.subscription.retrycount=100

- All subscribed packages:

*<property>,subscribers=<value>*—Sets a property in all packages that are part of the policy, except channels that have these states `uninstall`, `exclude`, `primary`, or `advertise`.

- All packages:

*<property>,\*=<value>*—Sets a property in all packages on the tuner *except* the Policy Service channel.

- One specific package:

*<property>,<url>=<value>*—Sets a property in the package specified by *<url>*.

- Policy Service channel:

*<property>,service=<value>*

---

#### Notes:

- Deleting a property from the Properties text box does not remove it from the target endpoints. This causes the property to remain “as is” on the target tuner (not managed by Policy Manager). You can remove a tuner or package property from the target endpoints by setting its value to *<null>* as in the following example:

*<property>=null*

- Spaces are not allowed before or after the equals sign (=) or after the property value. For example, the following example is in the proper format:

*<property\_name>=<property\_value>*

But the following example is not:

`<property_name> = <property_value>`

Depending on the property, spaces within the property value can be acceptable when the property value itself contains spaces. For example:

`marimba.schedule.filter=ANYTIME on sun+mon+tue`

---

## Deleting tuner and package properties

If you no longer want tuner or package properties to be set on the endpoint tuners, you can delete it using Policy Manager. You do this by setting the property's value to `<null>`, including the angle brackets (`<>`), or leaving the value blank, as in `<no value>`.

When you delete a tuner property using Policy Manager, it is deleted from the `prefs.txt` file in the tuner's workspace directory and usually it no longer takes effect. However, if you previously had set the property when packaging the tuner, it cannot be deleted through Policy Manager. This is because setting the property during packaging sets the property in the `properties.txt` file located in the tuner's workspace directory (typically, this is `C:\Program Files\Marimba\Tuner\marimba\Marimba\properties.txt` on Windows), and Policy Manager does not remove the property from the `properties.txt` file. Usually, the tuner properties in the `prefs.txt` file override the default tuner properties in the `properties.txt` file. In this case, however, the tuner property only exists in the `properties.txt` file, so it takes effect.

---

Note: You can set target-level settings, such as the tuner and package properties, for one target at a time only. You cannot set target-level settings if you are in multiple-selection mode and are editing policies for multiple targets at one time. However, you can set target-level settings for all targets by selecting the All Endpoints target.

---

This procedure assumes you are on the Packages tab of the Edit Policy page. See “General directions for creating and editing policies” on page 209.

## ► To delete properties from targets

- 1 On the Edit Policy page, click the Advanced > Tuner/Package Properties tab.

The Tuner and Package Properties page appears.

- 2 For tuner properties, find the Properties text box and enter the property name and value with the value set to <null>, including the angle brackets (<>), or leaving the value blank.

For example: marimba.subscription.retrycount=<null>

Or: marimba.subscription.retrycount=<no value>

Follow the format described in “Tuner and package properties format” on page 287.

- 3 For package properties, perform the following steps in the Package Properties table:

- a From the Package list, select the package or group of packages for which you want to set properties. You can select one of the following:

- **The URL for a specific package**—Select this option if you want to set the property for one specific package only.
- **All packages**—Select this option if you want to set the property for all packages on the tuner (for the targets), except for the Policy Service channel.
- **All subscribed packages**—Select this option if you want to set the property for all packages that are part of the policy, except those that have these states `uninstall`, `exclude`, `primary`, or `advertise`. Also, the Policy Service channel is not included.

- **Policy Service channel**—Select this option if you want to set the property for the Policy Service channel only.

- b Enter the property name and value, with the value set to <null> or by leaving the value blank, as in <no value>.

- c Click **Apply**.

The property is added to the Properties text box.

**Note:** You can also enter the package URL, package property name, and value directly in the Properties text box. Follow the format described in “Tuner and package properties format” on page 287.

- 4 Review the tuner and package properties listed in the Properties text box.

---

Note: Deleting a property from the Properties text box does not remove it from the target endpoints. This causes the property to remain as is on the target tuner (not managed by Policy Manager). You remove a tuner or package property from the target endpoints by setting its value to <null>.

---

When you are finished editing the policy, preview and save the policy. See “Previewing and saving policy changes” on page 211.

## Conflict resolution: property values

Without attention to best practices, tuner property and channel property conflicts can arise when you set different values for the same property in different groups. This section describes how to resolve property value conflicts. For more information about tuner and package properties, see “Setting tuner and package properties for a target” on page 275.

**Tuner properties.** If you have included an endpoint as a member of more than one group, you can in theory set different values for a single tuner property in each group’s policy. When the transmitter sends different values for a single tuner property during a policy update, the Policy Service enforces the property value that has the highest priority value as set from the Tuner/ Package Properties tab on the Edit Policy page and described in “To set tuner properties for a target” on page 285.

If you do not set the priority of the property to resolve value conflicts, you cannot predict which value prevails.

**Package properties.** If a package property is specified both for all subscribed packages and for an individual package, the property value for the individual package takes precedence. For example, consider these property settings:

semisilent=false for all subscribed packages  
(The syntax is semisilent,subscribers=false.)

semisilent=true for the WordProcessorApp package  
(The syntax is semisilent,http://trans.company.com:5282/  
Applications/WordProcessorApp=true.)

In this case, the value of the property semisilent for the WordProcessorApp package is set to true. If the property semisilent is Null for other subscribed packages, Policy Manager sets the value to false.

You can also set the priority of the for the package property. When the transmitter sends different values for a single package property during a policy update, the Policy Service enforces the property value that has the highest priority value as set from the Tuner/Package Properties tab on the Edit Policy page and described in “To set package properties for a target” on page 286.

## Conflict resolution: When multiple users edit properties

If more than one user edits the tuner and package properties page at the same time, and two users edit the same property, the last user to save changes to the page overwrites other changes to that property.

### ► To discern outcomes when multiple users edit policies

- 1 User 1 goes to the Tuner and Package Properties page and edits properties for target Machine A.
- 2 Before User 1 saves changes to the properties, User 2 edits and saves the properties for the same target.
- 3 User 1 saves changes after User 2 has left the page.

The policy that is originally loaded by both users is Policy A. Two copies of Policy A are created: Policy A1, the copy that User 1 is editing, and Policy A2, the copy that User 2 saved. The following table describes the results for the different cases:

Table 13-4: When multiple users edit properties

User 1:	User 2:	Final result
Adds a <i>new</i> property that is not present in Policy A.	Edits Policy A, but does not add, edit, or delete the same property.	Policy Manager adds the new property to Policy A.
Adds a <i>new</i> property that is not present in Policy A.	Sets a different value for the same property.	Policy Manager adds the new property to Policy A with the value specified by User 1.
Removes a property from Policy A.	Edits Policy A, but does not edit or delete the same property.	Policy Manager removes the property from Policy A.
Removes a property from Policy A.	Edits the same property (sets it to some value) in Policy A.	Policy Manager removes the property from Policy A.

Table 13-4: When multiple users edit properties

User 1:	User 2:	Final result
Edits Policy A, but does not add, edit, or delete the same property as User 2.	Edits or removes a property in Policy A.	Policy Manager changes or removes the property User 2 worked on.
Modifies the value for a property in Policy A.	Modifies the same property in Policy A.	Policy Manager saves the property with the value User 1 specified.
Deletes packages from Policy A.	Modifies package properties for packages deleted by User 1.	Policy Manager deletes the packages and does not save the properties User 2 modified.
Adds new packages to Policy A.	Edits Policy A and modifies properties unrelated to the new packages added by User 1.	Policy Manager saves the packages User 1 added.

For information about conflict resolution when multiple users edit the same policy, see “Conflict resolution: when multiple users edit the same policy” on page 252.



# Chapter 14 Integration with Patch Management

This chapter describes how you can use Policy Manager with the BMC Patch Management module to deploy patches to target machines.

The following topics are provided:

- Prerequisites for integration with Patch Management (page 296)
- What is a patch group? (page 296)
- What is a patch group assignment state? (page 297)
- What is Patch Service? (page 297)
- Assigning patch groups to targets (page 297)
- Simulating the installation of patches (page 302)
- Viewing more details about the installation of patch groups (page 303)
- Overriding the Patch Service update schedule for target machines (page 304)
- Exempting Patch Service from the blackout period (page 305)
- Policy compliance for patch groups (page 307)

The BMC Patch Management solution enables you to figure out what patches are needed by the machines in your enterprise, group and configure the patches, and publish them to a transmitter. Then, you use Policy Manager to deploy patch groups to target machines that need them. For more information about the BMC Patch Management solution, see the *BMC Marimba Client Automation Patch Management Users Guide*, available on the BMC Customer Support website.

## Prerequisites for integration with Patch Management

For information about installing the components mentioned in this section, see the *BMC Marimba Client Automation Installation Guide*. For information about configuring Patch Manager and Patch Service, see the *BMC Marimba Client Automation Patch Management User Guide*. For information about configuring Report Center and Scanner Service, see the *BMC Marimba Client Automation Report Center User Guide*. These documents are available on the BMC Customer Support website.

**Server side.** You must have the following components installed and configured on the same machine as Policy Manager:

- Patch Manager
- Report Center

**Client side.** You must have the following components installed on the endpoints with Policy Service:

- Patch Service
- Scanner Service

## What is a patch group?

A *patch group* is a list of one or more patch IDs, each of which uniquely identifies a patch, along with other information associated with that patch. You can target patch groups to individual endpoints or to user-defined groups of endpoints. Patch groups give you broad flexibility in how you test and distribute patches. For example, you can create a group of security patches that you want to distribute only to endpoints used by your sales force.

In Policy Manager, assigning patch groups to target machines is similar to assigning packages to target machines. You select one or more target machines, edit the policies, and add the patch groups that you want to assign.

The patch group icon  identifies patch groups. If you want to see which patches are members of a patch group, click the name of the patch group (which usually appears as a blue link).

---

Note: You can only see the patches that are members of a patch group if the patch group has been published to the master transmitter specified when configuring Patch Manager.

---

## What is a patch group assignment state?

The patch group assignment state determines whether or not a patch group should be assigned to target machines. There are two patch group assignment states that you can assign to patch groups in Policy Manager:

- **Assign**—Choose this state if you want to assign a patch group to the specified target machines.
- **Exclude**—Choose this state if you do not want to assign a patch group to the specified target machines. For example, if you want to assign a patch group to a group of machines except for one machine, you can choose the *assign* state for the entire group and the *exclude* state for the one machine that you want to exempt.

Do not confuse the patch group assignment state with the patch action. The patch action determines what occurs to each patch in the patch group on the target machine, such as install, stage, uninstall, or do nothing. You specify the action for each patch in the patch group using Patch Manager.

## What is Patch Service?

Patch Service is the agent on the endpoint that controls patch installations. On each endpoint, Patch Service determines what patches must be installed, the order of their installation, and the commands that should be run before, during, and after installation. In addition, Patch Service decides when to install the patches, reports on whether or not installation was successful, and manages patch-related reboots. To perform some of these tasks, Patch Service relies on two other agents—Policy Service and Scanner Service.

## Assigning patch groups to targets

In Policy Manager, assigning patch groups to target machines is similar to assigning packages to target machines. You select one or more target machines, edit the policies, and add the patch groups that you want to assign.

---

Note: Although the interface allows you to assign patch groups to all types of targets, you should assign patch groups only to machines, machine groups, and collections. You should also be careful when assigning patch groups to containers—make sure the containers contain machines and machine groups only. In most cases, it does not make sense to assign patch groups to targets other than machines. For example, if you assign patch groups to users, patch groups might get applied to each machine that the users log on to, even machines where the patch groups do not apply.

---

You create a policy by choosing a target and then by specifying the patch groups that the target should have. For each patch group, you can specify the following information:

- Assignment state
- Blackout exemption

Additionally, you can specify the following patch information in a policy:

- The reboot schedule for the target following a patch installation
- The blackout period for the target and exempting Patch Service from the blackout period
- The update schedule for Patch Service on the target

---

Note: If you are in multiple selection mode and are editing the policies for multiple targets at one time, you can only edit the package and patch group information, such as states and blackout exemptions. The package and patch group information that you save apply to all targets that you specified. The tabs for editing other information, such as the blackout schedule and the Policy Service update schedule, only appear when you are editing the policy for a single target.

---

The following procedure describes how you create or edit a policy starting from the Target View page. You can also create or edit a policy from the following pages:

- Target Details
- Package View
- Package Details

If you use those other pages to create or edit a policy, the first step in the following procedure is different, but the rest of the steps should be the same.

## ► To assign patch groups to target machines

- 1 On Policy Manager's Target View page, locate and select the target for which you want to create or edit a policy and then click Edit.

The Edit Policy page appears, and, in the Packages tab, shows you any packages assigned to the target. Because this procedure focuses on patches and patch groups, it does not include information about packages. For more information about assigning packages to targets, see “General directions for creating and editing policies” on page 209.

- 2 Click the Patches tab.

The Patch Groups tab appears and shows any patch groups assigned to the target. It shows both patch groups that are directly assigned to the target and patch groups that are indirectly assigned as a result of the target being a member of another group or container.

- 3 To add or remove patch groups from the policy:

- a Click **Edit Patch Group List**.

The left side of the page displays the patch groups that have been published to the transmitter. The patch groups are stored in the PatchManagement/PatchGroups folder on the transmitter.

- b To add a patch group to the policy, select a patch group on the left side of the page so that it appears on the right side of the page.

When you select a folder containing one or more patch groups from the Transmitter, Policy Manager displays a pop-up dialog asking you if you want to add all of the patch groups under that folder to the Patch Group List.

- c If you want to see which patches are members of a patch group, click the name of the patch group (which usually appears as a blue link) under the Patch Group List.

**Note:** You can only see the patches that are members of a patch group if the patch group has been published to the master transmitter specified when configuring Patch Manager. Also, you cannot see the members of the patch group if you assign the patch group to a target using a URL different from the one you used when you published the patch group. For example, you used the host name when publishing the patch group, but you used the IP address when assigning the patch group to a target.

- d To remove a patch group from the policy, select it on the right side of the page and then click Remove. See “Removing patch groups from a policy” on page 301.
- e When you are finished adding and removing patch groups, click OK.
- 4 On the Patch Groups tab, specify assignment states for the patch groups that you added to the policy. See “What is a patch group assignment state?” on page 297.
- 5 If you want to exempt patch groups from the blackout period, on the Patch Groups tab, select the check boxes for the corresponding patch groups. You must also exempt Policy Service and Patch Service from the blackout period. For more information about blackout periods, see “Setting a blackout period for a target” on page 195 and “Exempting Patch Service from the blackout period” on page 305.
- 6 If you want to wake powered-down machines for patch updates, click the WOW deployment options as described in “Creating WoW deployments” on page 219.
- 7 If you want to simulate the installation of the patch groups assigned to the target machine, click Run Simulation. See “Simulating the installation of patches” on page 302.

**Note:** To simulate patch installation, the target that you select for patch groups should be a single machine.

- 8 Click the Patch Service Schedule tab to set a schedule for updating Patch Service on a target.

During an update, Patch Service scans the target for the list of required and already installed patches. It also installs a set of patches, taking into account the dependencies of each patch. You can specify that updates take place on a recurring schedule. See “Overriding the Patch Service update schedule for target machines” on page 304 and “Exempting Patch Service from the blackout period” on page 305.

- 9 If you are creating or editing the policy for a single target, you can also complete the following tasks:
  - “Setting a blackout period for a target” on page 195
  - “Specifying the Policy Service schedule for a target” on page 270
  - “Setting tuner and package properties for a target” on page 275
  - “Specifying transmitter permissions for a target” on page 308
  - “Specifying the profile for a target” on page 310

- 10 When you are finished creating or editing the policy, click Preview.

The Preview page allows you review changes to the policy before you save them and apply them to the target. See “Previewing and saving policy changes” on page 211.

- 11 After reviewing your changes to the policy, click Save to confirm your changes and save them.

The next time that Policy Service updates and runs on the endpoint, it downloads and applies the policy that you created or edited.

## Removing patch groups from a policy

When you remove a patch group from a policy, the patch group is removed from the target the next time that Policy Service updates. However, removing the patch group does not cause the patches in the patch group to be uninstalled from the target.

► **To uninstall patches, you must use Patch Manager to perform the following steps**

- 1 Edit the patch group.
- 2 Specify the uninstall action for individual patches that you want to uninstall.
- 3 Republish the patch group.

The next time Patch Service runs, it updates the patch group and uninstalls the specified patches. See the see the *BMC Marimba Client Automation Patch Management User Guide*, available on the BMC Customer Support website.

# Simulating the installation of patches

You can only view a simulation of patch installation if the following service channels have run at least once on the target endpoint:

- Patch Service
- Scanner Service

When they run, these service channels collect information about the simulated installation of patches.

When you simulate patch installation, simulation is performed for all the patch groups that apply to the selected target, not only the ones in the policy. If patch groups were assigned to all endpoints, to a container, or to a group to which the selected target belongs, those patch groups would be included in the simulation.

---

Note: To simulate patch installation, the target that you select for patch groups should be a single machine.

---

## ► **To simulate the installation of patch groups**

From the Patch Groups tab, click Run Simulation.

The Simulate Patch Installation page appears and shows the order in which patches are installed on the target machine. In addition, the page shows the following information:

- A brief description of the patch
- The ID that Patch Manager assigned to the patch. This ID is different from the ID assigned by the vendor who supplied the patch.
- The action that you specified for the patch using Patch Manager
- Any notes that the vendor provided with the patch

You can view more details about these patches by clicking View Patch Simulation Logs. See “Viewing more details about the installation of patch groups” on page 303.

# Viewing more details about the installation of patch groups

## ► To view more details about patch group installation

On the Simulate Patch Installation page, click View Detailed Logs.

The Patch Simulation Logs page appears and shows more details about the patches. You can find out which patches are obsolete or which patches cannot be installed because of conflicts.

Example of patch simulation log entries:

Filtering patch groups

Processing stage MS04-004.Q832894.Q832894-6SP1.exe.\_1139234174  
checking MS04-004.Q832894.Q832894-6SP1.exe.\_1139234174 for  
obsolescence

Processing install MS04-007.Q828028.Windows2000-KB828028-x86-  
ENU.exe.\_1592190521  
checking MS04-007.Q828028.Windows2000-KB828028-x86-  
ENU.exe.\_1592190521 for obsolescence

# Overriding the Patch Service update schedule for target machines

This topic describes how you can use Policy Manager to override the Patch Service update schedule for specific target machines. By default, Patch Service follows the update schedule specified using Patch Manager. For more information about setting the default update schedule for Patch Service, see Patch Manager online help.

During an update, Patch Service scans the target for the list of required and already installed patches. It also installs a set of patches, taking into account the dependencies of each patch. You can specify that updates take place on a recurring schedule. If you do not set a schedule using Policy Manager, the schedule specified using Patch Manager is used. If Patch Service is not directly assigned to the target as part of the policy, Policy Manager assumes that Patch Service comes from the same transmitter as Patch Manager.

If you have set a blackout period for the target, you can exempt Patch Service from the blackout period. This exemption is useful if you anticipate emergency situations when you want to update or install patches during the blackout period. See “Exempting Patch Service from the blackout period” on page 305.

---

Note: You can set target-level settings, such as the update schedule for Patch Service, for one target at a time only. You cannot set target-level settings if you are in multiple-selection mode and are editing policies for multiple targets at one time. However, you can set target-level settings for all targets by selecting the All Endpoints target.

---

This procedure assumes you are on the Edit Policy page. See “General directions for creating and editing policies” on page 209.

## ► To override the default update schedule for Patch Service

- 1 On the Edit Policy page, click the Patches > Patch Service Schedule tab. The Patch Service Schedule page appears.
- 2 Choose Use Policy Manager to set the Patch Service update schedule.
- 3 In the Recurrence section, specify how often you want to update Patch Service:

- **Daily**—Patch Service can be updated every day, only on weekdays (Monday through Friday), or every 1 to 365 days, depending on the number you set.
  - **Weekly**—Patch Service can be updated every 1 to 52 weeks, depending on the number you set. During the week when an update is scheduled, an update occurs only on the days you select. You must select at least one day of the week.
  - **Monthly**—Patch Service can be updated every 1 to 12 months on the day you set. If you choose day 30 or 31 and the month when an update occurs doesn't have that day (such as February), the update occurs on the first day of the next month.
- 4 In the Update time section, specify the time of the day when you want to update the Patch Service:
- **Update at**—Specify one time of the day when updates occur.
  - **Update every**—Specify multiple times during the day when updates occur. You can set updates to happen every specified number of minutes or hours during the day, and you can limit updates to a single range of hours during the day. For example, you can have updates occur every hour between 9 AM and 5 PM or every 30 minutes from 5 PM to 9 AM.

When you are finished editing the policy, preview and save the policy. See “Previewing and saving policy changes” on page 211.

## Exempting Patch Service from the blackout period

If you have specified a blackout period when you want to prevent downloads and installations from occurring, you might want to use Policy Manager to exempt Patch Service.

**Exempting Patch Service from the blackout period.** When you set a blackout period, you can exempt Patch Service so that it can still update and run during the blackout period. This exemption is useful if you anticipate emergency situations when you want to install patches during the blackout period.

**Exempting patch groups from the blackout period.** On the Patch Groups tab of the Edit Policy page, you can indicate which patch groups you want to exempt from the blackout period. This exemption is useful if you anticipate emergency situations when you want to deploy patches as soon as possible, even though some endpoints are in a blackout period during that time. See “Assigning patch groups to targets” on page 297.

---

**Note:** Exempting patch groups from the blackout period only applies during the first time you deploy patch groups to target machines. Patch Service manages installation and any other actions that take place after patch groups have been deployed.

---

This procedure assumes you are on the Packages tab of the Edit Policy page. See “General directions for creating and editing policies” on page 209.

## ► To exempt Patch Service from the blackout period

- 1 On the Edit Policy page, click the Blackout Period tab.

The Blackout Period page appears.

**Note:** The options in step 2 only appear if you have specified a blackout period.

- 2 Choose one of the following options to specify Patch Service behavior:
  - Enable Patch Service to update during the blackout period and allow operations for patch groups that are exempt from the blackout period.
  - Disable Patch Service from updating during the blackout period. No operations will be performed for patch groups, even those that are exempt from the blackout period.

- 3 When you are finished creating or editing the policy, click Preview.

The Preview page allows you to review changes to the policy before you save them and apply them to the target. See “Previewing and saving policy changes” on page 211.

- 4 After reviewing your changes to the policy, click Save to confirm your changes and save them.

## Policy compliance for patch groups

Policy compliance is a feature that compares policies defined in Policy Manager (the packages or patch groups that an endpoint is supposed to have at this time) against the data collected by the Scanner Service from endpoints (the packages or patch groups that an endpoint actually has). You view policy compliance reports using Policy Manager.

Policy compliance for patch groups is reported in a similar manner to the policy compliance for packages, with the following exceptions:

- Expanding a patch group does not display any schedule information.
- Clicking on a patch group's name displays the patches that it contains and the action associated with each one.
- You can only view compliance for the entire patch group, not for individual patches. For example, if a patch group is subscribed successfully and one of the patches in the patch group failed to install properly, the patch group is reported as *non-compliant*. You can view machine details for the non-compliant patch groups to see the specific patches that failed to install.
- If a patch group includes one or more patches that require a reboot and the machine has not yet rebooted, installation is not considered complete and the patch group is considered non-compliant.

For more information about policy compliance, see “Viewing policy compliance” on page 317.

# Specifying transmitter permissions for a target

This section describes how to manage transmitter permissions. It includes the following topics:

- “What are transmitter permissions?” on page 308
- “Adding or editing transmitter permissions” on page 308
- “Deleting transmitter permissions” on page 309

## What are transmitter permissions?

*Transmitter permissions* refer to the user name and password that a transmitter can require before allowing target tuners to subscribe to and download packages. If the transmitter for your enterprise has been configured to require transmitter permissions, target tuners are prompted for a user name and password when subscribing to a package. This is a problem for unattended or silent installations, where you do not want the target users to have to supply user names and passwords for the transmitter.

Policy Manager allows you to specify user names and passwords for use with particular transmitters. These user names and passwords are then used when subscribing target tuners to packages on transmitters that require permissions.

For instructions on adding transmitter permissions, see “Adding or editing transmitter permissions” on page 308. For information about setting permissions on transmitters, see the online help for Transmitter Administrator.

## Adding or editing transmitter permissions

This topic describes how to add or edit permissions for a transmitter to Policy Manager. The following instructions assume that you know the user name and password required for the transmitter. If you do not have this information, contact the person responsible for administering your transmitters.

When multiple users edit the Transmitter Permissions page for the same target, the most recent edits to be saved are applied.

---

Note: You can set target-level settings, such as transmitter permissions, for one target at a time only. You cannot set target-level settings if you are in multiple-selection mode and are editing policies for multiple targets at one time. However, you can set target-level settings for all targets by selecting the All Endpoints target.

---

This procedure assumes you are on the Packages tab of the Edit Policy page. See “General directions for creating and editing policies” on page 209.

## ► To add or edit the permissions for a transmitter

- 1 On the Edit Policy page, click the Advance > Transmitter Permissions tab. The Transmitter Permissions page appears.
- 2 Do one of the following:
  - To add transmitter permissions, click Add Transmitter Permissions.
  - To edit existing transmitter permissions, select a transmitter and click Edit.

The Add/Edit Transmitter Permissions page appears.

- 3 For Host name:Port number, enter the host name and port number for the transmitter, such as `transmitter_name:80`.
- 4 Enter the user name and password that you want to use for the transmitter. Confirm the password by entering it again.
- 5 Click OK to save the transmitter permissions.

The added or edited transmitter permissions appear in the list.

When you are finished editing the policy, preview and save the policy. See “Previewing and saving policy changes” on page 211.

The user name and password you specified are used when targets subscribe or download a package from the transmitter.

## Deleting transmitter permissions

This topic describes how to delete permissions for a transmitter in Policy Manager. After you delete permissions for a transmitter in Policy Manager, target tuners cannot subscribe to packages on that transmitter unless the user name and password are supplied.

---

Note: You can set target-level settings, such as transmitter permissions, for one target at a time only. You cannot set target-level settings if you are in multiple-selection mode and are editing policies for multiple targets at one time. However, you can set target-level settings for all targets by selecting the All Endpoints target.

---

This procedure assumes you are on the Packages tab of the Edit Policy page. See “General directions for creating and editing policies” on page 209.

### ► To delete the permissions for a transmitter

- 1 On the Edit Policy page, click the Advance > Transmitter Permissions tab. The Transmitter Permissions page appears.
- 2 Select a transmitter and click Delete.

The deleted transmitter permissions no longer appear in the list.

When you are finished editing the policy, preview and save the policy. See “Previewing and saving policy changes” on page 211.

## Specifying the profile for a target

You can use Policy Manager to change the profile for a target. As part of the target’s policy, you can specify which profile the target should use.

This section includes the following topics:

- “What is a profile?” on page 310
- “Changing the profile for a target” on page 311

### What is a profile?

Profiles contain the configuration settings, rather than the product binaries, for BMC Marimba Client Automation infrastructure components. You can create several *profiles*—one for each type of component, including proxies, transmitters, managed nodes (endpoints), mirrors, and repeaters. Each profile is saved as a segment of the Infrastructure Service. If you change a profile setting, the next time an endpoint updates its Infrastructure Service channel, the endpoint gets the new profile settings.

When you select a profile and assign it to an endpoint using Policy Manager, the endpoint's `marimba.tuner.update.profile` tuner property is set to the name of the profile that you specified.

For more information about creating profiles and profile types, see the Help for setup and deployment.

## Changing the profile for a target

If you want to assign a different profile to a target (or assign a profile to a target that does not have one), you can make that change in the target's policy. For more information about creating and editing profiles, see the Help for setup and deployment.

Remember the following points when changing the profile on targets:

- You assign a new profile to targets by using the Profiles tab in Policy Manager or setting the `marimba.tuner.update.profile` tuner property through Tuner Administrator. This document provides instructions on using Policy Manager to assign a new profile. When you use Policy Manager to assign a profile, you can leverage collections. A collection is a list of users or machines that match certain criteria that you specify in an LDAP or database query. After the query runs, you can use Policy Manager to assign a profile to the resulting list (the collection).
- The Infrastructure Service channel runs on a schedule. The next time the Infrastructure Service starts on machines, it downloads and applies any available changes to profile settings and to the tuner binaries. Some changes are applied to machines after the tuner on those machines is restarted. For information on updating the tuner's binaries, see the *BMC Marimba Client Automation Installation Guide*, available on the BMC Customer Support website.

Before you assign the policy, make sure you perform the following tasks:

- If the profile does not yet exist, create the new profile that you want to assign to a target.
- If you want to assign the profile to a collection, define the collection that includes the set of machines to which you want to assign the new profile:
  - a Create a query and specify the criteria for the set of machines that you want. See "What is a collection?" on page 161.
  - b Run the query, either by setting a schedule for it or by running it manually.

After the query runs, the resulting list of machines (the collection) appears as a target in Policy Manager.

---

Note: You can set target-level settings, such as the profile, for one target at a time only. You cannot set target-level settings if you are in multiple-selection mode and are editing policies for multiple targets at one time. However, you can set target-level settings for all targets by selecting the All Endpoints target.

---

This procedure assumes you are on the Packages tab of the Edit Policy page. See “General directions for creating and editing policies” on page 209.

## ► To change the profile for a target

- 1 On the Edit Policy page for the target, click the Advanced > Profile tab.

If the target already has a profile assigned to it, the name of the profile and the URL for Infrastructure Service channel (the source for the profile) appears on this tab. If the target does not have a profile assigned to it, no profile name appears.

- 2 On the Profile tab, click Edit.

The Choose Profile page appears. If the target does not have a profile assigned to it, Policy Manager displays the profiles in the first Infrastructure Service channel that it finds on the transmitter from which the Policy Manager channel is subscribed. For example, if the Policy Manager channel is subscribed from `http://trans.company.com:5282/Marimba/Current/SubscriptionManager`, it displays the profiles in the first Infrastructure Service channel it finds located on the same transmitter, such as `http://trans.company.com:5282/Marimba/Current/InfrastructureService`.

- 3 If you want to see profiles from a different location, enter the transmitter’s URL (such as `http://trans.company.com:5282`) and, if access permission is set for the transmitter, enter the user name and password for subscribing to channels. Click Go.

A list of profiles appears on the left side of the page. These profiles are available with the Infrastructure Service on the transmitter you specified. Each profile is saved as a segment of the Infrastructure Service channel. Therefore, which profiles you see in the profiles list depend on the Infrastructure Service that you are viewing.

- 4 Click a profile so that it appears under Selected Profile on the right side of the page, and then click OK.

You return to the Profile tab, which displays the profile you have selected.

- 5 When you are finished editing the policy, preview and save the policy. See “Previewing and saving policy changes” on page 211.

Performing this procedure sets the following tuner property for the target machines:

```
marimba.tuner.update.profile
```

The value for this property is the name of the new profile segment that you want to assign to the target machines. The name of a profile segment has the following format: `.profile_<name>`

Example: `marimba.tuner.update.profile=.profile_ny_mirrors`

The next time Policy Service runs on a machine, this tuner property is set in the tuner’s `prefs.txt` file. The next time Infrastructure Service runs on the machine, and after the tuner is restarted, the new profile is applied to that machine.

## Provisioning unprovisioned Intel AMT vPro computers

Starting with version 8.2.00 of Policy Management, the tuner can provision computers not previously configured with Intel AMT vPro. Rather than through a remote authenticated console as in the case of vPro computers, administrators can provision un-provisioned vPro machines and configure management interfaces, power management settings, and network settings locally. The provisioning of un-provisioned vPro computers will be referred to as Host Based Provisioning (HBP) in the remainder of this document.

## Configuration

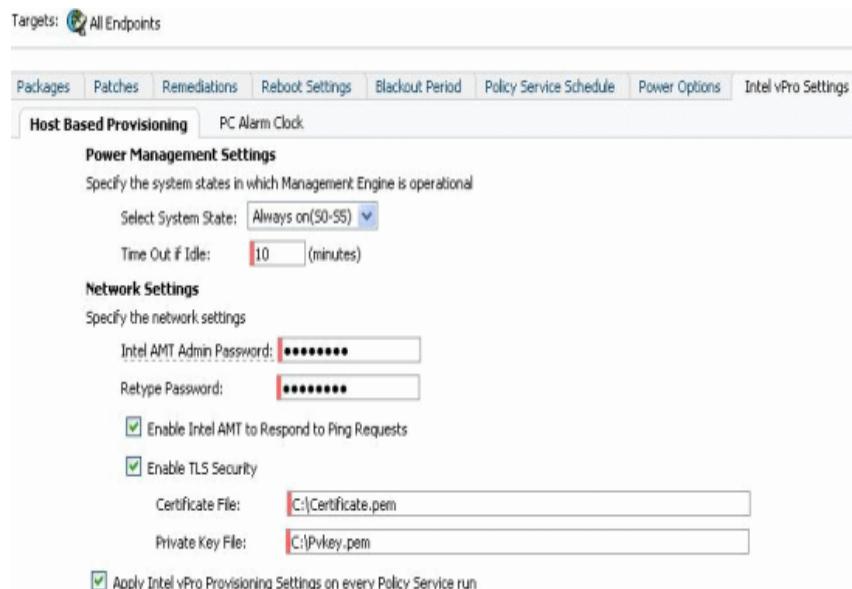
Intel vPro processors host a web service that allows an administrator to perform various operations, including provisioning of the machine.

To configure Intel vPro computers, select the **Host Based Provisioning** settings (displayed in Figure 14-1 and Figure 14-2) in the policy edit page in Policy Manager.

Figure 14-1: Policy Manager UI options for specifying provisioning settings



Figure 14-2: Additional Policy Manager UI options for specifying provisioning settings



When Policy Service is updated at the endpoint, it creates a host configuration XML based on the specified configuration in Policy Manager. The tuner parses this XML and uses the web service provided by Intel to trigger HBP.

After provisioning is successful, the HBP service automatically does the following:

- 1 Set the tuner property "marimba.tuner.amt.hoststatus" to "present-enabled". Inventory Service picks up the value of this property during an inventory scan; queries in Report Center based on this property will tell administrators whether or not the AMT machine was successfully provisioned.
- 2 The tuner will extract the vPro administrator password from the XML, and set it in the property **marimba.tuner.amt.password**. It will also extract information about whether or not the vPro host was configured for TLS mode, and set the host port number to 16992 (in case of non-TLS mode) or 16993 (in case of TLS mode) accordingly.

- 3 The tuner will delete the XML file from the tuner workspace after provisioning, to prevent inadvertent security leaks.
- 4 The tuner will determine whether or not the machine has been provisioned in Transport Layer Security (TLS) mode, and set the following properties:

```
TLS:  
runtime.amt.port=16993  
runtime.amt.secure=true  
Non-TLS:  
runtime.amt.port=16992  
runtime.amt.secure=false
```

Policy compliance is a feature that compares policies defined in Policy Manager (the packages that an endpoint is supposed to have at this time) against the data collected by the Scanner Service from endpoints (the packages that an endpoint actually has).

The following topics are provided:

- What is policy compliance? (page 318)
- Definition of compliance (page 318)
- Prerequisites for policy compliance (page 321)
- Viewing policy compliance for targets and packages (page 328)
- Compliance reports (page 338)
- OS migration compliance (page 340)

## What is policy compliance?

Policy compliance provides reports to administrators to help them answer the following types of questions:

- Are the distributed packages in the desired state?
- How many endpoints have the packages in the desired state?
- Which endpoints do not have the packages in the desired state?
- How many endpoints have not checked in for policy updates?

You view policy compliance reports using Policy Manager.

## Definition of compliance

The definition of compliance depends on the following elements:

- A package is compliant if the state on the endpoint matches the one specified in the policy.
- An endpoint is compliant if all the packages directly or indirectly assigned to it (in an active policy) are compliant.
- An aggregate target (group, container, domain, All Endpoints) is compliant if all its members are compliant.
- If all the packages on an endpoint are inactive (its activation schedule has not yet arrived), the endpoint will not be included in the calculation of percentages—that is, it will be added in the set of machines with the *Not checked in* status.

---

Note: User-based compliance is not supported. When the target is a user or user group, the Inventory database tracks the machine on which the user last logged in and determines policy compliance based on the status of that machine.

---

**Compliance by package.** A package is compliant if the following two conditions are true:

- The package is compliant in its active state.
- The most updated version of the package is available on the endpoint.

Packages usually refer to applications, files, and other information that someone in your organization has packaged into the channel format using a BMC Marimba Client Automation component called Application Packager. You might also have packages that were not packaged using Application Packager. These are usually BMC Marimba Client Automation components that are distributed using the channel format and are referred to as *channels*. The states collected by the Scanner Service might differ for packages and channels. For more information about these states, see “Overview of installation states” on page 215.

Packages fall into a policy compliance category based on the package state assigned in the policy and the package state at the endpoint as shown in “State mappings for policy compliance” on page 445. Persistence information is not collected by the Scanner Service, so policy compliance cannot distinguish between the install, install-persist, install-start, and install-start-persist states. Also, policy compliance does not show whether a channel is currently running, or if a particular channel has been made the primary channel (although that information is in the database).

Compliance is measured against the current state only. A package assigned to a target might have several states in its lifecycle:

Policy Manager reports compliance for all states in the package’s lifecycle. If an activation schedule is defined for a state, Policy Manager reports policy compliance for that state after the schedule is activated. Otherwise, the state is considered inactive and policy compliance for the package is reported for the previous state. If the activation schedule for the primary state has not yet arrived, the package is considered inactive. To determine the active states for a package, Policy Manager checks Policy Service’s last update time against the schedules in the policy for each state.

If an endpoint receives more than one policy for the same package, Policy Manager checks policy compliance for the package against the policy that won conflict resolution. See “Conflict resolution: states and schedules in policies” on page 244.

---

Note: When the target is a user or user group, the foreign-key relationship in the Inventory database is used to find the machine on which the user last logged in—policy compliance is based on that machine.

---

For the various states, the package is considered compliant if the following conditions are true:

Table 15-1: Conditions that make a package compliant

State	Conditions	
	Package state	Package update time
<b>Primary</b> The package maintains the primary state unless a secondary state and schedule is specified for the package.	The state of the package on the endpoint matches the primary state assigned to the package in the policy.	The last updated time for the package (as reported by the Scanner Service) is later than the last published time for the package (from the transmitter).
<b>Secondary (optional)</b> If you set a secondary state for a package, the package transitions to that state according to the activation schedule that you specify.	The state of the package on the endpoint matches the secondary state assigned to the package in the policy.	
<b>Update (optional)</b> The package only enters the update state if the package has been installed on the endpoint. Policy compliance takes into account the update state even if the update schedule was not set through a policy (that is, the schedule was set when the package was published).	The state of the package on the endpoint matches the primary (or secondary) state assigned to the package in the policy.	

**Compliance by target.** Although targets can refer to anything that you can assign a policy to (machines, users, groups, containers, and so on), machine targets are used when computing policy compliance. Machine targets are taken into account because policies are implemented by tuners and tuners are installed per machine (not per user or group).

A machine target is compliant only if all packages for that target are compliant.

Table 15-2: Conditions that make a target compliant

Compliance result	Condition
Compliant	All packages for the target are compliant.
Non-compliant	At least one package for the target is non-compliant.
Not checked in	At least one endpoint in the target has not checked in.

# Prerequisites for policy compliance

To view policy compliance information, you must set up the console server and the endpoints as described in the following sections:

- “Setting up the console server for policy compliance” on page 321
- “Setting up endpoints for policy compliance” on page 322

## Setting up the console server for policy compliance

**Install Policy Manager and Report Center.** Policy compliance requires the Policy Manager and Report Center channels on the console server. If you followed the instructions for installing and setting up the console server as described in the *BMC Marimba Client Automation Installation Guide*, available on the BMC Customer Support website, you should have the following channels installed and running:

- Common Management Services
- Policy Manager
- Report Center

---

**Important:** Policy compliance information is based on the tuner/transmitter component scan. If different scan schedules are set for the components on Report Center’s inventory plug-in configuration page and the tuner/transmitter component scan does not run, be aware that Policy Manager might not show the correct policy compliance information.

---

You should also make sure that you meet the system requirements described in the *BMC Marimba Client Automation Installation Notes* and *BMC Configuration Automation for Clients Installation Guide*, available on the BMC Customer Support website.

Make sure you have installed and configured Policy Manager as described in “Initial configuration” on page 27. Also, make sure you have installed and configured Report Center (including configuring the database and importing the query library) as described in the *BMC Marimba Client Automation Installation Guide* and the *BMC Marimba Client Automation Report Center Guide*. All the books mentioned are available on the BMC Customer Support website.

**Enable policy compliance.** By default, the collection of policy compliance data is disabled starting with version 6.0.2.1. If you want to view policy compliance reports, make sure you enable the collection of policy compliance data and set the options for policy compliance to suit the needs of your enterprise as described in “Configuring policy compliance settings” on page 155.

**Synchronize the directory service and database.** To compute compliance, you must first synchronize the directory service with the database using the console’s system settings (Console > System Settings > Data Source > LDAP-to-Database Synchronization Service). See the console’s system settings online help.

## Setting up endpoints for policy compliance

Policy compliance requires the Scanner Service and Policy Service channels on the endpoints (also called *managed nodes*). If you followed the instructions for creating profiles and installers for the managed nodes, as described in the *BMC Marimba Client Automation Installation Guide*, available on the BMC Customer Support website, you should have the following channels included in your managed nodes:

- Infrastructure Service
- Scanner Service
- Policy Service

## Best practices for policy compliance

Viewing policy compliance reports is a data- and time-intensive process that depends on the number of packages assigned to a target and the number of machines included in a target. To improve the performance when viewing policy compliance reports, consider the following best practices:

- Depending on the number of endpoints and packages in your enterprise, computing compliance reports for the first time might take a long time (up to several hours). Try to schedule the first time computation at off hours (for example, in the evenings), so that there is less impact on the database and console server.

- Be aware that the first time you view compliance, you might see the “Not Checked In” state for endpoints, especially if Scanner Service channels on the endpoints have not yet sent machine information to the database. The compliance information appears (no longer the “Not Checked In” state) after the next time the Scanner Service channel runs on the endpoints.
- Limit the number of endpoints in the target for which you want to view policy compliance reports. Depending on the number of packages assigned to the endpoints in the target, viewing policy compliance reports might take a long time. For example, you can usually view the policy compliance report for a target that includes around 100 endpoints with about 20 assigned packages within a reasonable amount of time. Increasing the number of endpoints in the target and the number of packages assigned to the endpoints correspondingly increases the amount of time it takes to display the policy compliance report.
- When recomputing compliance, be aware that the larger the number of endpoints and packages the target includes, the longer the calculation will take. Recomputing compliance for a large number of endpoints and packages is not recommended.
- If you frequently view the compliance reports for specific groups or containers, you might want to cache the compliance reports for those groups or containers on the Compliance Options page. You can also set a schedule for recomputing the compliance reports. The recommended schedule is after Scanner Service has run on the endpoints and sent information to the database. See “Configuring policy compliance settings” on page 155.
- To reduce load on the database and console server, schedule the computation of compliance reports during off-peak hours.
- When caching compliance information, be aware that the larger the number of endpoints and packages the target includes, the longer the caching this information will take. Caching compliance information for a large number of endpoints and packages is not recommended.
- Make sure that you meet the system requirements described in the *BMC Configuration Management System Requirements Guide*, available on the BMC Customer Support website. This document describes the hardware requirements for the machines running the console and the database, specifically if you want to use the policy compliance feature.

# How does policy compliance work?

## ► Compliance works in two stages

- 1 **Service stage.** In this stage, Scanner Service runs and starts Policy Service. Scanner Service starts Policy Service only when it performs the tuner/transmitter-specific scan (not during the system/hardware-specific scan or the application-specific scan). Policy Service calculates the compliance data based on the package state information collected by the Scanner Service and the latest config.xml that the endpoint received.
- 2 **Server stage.** In this stage, Policy Manager performs the compliance calculation. During this stage, compliance might be downgraded for two reasons:
  - The policy has changed in the directory service, but the endpoint has not received the policy or the latest information about the endpoint is not yet in the database. Compliance is set to the *not checked in* state.
  - A newer version of the package is published to the transmitter, but the endpoint has not updated to the new version. Compliance is set to the *not compliant* state.

## Components of policy compliance

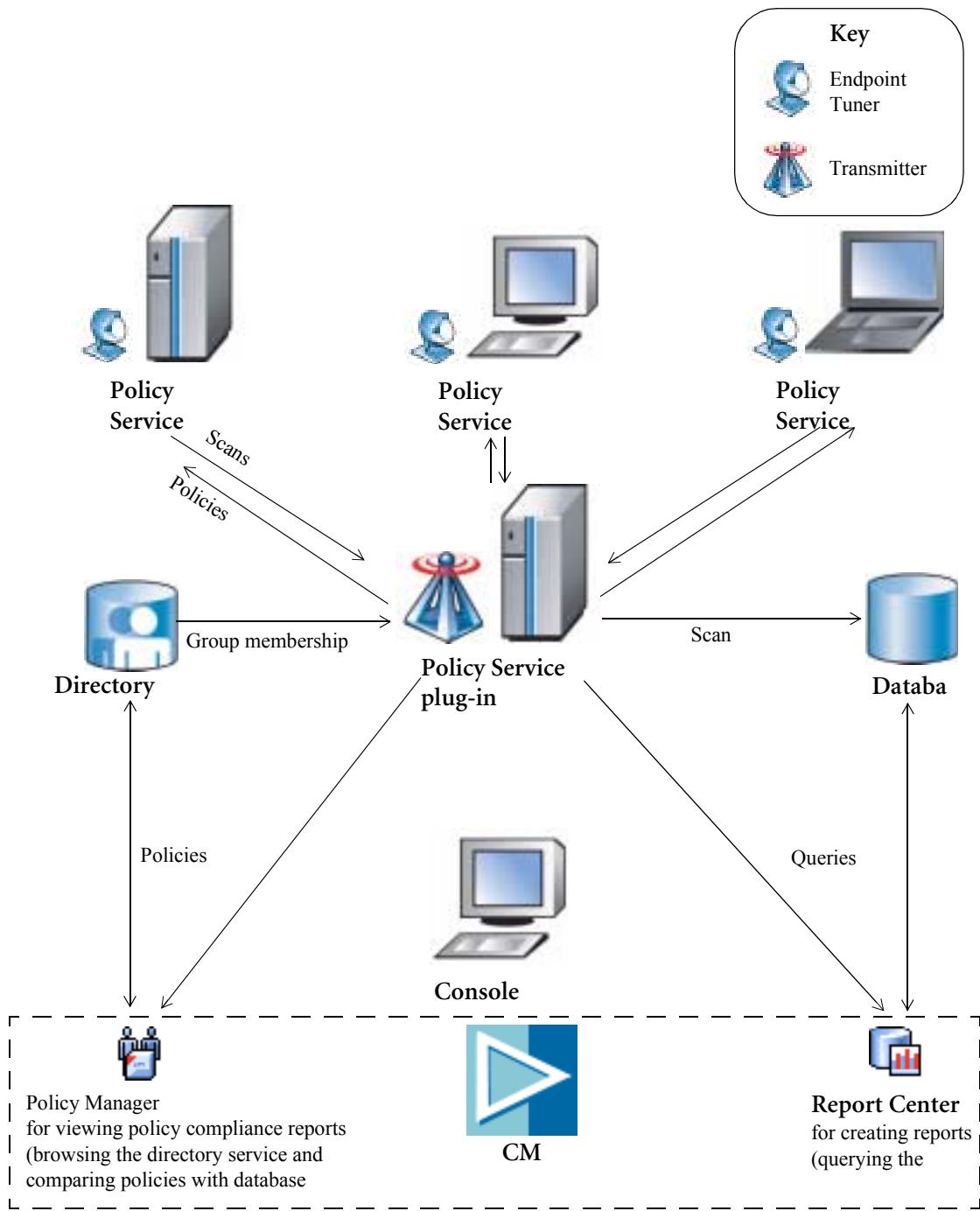
The following components are part of policy compliance:

- **Report Center** provides the browser-based interface you use to perform queries, create reports, and drill down on information for specific machines. You also use this component to specify configuration settings for inventory and centralized logging. You can also export the results of policy compliance reports to Report Center so that you can drill down to specific machines.
- **Policy Manager** provides the browser-based interface you use to view policy compliance reports. You can use Policy Manager to browse and select targets to view policy compliance. This component compares the data gathered from the Inventory database with the policies defined in Policy Manager to provide policy compliance reports.
- **Policy Service plug-in** provides group membership information for targets.

- **Policy Service** parses and writes additional group membership information for targets. It also writes out the policy compliance information into the Inventory tree.
- **Scanner Service** scans the endpoints and gathers information about packages, including the status, from the endpoints. Later, this information will be compared to the status for those packages described in the policies.
- **Inventory plug-in** inserts information collected by the Scanner Service into the Inventory database.

The following diagram illustrates how the components interact in a policy compliance system:

Figure 15-1: Components of a policy compliance system



## Limitations

Policy compliance has the following limitations:

- Policy compliance reports endpoints that do not have Policy Service or Scanner Service installed as not checked-in. It does the same for targets that do not have a policy, or endpoints that do not have inventory data in the database.
- Policy compliance is based on packages and their states only. The values of tuner and channel properties are not included.
- Policy compliance results are a snapshot in time. They might not be accurate at all times because of the synchronization of system components. For example, a race condition might occur in which Scanner Service and Policy Service run at the same time, resulting in the following scenario:
  - a A target does not have any packages assigned.
  - b An administrator creates a policy and assigns two packages to the target.
  - c Scanner Service starts scanning the machine and sending data to the database. Scanner Service starts Policy Service also.
  - d Policy Service contacts the plug-in and gets the policy for the target. Two things might occur at the same time:
    - Policy Service's helper thread starts installing the packages assigned in the policy.
    - Policy Service's main thread computes policy compliance.
  - e Because the packages had not yet finished installing, the policy compliance results show that the target is not compliant for these two packages.

**Note:** The next time Policy Service updates (the default is every 90 minutes), the policy compliance information will be accurate.

- Compliance for the repair state is not reported.
- Any group membership changes in the directory service will only reflect in the compliance report after you run LDAP sync in CMS.
- If the system removes a member from a group, policy compliance will no longer consider that member after you run LDAP sync in CMS.
- In a multi-domain Active Directory architecture, the calculation of policy compliance results include members from other domains, such as universal groups containing machines from another domain.

- When the target is a user or user group, the foreign-key relationship in the Inventory database is used to find the machine on which the user last logged in—policy compliance is based on that machine.
- **Multiple master transmitters.** Compliance queries and reports cannot differentiate among packages you publish at different times across multiple master transmitters in your environment. For example, say transmitter A and transmitter B have packages with matching URLs, and the packages contain different versions of the same software. In this scenario, Policy Manager can't calculate compliance accurately for either package.
- **Multiple channel segments.** Compliance queries and reports do not differentiate among multiple channel segments that you publish at different times. For example, say you publish segment A for Linux at 9:00 AM and segment B for Windows at 10:00 AM. In this scenario, Policy Manager can't calculate compliance accurately.

## Viewing policy compliance for targets and packages

You can view the policy compliance information for targets and packages using any of the following tabs:

- The Target View tab
- The Target Compliance tab
- The Package Compliance tab

If your target contains members (such as groups, containers, and organizational units), you can view three types of compliance information on the Target Compliance tab:

- The **Policy Compliance** tab shows compliance information for policies directly and indirectly assigned to targets. Queries you run here consider the latest LDAP-synced policies. For more information about LDAP synchronization, see “Setting up the console server for policy compliance” on page 321.
- The **Overall Compliance** tab shows compliance information for all the machines in the target, regardless of the policies from which a given machine receives its packages.

- The Power Options Compliance tab shows compliance information about Windows Power Options settings for a selected endpoint or target group.

## Overall compliance queries and compliance reports

Overall compliance is a quick and useful way to track deployment progress. Overall compliance queries run faster than the more robust compliance reports, in part because they do not consider recently-published LDAP-synced policies. Information about deployment progress trickles in as Scanner Service scans endpoints and sends reports to the database.

Overall compliance queries help you to:

- Periodically track the progress of a deployed policy
- Focus on inventory scan performance
- Find out how many machines are checking in

Use compliance reports when you require synchronization with the latest policy information. For more information about the compliance report, see “Compliance reports” on page 338.

## Locating targets and running queries

### ► To locate a target and run a compliance query

- 1 From the Target View tab in the left pane under Targets, select the target for which you want to view compliance information.

You can also begin on the Target Compliance tab.

The members of the target group appear in the right pane.

Figure 15-2: The Target View tab

The screenshot shows the 'Target View' tab in the BMC Configuration Automation for Clients Policy Manager. The left pane, titled 'Targets', contains a navigation tree starting from 'Home > marimba.com'. It includes search filters for 'Basic Search' and 'Advanced Search', and a selection mode switch between 'Single' and 'Multiple'. The right pane, titled 'Computers', displays a list of packages in 'Basic View'. The table has columns: Type, Packages, Primary State, Secondary State, and Directly Assigned To. There are 20 entries, each labeled 'File Package' followed by a number (01 to 20). All packages have 'Advertise' as the primary state and 'N/A' as the secondary state. Under 'Directly Assigned To', all entries point to 'Computers'. At the top of the right pane, there are buttons for 'Edit', 'Copy', 'Update', 'Delete', and 'Compliance'.

Type	Packages	Primary State	Secondary State	Directly Assigned To
File Package 01		Advertise	N/A	Computers
File Package 04		Advertise	N/A	Computers
File Package 10		Advertise	N/A	Computers
File Package 11		Advertise	N/A	Computers
File Package 12		Advertise	N/A	Computers
File Package 13		Advertise	N/A	Computers
File Package 14		Advertise	N/A	Computers
File Package 15		Advertise	N/A	Computers
File Package 16		Advertise	N/A	Computers
File Package 17		Advertise	N/A	Computers
File Package 18		Advertise	N/A	Computers
File Package 19		Advertise	N/A	Computers
File Package 2		Advertise	N/A	Computers
File Package 20		Advertise	N/A	Computers

- From the Target View tab, click the Compliance button on the right to open the Target Compliance tab and display compliance information.

The target name appears on the Target Compliance tab at the top of the right pane. If your target contains members, you can view policy compliance, overall compliance, or power options compliance. The Policy Compliance tab is active by default.

Figure 15-3: The Target Compliance tab with the Policy Compliance tab active

Package	State	Compliance	Directly Assigned To
File Package 01	Advertise	Not yet calculated	computers
File Package 04	Advertise	Not yet calculated	computers
File Package 10	Advertise	Not yet calculated	computers
File Package 12	Advertise	Not yet calculated	computers
File Package 13	Advertise	Not yet calculated	computers
File Package 14	Advertise	Not yet calculated	computers
File Package 15	Advertise	Not yet calculated	computers
File Package 16	Advertise	Not yet calculated	computers
File Package 17	Advertise	Not yet calculated	computers

## Viewing policy compliance for targets

You can view compliance information for targets that include members or targets that are individual machines. The procedures that follow apply to any target.

► **To view compliance information for all assigned packages in a selected target**

- 1 Locate a target on the Target Compliance tab.
- a In the left pane, browse and select the target for which you want to view compliance information. The target name appears at the top of the right pane, and the Policy Compliance tab is active.
  - b Select all packages by selecting the check box in the header row, and click Calculate.

Figure 15-4: Calculating compliance for all packages in a target

The screenshot shows the BMC Configuration Automation for Clients interface. At the top, there's a navigation bar with links for Applications, Status, Toggle Info Text, About, Logout, and Help. Below the navigation bar, there are tabs: Target View, Package View, Target Compliance (which is selected), Package Compliance, Compliance Reports, and Configuration.

The main content area is titled "Compliance: Target View". It contains a message box stating: "From this page, you can view the compliance information for a selected target. Use the left side of the page to find the target, and then click the target name to view the compliance information on the right side of the page." On the left, there's a tree view under "Targets" showing categories like Home, Computers, ForeignSecurityPrincipals, Roles, Users, Marimba, and BMC Software. On the right, there's a table titled "Computers" with tabs for Policy Compliance, Overall Compliance, and Power Options Compliance. The table lists "File Package 01" through "File Package 17" with their respective states: Advertise, and compliance percentages: Not yet calculated. A "Calculate" button is visible at the top of the table.

► To view compliance information for specific packages you have assigned to the selected target

- 1 On the Policy Compliance tab, select the check box for each individual package that you want to include in your query.
- 2 Click Calculate.

Policy Manager updates the selected packages with status information about your query. The following table displays available states.

Query state	Description
Not calculated	No compliance information is available.
Calculating	Policy Manager has not yet added your query to the compliance queue.
In-Queue	The query is in the compliance queue.
Processing	The query is executing on the compliance server.
Error	The query failed to execute.

► To view information about endpoints that are compliant, non-compliant, or not checked-in

- 1 Click the pound sign (#) in the Compliance column header to see the exact number of endpoints for each status.
- 2 Click the percent sign (%) in the Compliance column header to see the number of endpoints as a percentage of the total number of endpoints in the target.

The numbers or percentages appear in colors that indicate the status:

- Green for compliant.
- Red for non-compliant.
- Blue for not checked-in.

- 3 Click the displayed numbers or percentages in a package row to view a list of machines related to packages with a given status.

Figure 15-5: Target compliance query results by policy

The screenshot shows the 'Target Compliance' section of the BMC Configuration Automation for Clients Policy Manager. The top navigation bar includes links for Applications, Status, Toggle Info Text, About, Logout, and Help. Below the navigation is a breadcrumb trail: Home > marimba.com > Computers. The main content area is titled 'Compliance: Target View' and contains two main sections: 'Targets' and 'Computers'.

**Targets:** A list of targets with a dropdown menu showing 'previous 1-6 of 6 next'. The targets listed are: Computers, ForeignSecurityPrincipals, Roles, Users, Marimba, and BMC Software.

**Computers:** A table showing policy compliance data for 21 computers. The columns are: Package, State, Compliance (%), and Directly Assigned To. The table lists 16 packages, all of which are Advertise state. The compliance percentages are 75% for 14 packages and 25% for 2 packages. The 'Directly Assigned To' column shows that all packages are assigned to 'computers'.

Package	State	Compliance (%)	Directly Assigned To
File Package 01	Advertise	75% 0% 25%	computers
File Package 04	Advertise	75% 0% 25%	computers
File Package 10	Advertise	75% 0% 25%	computers
File package 11	Advertise	75% 0% 25%	computers
File Package 12	Advertise	75% 0% 25%	computers
File Package 13	Advertise	75% 0% 25%	computers
File Package 14	Advertise	75% 0% 25%	computers
File Package 15	Advertise	75% 0% 25%	computers
File Package 16	Advertise	75% 0% 25%	computers

► **To view compliance information for a single machine**

- 1 Select a target that is an individual machine in the Targets pane.

The Machine Compliance tab is active by default and provides compliance information for the selected machine.

- 2 Click the Power Options Compliance tab to display the setting and compliance status for each power option.

► **To view compliance information for all machines in the selected target**

- 1 Click the Overall Compliance tab. The green, red, and blue compliance status tabs represent the machines in the selected target that are compliant, non-compliant, and not checked-in, respectively.

- 2 Click the green, red, or blue compliance status tabs to view the list of machines related to each status.

- The **Green** tab displays all the machines in this target that are compliant with all policies that you have assigned to them, both directly and indirectly.
- The **Red** tab displays all the machines in this target that are not compliant with all policies you have assigned to them, both directly and indirectly. The list displays one cause of non-compliance for each machine.

**Note:** The list displays only one cause of non-compliance for performance reasons. Click the machine name to view more information about a non-compliant machine.

- The **Blue** tab displays all the machines in this target that have not checked in since you made any policy changes that affect the machines. For each machine, the list displays the date when the machine last reported status.

---

Note: When calculating compliance status, Policy Manager only considers machines that have checked in during the time period you set in Compliance Options on the Configuration tab. Otherwise, Policy Manager considers machines to be not checked-in. See “Configuring policy compliance settings” on page 155.

---

Figure 15-6: A list of compliant machines on the Overall Compliance tab

► To view Windows Power Options compliance information for all machines in the selected target

- 1 Click the Power Options Compliance tab. The green, red, and blue compliance status tabs represent the machines in the selected target that are compliant, non-compliant, and not checked-in, respectively as shown in Figure 15-7 on page 336.
- 2 Click the green, red, or blue compliance status tabs to view the list of machines related to each status.

Power Options Compliance has the following prerequisites:

- On endpoint computers, the following components must be version 8.0 or later:
  - Tuner
  - Inventory Service
  - Subscription Service (Policy Service)

- On the CMS console computer, the following components must be version 8.0 or later:
  - CMS
  - Report Center
  - Schema Manager
  - Subscription Manager (Policy Manager)
- After upgrading to version 8.0, you must clear the LDAP sync data and perform a one-time initial sync.

Figure 15-7: A list of machines on the Windows Power Options Compliance tab

The screenshot shows a web-based interface for policy management. At the top, there's a navigation bar with tabs for 'Policy Compliance', 'Overall Compliance', and 'Power Options Compliance'. The 'Power Options Compliance' tab is active. In the center, a message states: 'All machines that are part of this group have the following power options compliance data. The compliance for each machine is calculated based on all of its inherited and directly assigned policies.' Below this, a progress bar indicates 'Power Properties Compliance' at 100%, with 0% and 0% also shown. A red header labeled 'Failed Machines (100%)' is followed by a table with the following data:

Machine	Cause of Failure	Desired Value	Actual Value
css-iq3t9rucuf8	Monitor idle time	120	240
css-iq3t9rucuf8	Hard disk idle time	180	240
css-iq3t9rucuf8	Standby idle time	180	0
css-iq3t9rucuf8	Hibernate idle time	240	N/A
css-iq3t9rucuf8	Enable hibernate	1	0

## ► Reviewing query details

To review the setup and results of the query in Report Center, click the Show Report Center Query button.

Figure 15-8: Query details in Report Center

The screenshot shows the 'Report Center' interface. At the top, there's a navigation bar with 'Applications ▾', 'Status', 'Toggle Info Text', 'About', 'Logout', and 'Help'. Below the navigation bar, there are two tabs: 'Queries' (selected) and 'Configuration'. The main content area displays a query titled '/Query Library/Policy Compliance/OverallNonCompliantMachineByInventory'. A search input field contains the placeholder text 'Enter the search criteria requested and click View Results.' Below the search field, there's a box containing 'Created by: rpl\_customer-support@bmc.com on Jul 22, 2005 2:23:40 PM GMT-08:00', 'Description:', and 'Query Access: Filtered. If you make a copy of this query using "Save" or "Save As" option it will be saved as Unfiltered.' At the bottom of the main content area, there are three tabs: 'Query Form' (selected), 'Results', and 'Setup'. A note below these tabs says: 'Following is a list of the search criteria this form uses to create a query. In text boxes, you can use an asterisk (\*) as a wildcard for one or more characters. Use a period (.) as a wildcard for only one character.'

### ► Generating a report

To generate an overall compliance report for the target, click the Generate Compliance Report button. Confirm the message window, and the Compliance Reports tab appears.

---

Note: Reports take a longer time to run. When you click the Generate Compliance Report button, the start time displays immediately after the confirmation message. The end time displays only after the report has completed execution.

---

See “Compliance reports” on page 338.

### Viewing policy compliance for packages

You can view policy compliance information for a package to find out:

- What targets should have this package?
- Do those targets have this package?

Viewing package compliance is similar to viewing target compliance, except you use the Package Compliance tab to query targets related to a package, instead of the Target Compliance tab to query packages related to a target.

The Search region enables you to find and filter the list of packages you want to include in your query by searching on the package name or date of publication. The search function supports the asterisk (\*) and percent sign (%) as wildcards. See “To view compliance information for all assigned packages in a selected target” on page 331.

Figure 15-9: The package compliance tab and the search region

The screenshot shows the BMC Configuration Automation for Clients Policy Manager web interface. At the top, there's a navigation bar with links for Applications, Status, Toggle Info Text, About, Logout, and Help. Below the navigation bar, there's a horizontal menu with tabs: Target View, Package View, Target Compliance, **Package Compliance**, Compliance Reports, and Configuration. The Package Compliance tab is currently active. On the left side, there's a search interface for packages. It includes a search input field with placeholder text "Package Name: %", a dropdown for "Published-after:" with "M/d/yy" selected, and a date input field with "Example Date: 2/10/05". There are "Go" and "Reset" buttons. Below the search interface is a list of packages, each with a thumbnail icon, a link labeled "File Package 01", and a number from 1 to 19. On the right side, there's a detailed view for "File Package 01". This view includes a "Calculate" button, a table header with columns for Targets, Primary State, and Compliance (%), and a single row for "computers" which is listed under "Advertise". A progress bar indicates 75% completion. Navigation buttons for "previous" and "next" are at the top right of the detailed view area.

## Compliance reports

In addition to running compliance queries on the fly based on targets or packages, you can run reports that calculate overall target compliance, and save them for review later. You start reports on the Overall Compliance tab on the Target Compliance page by clicking the Generate Compliance Report button, and store them for review or removal on the Compliance Reports tab. (For more information on running target compliance queries, see “To view compliance information for all assigned packages in a selected target” on page 331.)

Figure 15-10: The Compliance Reports tab, displaying a saved compliance report

The screenshot shows the BMC Configuration Automation for Clients Policy Manager interface. At the top, there's a navigation bar with tabs: Target View, Package View, Target Compliance, Package Compliance, **Compliance Reports**, and Configuration. To the right of the tabs are links for Applications, Status, Toggle Info Text, About, Logout, and Help, along with the BMC Software logo.

The main content area is titled "Compliance Reports". It contains a message: "From this page, you can view the list of cached policy compliance reports. These compliance reports were generated and cached when you (or another user) calculated the compliance information for a target." Below this is a table with the following data:

<input type="checkbox"/>	Target	Overall Compliance	Start Time	End Time
<input type="checkbox"/>	computers	<div style="width: 100%;">3.00</div>	Feb 10, 2006 5:25:41 AM	

At the bottom right of the table area, there are links for "previous", "1-1 of 1", and "next".

You can also use the Compliance Reports tab to delete saved compliance reports.

#### ► To view or delete compliance reports

- 1 In Policy Manager, click the Compliance Reports tab.

The saved compliance reports appear. For each report, you can view additional information:

- Hold the mouse pointer over the compliant, non-compliant, or not checked-in percentages to view the number of machines for each status.
- Click the compliant, non-compliant, or not checked-in percentages to view the list of machines for each status.

- 2 To delete reports, select the reports that you want to remove and click Delete.

# OS migration compliance

You can also view the status of migration to the Windows 7 OS. Click the OS Deployment Compliance tab on the Target Compliance page to view the endpoints for which the migration was successful, failed, or still in progress. Figure 15-11 shows the OS Deployment Compliance tab.

Figure 15-11: The OS Deployment Compliance tab

The screenshot shows the 'OS Deployment Compliance' tab selected in a navigation bar. Below it, a message states: 'All machines that are part of this group have the following OS Deployment compliance data. The compliance for each machine is based on the assigned OS template.' A progress bar indicates 'OS Deployment Compliance: 0% 0% 100%' with three colored segments: blue, red, and green. The assigned OS Template Name is listed as 'WINDOW\_7'. A table titled 'Pending Machines (100%)' lists three machines: 'hdc0005215', 'hdc0005741', and 'marimba107'. The table includes columns for 'Machine' and 'Updated Time'.

Machine	Updated Time
hdc0005215	No Entry
hdc0005741	No Entry
marimba107	0001-01-01T00:00:00

The CMS Console does not provide troubleshooting options. To debug OS migration failures, you must access the Empirum console for detailed logs about individual endpoints. For details, see the *BMC Marimba Client Automation OS Migration with Matrix42 User Guide*.

Chapter

# 16 Integrating with Deployment Manager

This chapter describes how you can use Policy Manager with the BMC Deployment Manager application to immediately update policies on target machines.

The following topics are provided:

- Prerequisites for integration with Deployment Manager (page 342)
- Enabling and disabling immediate policy updates (page 345)
- Performing an immediate policy update (page 345)
- Monitoring and viewing the status of a policy update (page 347)
- How does the immediate policy update work? (page 349)
- Blackout periods and immediate policy updates (page 350)

# Prerequisites for integration with Deployment Manager

For information about installing the components mentioned in this section, see the *BMC Marimba Client Automation Installation Guide*. For information about configuring Deployment Manager and Deployment Service, see the *Server Management User Guide*; for information about configuring Report Center and Scanner Service, see the *BMC Marimba Client Automation Report Center User Guide*. These documents are also available on the BMC Customer Support website.

**Server side.** You must have the following components installed and configured on the same machine as Policy Manager:

- Report Center
- Deployment Manager

---

Note: BMC recommends running Deployment Manager on a different machine from Policy Manager.

---

For a new installation of Deployment Manager, you must log in and accept the license before you can use it with Policy Manager. You also need to configure the Deployment Manager settings for the console (System Settings > General > Deployment Manager Integration Settings) so that Policy Manager can communicate properly with Deployment Manager. You need the following information:

- The host name of the machine on which Deployment Manager runs. This machine can be the same one where Policy Manager is running, or it can be a different machine on which Deployment Manager is running.
- The port number used by the tuner running the Deployment Manager channel. The default is 7717.
- The user name and password for accessing Deployment Manager. Deployment Manager must have been configured to give access to this user.
- The remote administration user name and password for the endpoint tuners to which Deployment Manager connects. The user name and password were assigned to the endpoint tuners either when creating the tuner installers or when connecting to the tuners using Tuner Administrator. All the endpoint tuners must use the same remote administration user name and password.

There are also optional settings that you can configure on the Deployment Manager Integration Settings page. In addition, you can log in to Deployment Manager to change its default settings, including the status port—the port used for sending status messages back from target servers to Deployment Manager. For more information about Deployment Manager settings, see the Deployment Manager online help.

Before you can perform updates, you also need to enable the immediate policy update feature; it is disabled by default. See “Enabling and disabling immediate policy updates” on page 345.

**Client side.** You must have the following components installed on the endpoints along with Policy Service:

- Deployment Service
- Scanner Service

---

Note: If you want to use the immediate policy update feature with collections, make sure that Scanner Service runs on the endpoints at least once after you create the collections. Running Scanner Service again on these endpoints makes sure that the membership information required by the immediate policy update feature is available in the database.

---

## Assumptions and limitations

In addition to the prerequisites mentioned in “Prerequisites for integration with Deployment Manager” on page 342, be aware of the following assumptions and limitations when performing immediate policy updates.

### Scalability considerations

- This feature is suited for systems where the total number of endpoints number in the hundreds (such as server endpoints in a data center) as opposed to desktop endpoints, which could number in tens of thousands.
- If an endpoint gets updates from a repeater, the applications and patches that you want the endpoint to get must be updated on that repeater.
- In an Active Directory environment, the policy does not update properly if the policy has not been replicated to the global catalog of the Policy Service plug-in.

- All endpoints need to communicate with the same Deployment Manager. There is no distributed sharing of the load across different Deployment Managers at this time.
- Because Policy Service and Patch Service are executed sequentially, and Deployment Manager waits for the completion of both channels, the time it takes for an immediate policy update could be long compared to asynchronous execution of Policy Service and Patch Service.

## Setup considerations

- All the endpoints that you use as targets for a policy update must use the same tuner remote administrator user name and password.
- You cannot perform immediate policy updates if the endpoints are separated from Deployment Manager by a firewall. A different Deployment Manager with Policy Manager must be set up in each firewall boundary, unless you have open ports through the firewall for communication between Deployment Manager and Deployment Service.

For information about configuring BMC Marimba Client Automation components and firewalls, see the *BMC Marimba Client Automation Installation Guide*, and the *BMC Marimba Client Automation CMS and Tuner User Guide*, available on the BMC Customer Support website.

- Scanner Service must run on the endpoint at least once before you can perform an immediate policy update on it. Policy Manager obtains group information and endpoint tuner information (such as the port number) from the database used for Inventory. For collections, Policy Service must also have run on the endpoint at least once.

## Availability and permissions

- At this time, only one immediate policy update can take place at a time. If an update is currently running, the immediate policy update feature is not available to you or other administrators. See “Monitoring and viewing the status of a policy update” on page 347.
- At this time, the immediate policy update feature is only available from the browser-based interface (not from the command line).
- You can perform immediate policy updates in single selection mode only. The immediate policy update feature is not available in multiple selection mode.

- To perform immediate policy updates, you must log in as a primary administrator or standard administrator.
- To perform immediate policy updates for a target, you must have policy read and write permissions to the target. See “Setting up access control lists” on page 153.

## Enabling and disabling immediate policy updates

Before you can perform immediate policy updates, you must enable this feature as described in this section. Remember that to use this feature, you must also configure the Deployment Manager integration settings (in System Settings).

To determine whether the immediate policy update feature is on or off, place your mouse pointer over the Status button in the upper-right corner of the console window.

### ► **To enable or disable immediate policy updates**

- 1 Click the Configuration tab.
- 2 On the Configuration page, click the Advanced Options link.  
The Advanced Options page appears.
- 3 To enable this feature, select the **Enable the policy update feature** check box. To disable it, clear the check box.
- 4 Click OK to save your changes and return to the Configuration page.

## Performing an immediate policy update

You might want to perform an immediate policy update if you have applications, patches, content, or updates that you want to immediately distribute to target machines.

You can perform policy updates to machine targets only. If you select an aggregate target (such as the all endpoints target, a domain, a container, or a group) that contains non-machine targets, only machine targets get policy updates. You can perform immediate policy updates while in single selection mode only.

---

Note: The restriction to machine targets applies only to immediate policy updates. When Policy Service runs, it checks for both user- and machine-based policies.

---

## ► To perform an immediate policy update

- 1 From the Target View page or Target Details page, select a target.
- 2 Click Update and choose one of the following options:
  - **Update Directly Assigned Policy**—Choose this option if you want to update the policy directly assigned to this target only and not any policies that the target inherits because it is a member of a group, container, domain, or the all endpoints target.
  - **Update All Policies**—Choose this option if you want to update all the policies that apply to this target, including the policy directly assigned to this target and any policies that the target inherits because it is a member of a group, container, domain, or the all endpoints target.

The Update Preview page appears and allows you to confirm that you are updating the policy or policies for the correct target.

- 3 Confirm your choice by clicking Update Directly Assigned Policy or Update All Policies.

Policy Manager starts to update the policy or policies for the target that you specified. For more information about how policy updates take place, see “How does the immediate policy update work?” on page 349.

The Status of Policy Update page appears and allows you to monitor the status of machines in the target. The page refreshes automatically every ten seconds. See “Monitoring and viewing the status of a policy update” on page 347.

- 4 When policy updates are complete for all the machines in the target you specified, Policy Manager displays the list of machines and indicates whether or not the update was successful.

For machines on which the update failed, you can click the Failed status to view the logs for that specific machine. You can also click Retry Failed to retry policy updates for machines in the target on which they failed.

- 5 Click Done to return to the Target View or Target Details page.

Notice that a check mark icon  appears next to the Update button to indicate that a policy update has completed for this target. If you click the Update button, the additional option View status of last update appears at the bottom of the drop-down list. See “Monitoring and viewing the status of a policy update” on page 347.

## Monitoring and viewing the status of a policy update

After you start a policy update for a target, the Status of Policy Update page appears and allows you to monitor the status of machines in the target.

If you return to the Target View or Target Details page after starting a policy update, the currently updating icon  appears next to the Update button to indicate that a policy update is currently running for this target. If you go to the Target View or Target Details page while a policy update is running for another target, a grayed out version of the icon  appears next to the Update button to indicate that you cannot start another policy update (even for a different target) while there is one currently running. If you click the Update button, the additional option View status of other target appears at the bottom of the drop-down list.

After a policy update runs, the status for the last run is available from the Target View or Target Details page. If you click the Update button, the additional option View status of last update appears at the bottom of the drop-down list. See “Monitoring and viewing the status of a policy update” on page 347. Policy Manager does not maintain a history of previous policy updates.

On the Status of Policy Update page, you can view information about the policy update, including the following items:

- **Target.** The name of the target on which you have chosen to perform the policy update.
- **Started on.** The date and time when Policy Manager started connecting to the endpoints that belong to the target and performing the policy update that you specified.
- **Connection size.** The total number of endpoints to which Policy Manager tries to connect and perform the policy update that you specified.
- **Progress.** The progress of the policy update. This is the percentage of endpoints that Policy Manager has connected to and started the policy update that you specified.

---

Note: The progress for the policy update reflects whether or not Policy Manager was able to connect to the endpoints and start the policy update. It does not reflect whether or not the specified update was completed successfully on the endpoints.

---

You can also view the current status of the individual endpoints (tuners) on which you are performing the policy updates. The status for an endpoint can be one of the following:

- **Pending.** Indicates that Policy Manager is still trying or will soon try to connect to and start the policy update on the endpoint.
- **Stopped.** Indicates that you or another administrator used Policy Manager to stop the policy update. You might also see this state if the policy update on endpoints is stopped because the minimum quorum is not met. (Quorum is a Deployment Manager setting.) The default quorum value is 0, which means that the deployment continues for all the endpoints in the group regardless of how many endpoints have already failed or succeeded.
- **Successful.** Indicates that Policy Manager connected to and started the action on the endpoint.
- **Failed.** Indicates that Policy Manager could connect to the tuner, but failed to complete the policy update on the endpoint.

You can click the Successful or Failed status to view the logs for a specific machine.

## Stopping and retrying policy updates

In addition to viewing status information about a policy update, you can control the policy update in the following ways:

- Stop the policy update.
- Retry the policy update.

### ► To stop or retry a policy update

- 1 On the Status of Policy Update page, do one of the following:
  - Click Stop Job to stop the update completely.

- Click Retry Failed to retry the update on endpoints on which the update failed.
- 2 When you want to exit the Status of Policy Update page, click Done. You return to the Target View or Target Details page.

## How does the immediate policy update work?

When an administrator performs an immediate policy update, Policy Manager communicates with Deployment Manager through RPC to create a deployment for performing the policy update. In Deployment Manager, all the items required for the deployment (server groups, server keychains, task groups, and so on) are created in a folder called *Policy Management Deployments*. If you are logged in to Deployment Manager as a Deployment Manager administrator, you see this folder at the root level. Do not attempt to move or rename it. The folders inside it are also automatically created, as are the rest of the objects inside these folders. For more information about Deployment Manager and deployments, see the *Server Management Administrator's Guide*, available on the BMC Customer Support website.

For the list of machine endpoints (known as a server group in Deployment Manager), Policy Manager uses both the directory service and the database used for Inventory to get information about the machine endpoints included in the target. Scanner Service must run on the endpoint at least once before you can perform an immediate policy update on it. Policy Manager obtains group information and endpoint tuner information (such as the port number) from the database used for Inventory. If an endpoint that belongs to the target is present in the directory service but not in the Inventory database, it is not included in the target list.

For the tuner credentials for the machine endpoints (known as a server keychain in Deployment Manager), Policy Manager uses the tuner user name and password specified in the Deployment Manager Settings page in the CMS console's system settings.

When Deployment Manager creates and runs the deployment, it sends the following commands to Deployment Service, so that Deployment Service can execute them on the endpoints:

- **Update and run Policy Service.** Policy Service updates the policy at the endpoint and starts any updates and installations of applications or patches. If an immediate policy update fails, Policy Service does not retry it automatically. You must retry the update manually using Policy Manager. See “Stopping and retrying policy updates” on page 348.
- **Run Patch Service.** If BMC Patch Management is installed and configured, Patch Service applies the appropriate patches (assigned through the policy) to the endpoint.

Deployment Manager tracks the status of command execution on target endpoint by monitoring the log entries produced by the commands. The Deployment Service on each target endpoint posts log entries to a URL specified by Deployment Manager. By default, the status port--the port used for sending status messages back from target servers to Deployment Manager--is the same port number used for accessing Deployment Manager. That is, by default, both the status port and the Deployment Manager port are 8000. You can change the status port from Deployment Manager. See the Deployment Manager online help. You can view the log entries from each target endpoint using Policy Manager.

## Blackout periods and immediate policy updates

Any blackout period that you set for targets is observed when you perform an immediate policy update. If you want immediate policy updates to take effect during a blackout period, you must exempt the following components from the blackout period:

- Policy Service
- Patch Service (if you have installed Patch Management)
- Specific packages that you want exempted from the blackout period.

See “Setting the blackout period for a target” on page 196 and “Exempting packages from the blackout period” on page 199.

# Chapter 17 Matrix42 OS Migration for BMC Marimba Client Automation-Windows

This chapter describes how you can use Policy Manager to seamlessly migrate to the Microsoft Windows 7 operating system using the BMC Marimba Client Automation 8.2 migration module. The migration module uses the Empirum engine from Matrix42 to perform the migration tasks. The BMC Marimba Client Automation console is used for assigning and managing the OS migration.

---

Note: The BMC Marimba Client Automation 8.2 migration module has been tested on Windows operating systems only, so BMC recommends that you use the module to migrate only Windows computers.

---

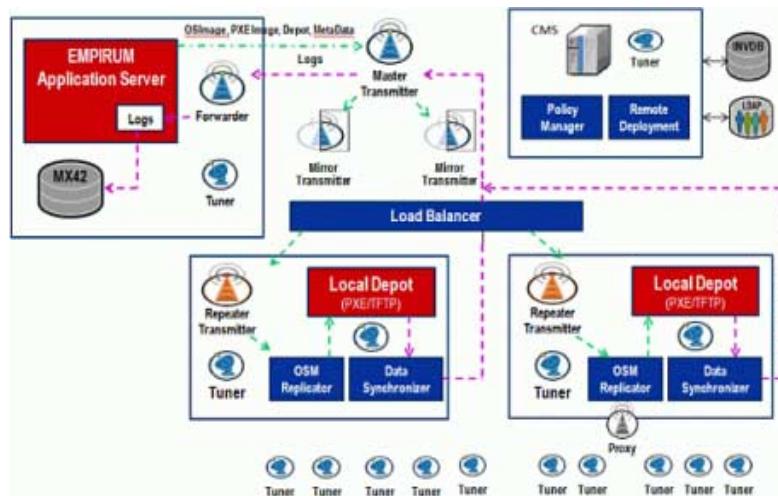
The following topics are provided:

- Architecture and components (page 351)
- Preparing your system for OS migration (page 354)
- OS migration workflow (page 359)
- Troubleshooting using logging codes (page 365)

## Architecture and components

Figure 17-1 on page 352 shows a high-level overview of the BMC Marimba Client Automation OS migration setup.

Figure 17-1: BMC Marimba Client Automation OS migration setup architecture



## Empirum Server or Empirum Masterdepot

The Empirum Server or Empirum Masterdepot, from Matrix 42, provides the components and services needed for OS migration. The Empirum server contains an Empirum Masterdepot *Empirum Masterdepot*, which is similar to the Master Transmitter in the BMC Marimba Client Automation setup. The Empirum Masterdepot contains scripts and infrastructure component installation packages.

---

Note: For more information about the Empirum server, see the *BMC Marimba Client Automation OS Migration with Matrix42 User Guide*.

---

## OS packaging workstation

This component (not shown in the figure) enables the packaging of the OS source files and OS templates.

## Console server

The console server hosts the Java based web server of BMC Marimba Client Automation.

## Master Transmitter

The master transmitter is used for storing any kind of files or folders within the BMC infrastructure. The master transmitter stores all the content required for OS migration, including packages such as OS source files and metadata information such as client logs.

## Repeater Transmitter

The repeater transmitter is a copy or subset of the master transmitter (or of one of its mirrors, as shown in Figure 17-1 on page 352).

## Empirum Subdepot

The Empirum Subdepot, from Matrix 42, hosts the actual or logical copy of the PXE services provided by Matrix42, which enable the clients to install an OS. This server is also used to back up the user's personal data.

The services use the existing files and folder structure (for example, in a sub-depot) and run in offline configuration mode.

## Proxy

The proxy is like a transmitter, but it requests the data from an upper transmitter only on a client's request. In contrast, a transmitter gets its data based on a scheduled and configurable process.

## Clients

The clients, or *endpoints*, contain the BMC Marimba Client Automation tuner component. The client must support PXE for running an OS installation.

---

Note: For information about configuring the PXE system for routed environments, see the *BMC Marimba Client Automation OS Migration with Matrix42 User Guide*.

---

## Preparing your system for OS migration

Before you can perform OS migration, you must specify the settings for the different components in the BMC Marimba Client Automation OS migration setup. This section explains how to configure the different OS migration components.

The following topics are provided:

- 1 Installing the Empirum Server (page 354)
- 2 Configuring Empirum Connection settings (page 355)
- 3 Configuring and deploying the Empirum Masterdepot (page 355)
- 4 Configuring and deploying the Empirum Subdepot (page 356)
- 5 Configuring the DataSync plug-in (page 357)
- 6 Understanding the DataSync channel workflow (page 357)
- 7 Client system requirements (page 358)
- 8 Using the new Report Center queries for OS migration (page 358)
- 9 Understanding the Scanner Service enhancements for OS migration (page 359)

### Installing the Empirum Server

To install the Empirum Server, use the Empirum Installer provided with the 8.2.00 release. For information about installing the Empirum Server, see the *BMC Marimba Client Automation OS Migration with Matrix42 User Guide*.

---

Note: Ensure that your system meets the Matrix42 platform requirements. For information about the Matrix42 supported operating systems and databases, see the *BMC Marimba Client Automation OS Migration with Matrix42 User Guide*.

---

---

Note: For information about downloading drivers from Matrix42, see the *BMC Marimba Client Automation OS Migration with Matrix42 User Guide*.

---

## Configuring Empirum Connection settings

You can configure the CMS Console to integrate with the web services for OS migration by specifying settings for the Empirum Server and the Master Transmitter. Applications that run on the console, such as Policy Manager, can then communicate with the Matrix42 Empirum console.

---

Note: For information about configuring the Matrix42 connection settings, see the *BMC Marimba Client Automation CMS and Tuner User Guide*.

---

## Configuring and deploying the Empirum Masterdepot

You must create a profile for the Empirum Masterdepot and create and deploy the tuner installer for the Empirum Masterdepot computer, that is, the computer on which the Matrix42 Empirum server resides.

---

Note: For information about configuring the Empirum Masterdepot settings, see the *BMC Marimba Client Automation CMS and Tuner User Guide*.

---

### Empirum Masterdepot services

The following Empirum services are available on the Empirum Server or the Empirum Masterdepot:

- Empirum - Interface Hosting Service
- Empirum-Activation
- Empirum-Driver
- Empirum-SWDepotLog
- Empirum-Iperf

## Configuring and deploying the Empirum Subdepot

You must create a profile for the Empirum Subdepot and create and deploy the tuner installer for the Empirum Subdepot computer.

---

Note: For information about configuring the Empirum Subdepot settings, see the *BMC Marimba Client Automation CMS and Tuner User Guide*.

---

### Empirum Subdepot services

The Empirum Subdepot hosts the Matrix42 EMP PXE and Matrix42 EMP TFTP services for enabling the clients that are installing an operating system. These services operate in the **offline** mode, that is, they lack a connection with the central database but use files for receiving and transmitting information.

- **EMP PXE service**

PXE makes it possible to install computers via the network. PXE-enabled clients are automatically registered in the Matrix42 Empirum database when booted via LAN.

Following are the PXE server modes:

- **PXE/DHCP:** The Empirum PXE/DHCP service works as the PXE server and simulates a functionally reduced DHCP server that is able to assign addresses.
- **PXE Only:** The Empirum PXE Only service only works as the PXE server. No DHCP server must be started on the Empirum server.

Following are the PXE service modes:

- **Online:** The online service writes and reads directly in the database.
  - **Offline:** In the offline mode, a connection to the database is not required. The required data is scanned from the respective OS.INI file. Version 8.2 of BMC Marimba Client Automation supports this mode only.
- **EMP TFTP service**

The TFTP service hosted on the Empirum Subdepot computer is the same as that on the Repeater computer, and can access the internal file structure of a repeater directly.

## Configuring the DataSync plug-in

For information about configuring the DataSync plug-in, see the *BMC Marimba Client Automation Report Center User Guide*.

## Understanding the DataSync channel workflow

The DataSync channel is bundled with the tuner that is deployed on the Empirum Subdepot. When the tuner starts, the DataSync channel is subscribed and starts running automatically. Based on the scan schedule configured in Report Center, the logs and credentials information are moved to the destination location specified in the DataSync plug-in configuration page.

The data transfer follows one of the following paths:

- Forwarder configured to insert files to destination location

If you have specified the Forwarder URL in the DataSync plug-in configuration page, the log and credentials files from the Local Depot are forwarded by the repeater to the mirror, and from the mirror to the Forwarder. The Forwarder plug-in places these files in the destination (on the Empirum Masterdepot).

- Forwarder *not* configured to insert files to destination location

If you have *not* specified the Forwarder URL in the DataSync plug-in configuration page, the log and credentials files from the Empirum Subdepot are forwarded by the repeater to the mirror, and from the mirror to the master transmitter. The master transmitter places these files in the destination (on the Empirum Masterdepot).

- Repeater configured to forward the files to the Forwarder

If you have specified the Forwarder URL in the DataSync plug-in configuration page *and* selected **Allow repeaters to insert data directly into the Forwarder**, then the log and credentials files from the Empirum Subdepot are directly forwarded to the Forwarder by the repeater. The Forwarder plug-in places these files in the destination (on the Empirum Masterdepot).

The transfer of data from the specified source location follows the schedule that you have specified in **DataSync Log Collection Schedule**.

## The OS migration Service channel

The OS migration Service channel is an extension channel to the Policy Service that runs backups at the scheduled activation times. The service channel enables execution of personal backups by installing the necessary dependencies, such as the .Net Framework. The channel also checks Subdepots for PXE activation status, and reboots the system only when it is ready for OS migration.

---

Note: The OS migration Service channel must be published on the same location as the Policy Service on the Master Transmitter. During migration activation, the channel is downloaded and executed.

---

## Client system requirements

As part of the OS migration solution, personal backup can be executed before migrating to the target OS. The personal backup feature requires the Microsoft .NET framework, which will be installed as part of the OS migration workflow on the client system. However, the Microsoft .NET framework has a few dependencies, which are not installed by default. You must install the following components for personal backup:

- Windows Imaging Component (WIC): Supported operating systems Windows XP SP2 or Windows Server 2003
- Windows Installer 3.1 or later
- Microsoft Internet Explorer 5.01 or later

## Using the new Report Center queries for OS migration

To facilitate OS migration to Windows 7, the Query Library in Report Center contains new OS migration queries. To access them, select **Windows Client Migration => Windows 7 Migration**.

---

Note: For information about the new Report Center queries for OS migration, see the *BMC Marimba Client Automation Report Center User Guide*.

---

## Understanding the Scanner Service enhancements for OS migration

For information about the Scanner Service enhancements for OS migration, see the *BMC Marimba Client Automation Report Center User Guide*.

## OS migration workflow

After you have specified the settings for the different components in the BMC Marimba Client Automation OS migration setup, you are ready to initiate OS migration. This section describes the OS migration workflow and explains how to initiate OS migration, install the tuner after OS migration, and replicate and synchronize the content across the Empirum Server (or Empirum Masterdepot), Master Transmitter, and the Empirum Subdepot.

The following topics are provided:

- Initiating OS migration (page 359)
- OS migration actions (page 361)
- Tuner installation after OS migration (page 363)
- Replicating content across the Empirum Masterdepot, Master Transmitter, and Empirum Subdepot components (page 364)

## Initiating OS migration

To initiate OS migration, you must create an assignment for the migration to an LDAP group or Collection.

### ► Before you begin

- 1 Create the OS Image in the Empirum Server.

For information about creating the OS Image, see the *BMC Marimba Client Automation OS Migration with Matrix42 User Guide*.

- 2 Create customized personal backup templates from the existing templates.

For information about creating customized templates, see the *BMC Marimba Client Automation OS Migration with Matrix42 User Guide*.

- 3 Ensure that all the BBCA components are updated to version 8.2.00.

- 4 Ensure that the end-points that will be migrated are changed to network reboot so that they can connect to the DHCP server.

- 5 In the BBCA console, under the Applications -> Console -> System Settings -> Empirum Settings page, provide the Publish Transmitter and Subscribe Transmitter host name and port.

The subscribe transmitter details are used by the Empirum Pre-boot Environment (EPE) to download the OS Image from the Empirum Subdepot repeater. The publish transmitter URL is used by the Infrastructure Administration profile page to find the publish location for the OS migration contents.

## ► To initiate OS migration

- 1 Create a collection using the Report Center query option.

You can use the Report Center Windows 7 migration query for this purpose.

---

Note: You must perform an LDAP Sync on the new collection to identify member machines. You can perform an LDAP synchronization on the new collection by navigating to System Settings -> Data Source.

---

- 2 Edit the current policy and specify the OS deployment settings, as described in Specifying policies for OS migration (page 266).
- 3 Enter the activation time and the expiration time for the migration.
- 4 Select the OS Template to be applied to the end-point.

The OS Template corresponds to the OS Image that will be deployed on the end-point.

- 5 Specify the Personal Backup settings, as described in Specifying personal backup settings for OS migration (page 267).

---

Note: For information about customizing the personal backup template, see the *BMC Marimba Client Automation OS Migration with Matrix42 User Guide*.

---

- 6 Select the Reboot Settings tab, and specify the reboot schedule.

This schedule governs when the system is rebooted. After a reboot, the system undergoes a network boot and contacts the Empirum Subdepot for OS migration.

- 7 Click Preview, and then click the Assignment workflow Start button to save the settings on the Empirum Server.
- 8 Click the Save button to store the settings to LDAP.

---

Note: The OS template and personal backup templates are loaded from the Empirum Server. Hence, you must ensure that the Empirum server already contains the OS templates. You must also ensure that the Empirum Server contains the EPE assigned to BMC group, so that the OS templates and the personal backup templates can be assigned to the Client/Assignment Group.

---

---

Note: Ensure that you have installed the Empirum Masterdepot and the Empirum Subdepot before staring the OS migration policy assignment. Also ensure that the local depots for the end-point machines are configured through SBRP configuration in Master Transmitter.

---

## OS migration actions

When you initiate OS migration, the following actions occur:

- 1 The OSM Replicator channel replicates the assignment (created on the Empirum Masterdepot), the Matrix42 Empirum server templates, and the configuration settings to the Master Transmitter and then to all repeaters and the Empirum Subdepot.

For information about the OSM Replicator channel, see [Replicating content across the Empirum Masterdepot, Master Transmitter, and Empirum Subdepot components \(page 364\)](#).

- 2 The Policy Service receives the OS migration policy and reboot schedule as a policy and sets the reboot schedule for the endpoint.
- 3 The local depot server assigned for Personal Backup is received as part of the policy from the SBRP configuration.

---

Note: For personal backup to execute, the SBRP configuration must be modified and the subdepot details must be provided. For details about the SBRP configuration, see the [BMC Marimba Client Automation Transmitter and Proxy Guide](#).

---

- 4 As per the migration activation schedule, the OS Migration service channel is executed for personal backup. After successful completion of personal backup, the PXE entry for the endpoint is activated through the Policy plug-in. The activation status can be verified using the Policy plug-in logs.
- 5 On successful PXE activation, the OS migration service queries the local DHCP server for PXE activation status. When the status is received, the machine gets registered for reboot with the Common Reboot Service (CRS).

---

Note: The PXE activation status might take a long time to reach the local DHCP/PXE server. This happens based on the OSM Replicator channel schedule and the replication schedule between the Master and Repeater transmitters.

---

#### Configuring forceful reboot of client machine for OS migration:

After successful personal backup, the client (endpoint) gets PXE activated in the system. The endpoint checks every 15 mins to see whether it is PXE-activated. By default, the client will wait for one hour (4 attempts) to check whether PXE is activated. After one hour, the system will be restarted to boot into network mode. But the DHCP/PXE activation status may require additional time to reach the Empirum Subdepot based on the replication interval of its environment. To increase the wait time for DHCP wait, set the following property:

`machine.activate.maxwaittime=<number of minutes to wait>`.

The default value is 60 minutes.

You can also use the following property to control the system reboot after the maximum wait time:

`machine.activate.forcereboot=<true/false>`.

If the value is **true**, the system is restarted after the maximum wait time. If the value is **false**, the system is not restarted after the maximum wait time. In this case, the administrator will need to perform a manual restart.

This property must be synchronized with the replication schedule for the Empirum Subdepot. For example, if the replication schedule is 15 mins, the value of this property must be 60 minutes. If the replication schedule is 30 mins, you must set the value of this property to 120 minutes.

- 
- 6 After the reboot is triggered, the endpoint enters network reboot and contacts the local DHCP server for boot loading. The DHCP server initiates boot loading on the Empirum Subdepot.

---

Note: Before initiating migration, ensure that the endpoint is changed to use network boot, so it can connect to the DHCP/PXE server for boot loading.

---

- 7 The Empirum Subdepot and the repeaters are equipped to download the OS image from the repeater to the endpoint and begin OS migration.
- 8 After OS Migration, the tuner can be installed as a post-migration task. For information about installing the tuner post-migration, see Tuner installation after OS migration. For information about configuring the other post-migration tasks, see the *BMC Marimba Client Automation OS Migration with Matrix42 User Guide*.
- 9 After the tuner is installed, the necessary BBCA Service channels will be downloaded using the tuner profile settings. During Policy Service update, the software specific to the end-point is installed.

---

Note: The software specific to the new OS must be available on the master transmitter.

---

## Tuner installation after OS migration

To install the tuner after OS migration, perform the following steps:

- 1 On the Empirum console, click the **Configuration -> OS Installer** tab.
- 2 Using the File menu, open the OS template that you want to deploy on the end-point.
- 3 From the list of OS template items (such as Partition, Protocols, and Command) displayed on the left, click the **Command** item, and modify the value in the **Command:** field to  
\\%EmpirumServer%\Configurator\$\User\BMCAgent.bat.
- 4 Copy the profile tuner installer which must be installed after migration to the Empirum Console server directory,  
\\<EmpirumServer>\Configurator\$\User\

For information about creating the profile tuner installer, see the *BMC Marimba Client Automation CMS and Tuner User Guide*.

- 5 Rename the tuner installer to **TunerSetup.exe**.

---

Note: You can repeat the preceding steps to install other software after OS migration.

---

Note: When personal backup is *not* used with OS migration, the **BMCagent.bat** file must be modified to *not* execute Personal Backup restoration. To accomplish this, remove the following line from the **BMCagent.bat** file: Call  
\\%EmpirumServer%\Configurator\$\User\Setup.exe  
"\%EmpirumServer%\Configurator\$\Packages\matrix42\Personal  
Backup\14.1\Install\Post\_Install\_Force\_Restore.inf" /S1. Rename the modified .bat file, create a different OS template for OS migration without personal backup, and point the new OS template to the modified .bat file (instead of to the original .bat file).

---

## Replicating content across the Empirum Masterdepot, Master Transmitter, and Empirum Subdepot components

The OS migration setup uses an OSM Replicator module to replicate the Empirum Masterdepot configurations and image files on the Empirum Subdepot on a scheduled basis. The OSM Replicator module ensures that when the endpoints periodically communicate with the Empirum Subdepot, they will obtain the updated OS migration configurations.

The OSM Replicator module is an extension of the Content Replicator design, and operates in the following modes:

- **Publish mode**

This mode uses the Empirum Masterdepot settings and data to publish the Empirum Masterdepot components on the Master Transmitter so that the same can be replicated by the Empirum Subdepot. To specify this mode, set the **marimba.mx42.depot.type** property to **masterdepot**:  
**marimba.mx42.depot.type=masterdepot**.

- **Install mode**

This mode uses the Empirum Subdepot settings to subscribe to or install the contents published on the Master Transmitter by the Empirum Masterdepot. To specify this mode, set the **marimba.mx42.depot.type** property to **subdepot**: **marimba.mx42.depot.type=subdepot**.

When OS migration is initiated, the OSM Replicator module reads the **marimba.mx42.depot.type** property. If the property is set to **masterdepot**, then the OS Image, PXE Image, Depot, OSM metadata, and configurations modified periodically in the Empirum Masterdepot are published on to the Master Transmitter as per the schedules specified.

If the property is set to **subdepot**, then the OS Images, PXE Images, Depot, OSM metadata, and configurator channels are installed on the Empirum Subdepot at the specified location. These operations follow the schedule specified in the Profiler configuration.

While the OSM Replicator is running, if any of configurations are changed using Infrastructure Administration, the changes are reflected without stopping the channel. If the configuration provided is incorrect or absent, the OSM Replicator channel is stopped. After 30 minutes (the nonconfigurable default value), the OSM Replicator restarts the channel and checks for the configuration again to start the publish or installation operation.

## Troubleshooting using logging codes

This section lists the information that appears in the log files, such as the log IDs and the corresponding log messages. You can use this information to monitor and troubleshoot problems during the OS migration.

### Policy plug-in

Table 17-1 lists the log IDs and corresponding log messages printed on the Master Depot while publishing the Policy plug-in.

Table 17-1: Log IDs and messages for Policy plug-in

<b>Log ID</b>	<b>Log message</b>
8261	Activate PXE request received
8262	Empirum server authentication failed
8263	PXE Activation failed with exception
8264	PXE Activation succeeded
8265	Empirum server authentication successful

Log ID	Log message
8266	Empirum connection properties not found
8267	Forwarding the request to master tx
8268	Repeater is allowed to connect Empirum server
8269	Transmitter of type Master
8270	Transmitter of type Repeater
8271	Unable to find the machine in Empirum

## OSM Replicator channel

Table 17-2 lists the log IDs and corresponding log messages printed on the Master Depot while publishing the OSM data channel.

Table 17-2: Log IDs and messages for OSM Replicator during Publish

Log ID	Log message
19001	Success: Finished publishing channel: < <i>channel URL</i> >
19010	Publishing: Publishing < <i>Source Folder path</i> > to < <i>Master Transmitter URL</i> >

Table 17-3 lists the log IDs and corresponding log messages printed on the Master Depot while installing the OSM data channel.

Table 17-3: Log IDs and messages for OSM Replicator during Install

Log ID	Log message
19001	Success: Finished installing channel: < <i>channelURL</i> >
19030	Installing: Installing channel: < <i>channelURL</i> > to < <i>destinationFolderPath</i> >

Table 17-4 lists other log IDs and corresponding log messages for the OSM Replicator channel plug-in. The log messages having severity level **MAJOR** are marked in Bold (like this); the remaining logs have severity level **INFO**.

Table 17-4: Additional Log IDs and messages for OSM Replicator

Log ID	Log message
19801	Another instance is running
19802	Initialized the commands successfully
19803	TX authentication settings found
19804	Command successfully parsed
19805	Input configuration is missing
19806	Found schedule available for package
19807	<b>Unable to schedule the command</b>
19808	Invoking publish or install operation of a package
19809	Next schedule for the package
19810	Rep command is successfully constructed
19811	Found configuration changes
19812	Schedule started for package
19813	Unable to identify the mode

---

Note: For more information about log messages related to the OSM Replicator channel, see the Content Replicator log messages in *BMC Marimba Client Automation Reference Guide*.

---

## DataSync channel plug-in

Table 17-5 lists the log IDs and corresponding log messages printed while publishing the DataSync channel plug-in. The log messages having severity level **MAJOR** are marked in Bold (like **this**); the log messages having severity level **MINOR** are marked in **bold and italics** (like **this**); the remaining logs have severity level **INFO**.

Table 17-5: Log IDs and messages for DataSync plug-in

Log ID	Log message
24000	Data Synchronizer started logging
24001	Data Synchronizer stopped logging
24002	<b>Failed to send logs/credentials to plugin</b>

Log ID	Log message
24003	<i>Failed to create temporary file</i>
24004	<i>Failed to compress log file</i>
24005	<i>Failed to connect to the transmitter</i>
24006	Failed to open a plugin connection
24007	Failed to send logs/credentials. Server disk is full

## Infrastructure Administration

Table 17-6 lists the log IDs and corresponding log messages printed while publishing the Infrastructure Administration channel. The log messages having severity level MAJOR are marked in Bold (like this); the remaining logs have severity level INFO.

Table 17-6: Log IDs and messages for the Infrastructure Administration channel

Log ID	Log message
36668	Successfully copied Empirum Local Depot installer files
36669	<b>Failed to copy the Empirum Local Depot installer files</b>
36670	Successfully modified Empirum LocalDepotConfig.ini file
36671	<b>Failed to modify Empirum LocalDepotConfig.ini file</b>

## Policy Service channel

Table 17-7 lists the log IDs and corresponding log messages for the Policy Service channel.

Table 17-7: Log IDs and messages for the Policy Service channel

Log ID	Log message
8655	HTTP reply : UNAUTHORIZED.Unable to authenticate with transmitter
8656	HTTP reply : UNSUPPORTED.SBRP is not configured in transmitter
8657	HTTP reply : INTERNAL_SERVER_ERROR.Internal Server error is occurred in transmitter
8658	Unknown Http reply

<b>Log ID</b>	<b>Log message</b>
8659	Got HTTP_OK reply from transmitter
8660	Failed to find subdepot for the endpoint from transmitter
8661	Found default subdepot for the endpoint
8662	Found dedicated subdepot for the endpoint
8663	Could not find the OSMigrationService channel in policy service channel directory
8664	Not able to subscribe the OSMigration Service channel
8665	[Personal Backup] Schedule reached; invoking OSMigration Service channel in START mode
8666	[Personal Backup] Starts waiting for backup to complete
8667	[Personal Backup] Completed waiting for backup to complete, status
8668	[Personal Backup] PXE activation status after the backup completion
8669	[Personal Backup] OS Migration service is started for PXE activation
8670	[Personal Backup] PXE activation failed, will not restart the machine
8671	Schedule is NOT reached or OUT of the schedule; invoking channel in SUBSCRIBE mode
8672	OSMigrationService channel is already subscribed
8673	Unable to start the OSMigrationService channel
8674	Get Local Depot Information
8675	Activating PXE
8676	[Personal Backup] Backup completed success, sending activate signal to Plugin
8677	[Personal Backup] PXE Activation Status
8678	[Personal Backup] Backup not enabled, doing activation only
8679	[Personal Backup] Activation completed with status

## Infrastructure Service

There are no new logs for the Infrastructure Service.



# Chapter 18 Provisioning a new operating system - Windows

This chapter describes how you can use Policy Manager to provision an operating system onto a new bare-metal computer. BBCA uses the OS Migration module and the Empirum engine from Matrix42 to perform OS provisioning on a bare-metal computer, take the personal backup and restore the backup, and self-provision an OS. You can use the BMC Marimba Client Automation CMS to perform OS provisioning.

---

Note: The BMC Marimba Client Automation 8.2 .01 migration module is tested on Windows operating systems only. BBCA recommends that you use the module to provision and self-provision only Windows operating systems.

---

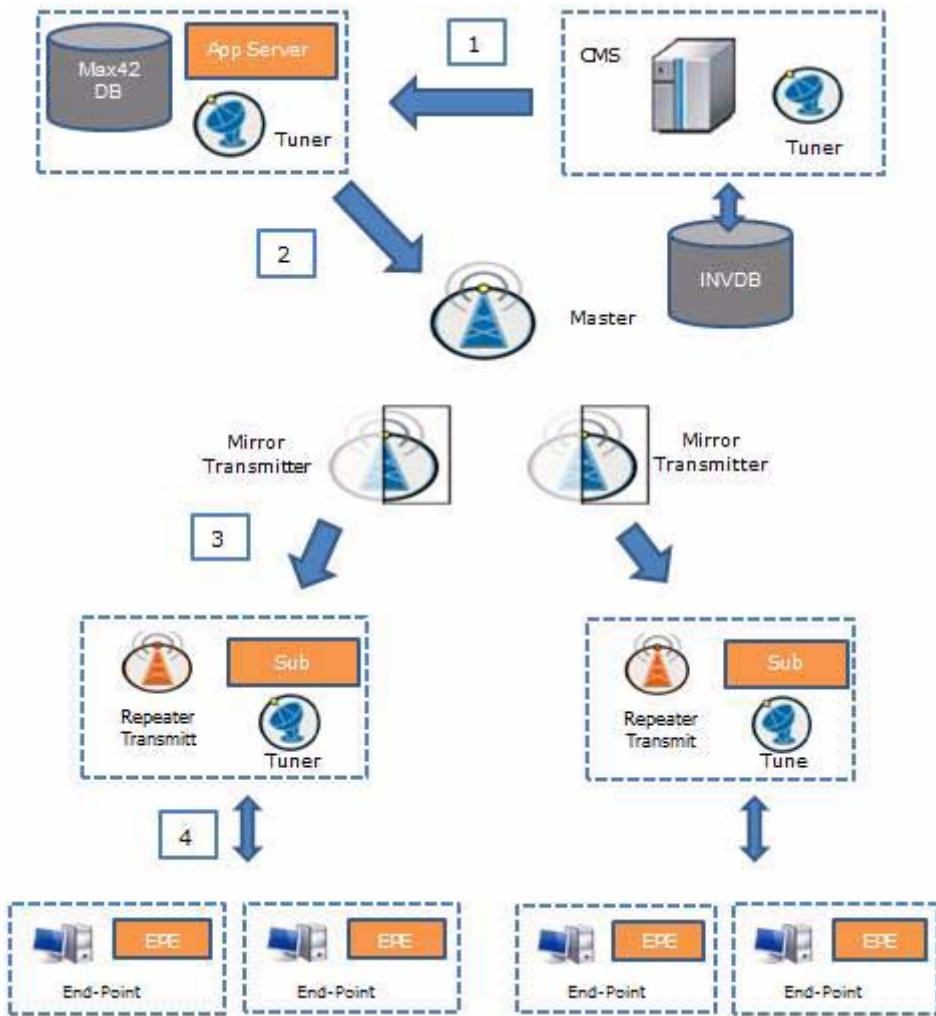
The following topics are provided:

- Architecture of OS provisioning (page 372)
- Prerequisites for OS provisioning (page 373)
- OS provisioning workflow (page 373)
- Provisioning an OS (page 374)

# Architecture of OS provisioning

Figure 18-1 on page 372 shows a high-level overview of the BMC Marimba Client Automation OS provisioning setup.

Figure 18-1: BMC Marimba Client Automation OS provisioning setup architecture



# Prerequisites for OS provisioning

Before using this feature, ensure that you meet the following prerequisites

- The BBCA migration module is configured with the Empirum Connection settings. For more information, see *Matrix42 OS Migration for BMC Marimba Client Automation-Windows* (page 351).
- If you are self-provisioning an OS, the administrators must be created in the Empirum Management Console.
- The OS templates required for OS provisioning are created and ready for deployment in Empirum Management Console.

## OS provisioning workflow

The BBCA CMS allows you to create a bare-metal OS provisioning task to install an OS on a new computer. The OS provisioning content is published to a Master Transmitter, and the content is again replicated to the sub depot using the Repeater Transmitter. BBCA uses the Empirum Pre-boot environment to provision the OS on a bare-metal computer.

This section describes the OS provisioning workflow and explains how to create a group, import the list of computers to be provisioned, select the OS image to be installed, customize the OS template settings.

Note: If you have large number of computers for which you need to provision an OS, you can create a .CSV file that contains the following details:

- Machine name
- Mac address
- Domain name

You can use BBCA OS Migration module to upload the .CSV file which contains the details of the computers.

The OS provisioning task consists of the following steps:

**Step 1** Create a group to provision an OS for a computer.

---

Note: You cannot provision an OS for a computer without creating a group.

---

**Step 2** Add one or more computers to the group.

**Step 3** For each computer, add or edit the OS provisioning template.

**Step 4** Activate OS provisioning or self provisioning for a group or for each computer.

## Provisioning an OS

The following topics are provided:

- Creating a group (page 374)
- Adding and editing a template to provision an OS (page 377)
- Restoring a backup to a computer (page 378)
- Self-provisioning an operating system (page 379)
- Activating and deactivating OS provisioning for computers (page 380)
- Activating and deactivating OS provisioning for groups (page 381)
- Viewing the compliance report for OS provisioning tasks (page 381)
- Editing variables (page 381)

## Creating a group

Before you can add a computer for OS provisioning, you must create a group for the computer. You can view the list of groups in the OS Provisioning tab of Policy Manager module. In the OS provisioning tab, you can perform the following actions on groups:

- Add a group.
- Edit the details of a group.

You only edit the name of the group and its description.

- Delete a group.
- Refresh the list of groups.
- View and edit the computers in a group
- Search for a group.

To search for an existing group, enter the group name in the **Search Group** text box, and click the **Search** icon. Policy Manager displays a list of matching groups in the left pane. When you create a group, Policy Manager creates the group in the Empirum Server.

## ► To add a group

- 1 Choose **Applications > Policy Manager > OS Provisioning**.

Policy Manager displays the OS Provisioning tab where you can see the Groups feature in the left pane.

- 2 To create a new group, click the **Create Group** icon.

Policy Manager displays the Add Group section in the right pane.

- 3 In the **Group Name** text box, enter the name of the group.

- 4 In the **Description** text box, enter the description of the group.

- 5 Click **Save**.

Policy Manager displays a confirmation dialog box which shows the success message.

- 6 Click **Close**.

Policy Manager displays the new group in the list of groups displayed in the left pane. If you do not see the new group, click **Refresh** button.

You can view the list of computers in a group, if the group contains computers. When you click on a group, Policy Manager displays the following group related information in the Group Information section:

- Group name
- Description of the group
- OS template name
- OS template description
- Number of machines in the group

To view the list of computers in an existing group, click on the required group name and then click on **Edit Machines** in the group details section. Policy Manager displays Add/edit machines for group page which shows the list of computers in the group. In this page, you can edit the details of the computers. In this page you can add computers to the group. To add many computers, you can upload a .CSV file.

## ► To add a single computer to a group

- 1 In the OS Provisioning tab, in the left pane, click the required group. Policy Manager displays the details of the selected group.
- 2 In the top right corner of the page, click **Provisioning** icon. Policy Manager displays the Add/Edit machines for group page for the selected group.
- 3 To add a single computer, click **Add**. Policy Manager displays a dialog box where you can specify the required details of the computer.
- 4 Specify the following computer details in the dialog box:
  - Machine Name
  - Mac Id
  - Domain Name
- 5 Click **Save**.

Policy Manager saves the details of the computer and displays the added computer in the Add/Edit machines for group page.

## ► To import a list of computers

- 1 In the Add/Edit machines for a group page, click **Import CSV**.
  - 2 Click **Choose File**.
  - 3 Policy Manager displays the Open dialog box where you can select the required .CSV file which contains the list of computers.

After you have selected the required .CSV file, the Add/Edit machines for group page displays the .CSV file name.
  - 4 Click **Upload**.
- Policy Manager adds the list of computer details and displays the list in the Add/Edit machines for group page.
- 5 To edit any of the computer details, click on the required field and edit. Policy managers saves the modified details.

## Deleting the details of a computer

To delete the details of any computer, in the Add/Edit machines for a group page, select the required computer and click **Delete**.

## Adding and editing a template to provision an OS

After you have added computers to a group, you can add an OS template to each computer. You can also edit the details of a template.

### ► To add or edit a template for provisioning an OS

- 1 In the Add/Edit machines for group page, select one or more computers.
- 2 Click **Next**.

Policy Manager displays the Add/Edit Templates for group page.
- 3 In the Select OS Template section, select the required OS template.
- 4 In the Regional Settings section, select the required details from the following lists
  - Input Locale
  - UI Language
  - System Locale
  - User Locale
- 5 Click **Preview**.

Policy Manager displays the Provisioning Assignment Preview page for group page.
- 6 Perform any of the following tasks, if required:
  - To edit the details of computers, click **Back**.

Policy Manager displays the Add/Edit Templates for a group page.
  - To save the details, click **Save**.

Policy manager displays the Assignment Confirmation Page page.
  - To save and activate provisioning, click **Save & Activate**.
  - To delete the details, click **Cancel**.
- 7 On the Assignment Confirmation Page page, click **OS Provisioning Home Page** link.

Policy Manager displays the OS Provisioning page.

If you click Detailed Status on the Assignment Confirmation Page, Policy Manager displays the OS Provisioning Assignment Logs dialog box where you can view the log details of OS provisioning. Click Done to close the dialog box.

## Restoring a backup to a computer

Policy manager allows you to restore a backup to a computer after provisioning an OS for the computer.

### ► To restore a computer from a backup

- 1 In the Add/Edit machines for group page, click the Restore icon in the Restore From column for required computer.  
Policy Manager displays the Edit Machine page where you can see the Restore from Backup section. By default, the No Restore option is selected.
- 2 If you want to restore a backup of the same machine, select **Same Machine**.
- 3 If you want to restore backup from a different machine, select **Different Machine**.  
Policy Manager enables the Select Subdepot list.
- 4 Select the required subdepot from the Select Subdepot list.  
Policy Manager displays the list of computers available from the selected subdepot.
- 5 Select the required computer from which you want to restore a backup.
- 6 Click **Apply**.

Policy Manager displays the computer name and the subdepot name in the Restore From and Subdepot columns of the Add/Edit machines for a group page.

The Edit Machine page displays the Advanced Variables section which contains the variables names and variable values required to perform restore from backup task. You can edit these variable values.

## Self-provisioning an operating system

You can also self-provision an operating system if you want the administrator to interactively control the installation of an OS on a computer.

### ► To self-provision an operating system

- 1 In the OS Provisioning tab, in the left pane, click the required group for which you want to perform OS self-provisioning..  
Policy Manager displays the details of the selected group.
- 2 In the top right corner of the page, click **Self-Provisioning**.  
Policy Manager displays the Edit Self-Provisioning for group page.
- 3 Select the **Activate Self-provisioning** check box.
- 4 In the Credentials section, select the required administrator.
- 5 In the **Role Display Name** text box, type the name of the role.
- 6 If required, in the Self Provisioning Variables section, select the required variable from the **Associated Variable** list.
- 7 In the **Description** text box, type the description.
- 8 In the **Default Value** text box, type the default value.
- 9 In the **Allowed Value** text box, type the allowed values.
- 10 Click **Add**.

or

To cancel the changes, click **Cancel**.

- 11 Click **Save**.

Policy Manager and Empirum starts the process of self provisioning an operating system.

---

Note: When you deactivate self-provisioning an operating system, Policy Manager fails to remove the folder name with Group ID in the Empirum App Server folder. The retained group folder does not affect self-provisioning an operating system in the endpoint.

---

## Activating and deactivating OS provisioning for computers

You can activate or deactivate OS provisioning for one or more computers.

### ► To activate provisioning an OS for a single computer

- 1 Choose Applications > Policy Manager > OS Provisioning.

Policy Manager displays the OS Provisioning tab where you can see the Groups in the left pane.

- 2 To search for an existing group, enter the required group name in the Search Group text box, and click the Search icon.

Policy Manager displays a list of matching groups in the left pane.

- 3 Click the required group.

- 4 In the top right corner of the page, click **Provisioning**.

Policy Manager displays the Add/Edit machines for group page for the selected group, which displays the list of computers in the group.

- 5 Select the required computers, and click **Activate**.

Policy Manager starts the OS provisioning process for the selected computer.

The last column field of each computer displays the activation status of the computer. The activation status is displayed in the following colors:

- Green

Specifies that the OS provisioning has not started.

- Blue

Specifies that the OS provisioning has started.

- Grey

Specifies that the OS provisioning is complete or that it is a new computer ready for OS provisioning.

---

Note: Once you activate OS provisioning for a computer, Empirum Server automatically starts the OS provisioning task only when the computer restarts. OS provisioning does not start immediately after you click the Activate button.

---

---

Note: To deactivate OS provisioning for a computer, in the Add/Edit machines for a group page, select the required computers and click **Deactivate**.

---

## Activating and deactivating OS provisioning for groups

You can activate and deactivate OS provisioning for groups. To activate OS provisioning for a group, in the OS Provisioning page, select the required group and click **Activate** in the top right corner of the OS Provisioning page. To deactivate OS provisioning for a group, in the OS Provisioning page, select the required group and click **Deactivate** in the top right corner of the OS Provisioning page.

## Viewing the compliance report for OS provisioning tasks

You can view the compliance report for OS provisioning tasks. To view the compliance report, in the OS Provisioning, click **Compliance**. Policy Manager displays the Compliance Information section for OS provisioning tasks. The Compliance Information section displays the following tabs:

- Success machine  
Displays the list of computers for which OS provisioning is completed.
- Failed machines  
Displays the list of computers for which OS provisioning has failed.
- Pending machines  
Displays the list of computers for which OS provisioning is pending.

## Editing variables

You can edit the following common variables for a group:

- Proxy's host name and port to download the OS image
- Proxy's user name to download the OS image
- Proxy's password to download the OS image
- Organizational unit in Active Directory where the machine needs to be added after OS Migration

- Fully qualified domain name that is assigned to the computer

► **To edit the variables for a group**

- 1 In the OS Provisioning page, select the required group.
- 2 In top right corner of the OS Provisioning page, click **Variables**.  
Policy Manager displays the Edit Group Variables page.
- 3 Edit the required values.
- 4 Click **Save**.

Policy Manager saves the modified values.

---

Note: The passwords provided in variables window need to be in Empirum password encrypted format. For more information on Empirum encrypted password format, see Empirum User Guide.

---

The personal backup for an endpoint can be scheduled through a policy. For more information, see Scheduling a personal backup through a policy.

Section

# III Appendices

The appendices discuss the following topics:

- “Command-line reference” on page 385
- “Improving Sun ONE directory LDAP performance” on page 423
- “Troubleshooting” on page 429
- “Removing Policy Management entries from the directory service” on page 439
- “State mappings for policy compliance” on page 445



## Appendix

# A Command-line reference

This appendix describes the command-line options for the Policy Manager channel. These command-line options provide an alternative to using the Policy Manager browser-based interface for some tasks.

The following topics are provided:

- Command-line basics (page 386)
- Using runchannel options (page 389)
- runchannel options by alphabetical order (page 390)
- runchannel options by function (page 392)
- Specifying schedules (page 418)

## Command-line basics

Each tuner comes with a program named `runchannel`, which you can run to start any channel on that tuner from the command line. This section describes the `runchannel` basics for the Policy Manager channel. For information about using the `runchannel` program with other channels, see the command-line chapter in the *BMC Marimba Client Automation Reference Guide*, available on the BMC Customer Support website.

### Location of the `runchannel` program

The `runchannel` program is located in the tuner installation directory. For example, on Windows, the default installation directory is `c:\Program Files\Marimba\Tuner`. On UNIX, there is no default. When you issue commands from the command line, you must switch to the directory in which the executable file is located, or set that directory in the path your system is configured to use for finding commands.

### Syntax for the `runchannel` program

You can use the `runchannel` program to start a channel from the command line.

Use the following syntax for the `runchannel` program:

```
runchannel <channel_URL> [options]
```

where `<channel_URL>` is the URL of the Policy Manager channel that is subscribed to by the local tuner, using the full URL for the channel:

```
http://<transmitter_name>[:<port>]/SubscriptionManager
```

For example, to view the list of command-line options for the Policy Manager channel originating from a transmitter named `t1.company.com`, enter the following command:

```
runchannel http://t1.company.com:5282/SubscriptionManager -help
```

---

Note: Although the `<channel_URL>` specifies the location on the transmitter, `runchannel` starts the channel from the tuner to which it was downloaded.

---

Keep the following syntax points in mind:

- Arguments that contain spaces must be enclosed in quotation marks.

- Each command must be typed without line breaks, although some examples in this section appear on multiple lines because of space limitations.

## Before using runchannel

Before running the Policy Manager from the command-line interface, set the **directory service (LDAP) connection parameters for the CMS channel**.

These parameters are used by Policy Manager when it connects to the directory service to authenticate users and store policies, machine names, and machine groups.

## Running multiple command-line sessions

Every time you invoke `runchannel` to run a Policy Manager command, the Policy Manager creates a new command-line session. Only one command-line session can be active in a Policy Manager at a given time. Therefore, if you invoke the Policy Manager command-line interface with `runchannel` while another invocation of `runchannel` is already running, the second command cannot start processing until the first one has completed.

## Using runchannel with Policy Manager stopped

If Policy Manager is already running at the start of a command-line session, it continues to run when the session ends.

If Policy Manager is not already running at the start of a command-line session, it exits at the end of the session.

---

Note: If you invoke the command-line interface with `runchannel` before you start Policy Manager, the CMS and Policy Manager channels start automatically. However, after each command, the Policy Manager then stops running, which usually results in degraded performance.

---

## Logging in

You must be logged in to the machine where Policy Manager has been installed in order to use the command-line interface. Unlike the browser-based console in which you log in to the Policy Manager one time and work in a single session until you log out, each command-line session requires user authentication, except the `-help` option.

## Providing user authentication

All command-line options for Policy Manager (excluding `-help`) require user authentication. Therefore, all commands that require authentication must include the `-user` and `-password` command-line options, such as:

```
runchannel http://trans.company.com:5282/SubscriptionManager  
-user john -password opensesame  
-list john user
```

For most command-line options, the user name and password must contain permissions for the directory service being used by Policy Manager.

See the `-user` and `-password` command-line options in “Using `runchannel` options” on page 389.

The authentication commands are required only once for each command-line session. You can execute several command-line options with a single authentication. See “Running multiple command-line sessions” on page 387.

## Case sensitivity

Command-line options are case-sensitive.

## Interpreting return codes

Policy Manager returns 0 if all commands specified are executed successfully. A return code of 1 indicates that a command has failed. If any command fails, all subsequent commands in the session are ignored.

## Setting Tuner properties

You can set both tuner and package properties with the `-tuner` command-line option. Note that using `-tuner` overwrites any previous tuner property setting defined by Policy Manager.

You can set a tuner property through Policy Manager, but if you subsequently publish a policy without that property, Policy Management will not remove it from the tuner. To set a property back to its default value, publish a policy with the property value set to null.

## Specifying multiple command instances

You can specify multiple commands on a single command-line session of Policy Manager using the `runchannel` program. If multiple instances of any particular command are specified, the last instance of the command is used.

For example, if you specify the following commands:

```
runchannel http://trans.company.com:5282/SubscriptionManager  
-user john -password opensesame  
-list -cascade  
-list
```

the command executed by the Policy Manager will be `-list`, not `-list -cascade`. Only one set of authentication commands are required for each session. See the `-user` and `-password` command-line options in “Using `runchannel` options” on page 389.

With the exception of multiple instances of the same command noted above, the commands you specify are processed independent of the order in which they are specified. Therefore, commands such as `-user`, `-password`, `-subscribe`, and `-list` can be specified in any order, as long as arguments for each command follow immediately.

---

Note: For users of versions before 5.0, Policy Manager ignores the following command-line options: `-auth`, `-basedn`, `-h`, `-p`, `-usedn`, `-Z`. Beginning with version 6.0, the user authentication options `-D` and `-w` are deprecated and replaced with `-user` and `-password`.

---

## Using `runchannel` options

You can use the `runchannel` program to configure and run Policy Manager, and create and assign policies from the command line.

The syntax of the command-line interface is as follows:

```
runchannel <PolicyManager_URL>
-user <user_name> -password <password>
<options>
```

where

<PolicyManager\_URL> is the URL of the Policy Manager channel, and <user\_name> and <password> are the user name and password that you use to log in to Policy Manager.

See also “Syntax for the runchannel program” on page 386.

## runchannel options by alphabetical order

This section lists the command-line options in alphabetical order. For more information about each option, see the cross-references.

- “-aclCheck -target <target> -perm <permission\_name> <permission>” on page 416
- “-aclGet {[-target <target>] | [-admin <administrator\_name>]} [-perm <permission\_name>]” on page 416
- “-aclRemove -admin <administrator\_name> -target <target> -perm <permission\_name>” on page 416
- “-aclSet -admin <administrator\_name> -target <target> -perm <permission\_name> <permission>” on page 416
- “-changeorder {<target\_name> <target\_type> | -dn <distinguished\_name> } {<package\_url>=[package\_priority\_number]}” on page 401
- “-clientcertpw <password>” on page 393
- “-configSet <key> <value> [-preview}” on page 393
- “-delete {<target\_name> <target\_type> | {<policy>} | -all | -cascade | -dn <dn>}” on page 402
- “-export <directory>” on page 403
- “-import {<directory> | {<file>} }” on page 403
- “-ldapservers <mapping\_file>” on page 394
- “-list [<target\_name> <target\_type> | {<policy>} | -cascade | -channel <channel\_URL> | -dn <dn> ]” on page 404
- “-machines <machines\_file>” on page 396

- “-namespace <child\_container>” on page 405
- “-password <password>” on page 392
- “-patchsubscribe [-modify] {<target\_name> <target\_type> | -dn <dn>} {<patchgroup\_URL>=<assignment\_state>, [<exempt\_from\_blackout>]} [=wowdep <true or false>] [-schedpatch <date\_time\_range\_frequency>] [noalert|countdown=<countdown\_minutes>, [postpone=<postpone\_minutes>]]” on page 406
- “-remedysubscribe [-modify] {<target\_name> <target\_type> | -dn <dn>} {<remedygroup\_URL>=<assignment\_state>, [<exempt\_from\_blackout>]} [=wowdep <true or false>]” on page 408
- “-publish <SubscriptionManager\_URL>” on page 396
- “-publishpw <user\_name> <password>” on page 397
- “-reporter <parameter\_list>” on page 415
- “-setpluginparam -pbinddn <bind\_dn> -bindpasswd <password> -pbasedn <base\_dn> -poolsize <pool\_size> [-usessl] {-phost <host>:<port>} [-expirytime <expiry\_time\_for\_last\_successful\_host\_in\_minutes>]” on page 398
- “-subscribe [-modify] {-remove] {<target\_name> <target\_type> | -dn <dn>} | -targetSource <path\_to\_textfileContaining\_target\_DNS> {<package\_URL>=<package\_state1>, [<package\_state2>], [<package\_priority\_number>], [<exempt\_from\_blackout>]} [-schedblackout <time\_range>] [-schedprimary {<package\_URL>=<date\_time\_range>}] [-schedsecondary {<package\_URL>=<date\_time\_range>}] [-schedupdate {<package\_URL>=<date\_time\_range\_frequency>}] [-schedverifyrepair {<package\_URL>=<date\_time\_range\_frequency>}] | -policySource <path\_to\_textfileContaining\_policy\_info>” on page 409

- “`-tuner [-modify] { [-remove] <target_name> <target_type> | -dn <dn>} {<property_name>[,<property_type>]=<property_value>} | -propertySource <path_to_textfile_containing_property_list>`” on page 412
- “`-txadminaccess <user_name> <password>`” on page 399
- “`-user {<dn> | <cn> | <uid> | <sAMAccountName> | <upn>}`” on page 392

## runchannel options by function

This section describes the Policy Manager command-line options listed earlier in greater detail under the following functional areas:

- “Authentication options” on page 392
- “Configuration options” on page 393
- “Policy options” on page 400
- “Policy Reporter options” on page 415
- “ACL and permission options” on page 416
- “Deprecated options” on page 417

### Authentication options

This section includes the following options:

- “`-user {<dn> | <cn> | <uid> | <sAMAccountName> | <upn>}`” on page 392
  - “`-password <password>`” on page 392
- `-user {<dn> | <cn> | <uid> | <sAMAccountName> | <upn>}` specifies the name of the user logging in to use Policy Manager. Together with the password, the user name is used to authenticate users before executing Policy Manager commands. It is also used when connecting to the directory service.

The user name can be in the following formats:

- <dn> specifies the name of a user in the distinguished name (DN) format. For example, you can specify an Active Directory user as `cn=john,cn=users,dc=company,dc=com` or a Sun ONE Directory user as `uid=john,ou=People,dc=company,dc=com`.
- <cn>, <uid>, or <sAMAccountName> specifies the common name (CN), user ID, or logon name (*sAMAccountName*) of a user. For example, you can specify an Active Directory user with the logon name `john` or a Sun ONE Directory user with the user ID `john`.
- <upn> specifies the user principal name (UPN). For example, you can specify an Active Directory user with the UPN `john@mycompany.com`

`-password <password>`

specifies the password of the user logging in to use Policy Manager. Together with the user name, the password is used to authenticate users before executing Policy Manager commands. It is also used when connecting to the directory service.

## Configuration options

This section includes the following options:

- “`-clientcertpw <password>`” on page 393
- “`-configSet <key> <value> [-preview}`” on page 393
- “`-ldapservers <mapping_file>`” on page 394
- “`-machines <machines_file>`” on page 396
- “`-publish <SubscriptionManager_URL>`” on page 396
- “`-publishpw <user_name> <password>`” on page 397
- “`-setpluginparam -pbinddn <bind_dn> -bindpasswd <password> -pbasedn <base_dn> -poolsize <pool_size> [-usessl] {-phost <host>:<port>} [-expirytime <expiry_time_for_last_successful_host_in_minutes>]`” on page 398
- “`-txadminaccess <user_name> <password>`” on page 399

`-clientcertpw <password>`

specifies the password for the client certificate used when publishing the Policy Service plug-in to the transmitter. You usually use it with the `-publish` option.

*where*

<password> specifies the client-certificate password.

The following example shows -clientcertpw used with the -publish option:

```
runchannel http://trans.company.com:5282/SubscriptionManager  
-user john -password opensesame  
-clientcertpw companycert  
-publish http://trans.company.com:5282/SubscriptionService  
-configSet <key> <value> [-preview]
```

sets the specified Policy Management configuration property. See “Editing attributes of a configuration object” on page 116.on the BMC Customer Support website

*where*

<key> <value> is the key name and value for the attribute in the Subscription configuration object that you want to change.

[-preview} allows you to view the new and old values for the attribute.

**Note:** Although the command-line interface might tell you it succeeded when using the -preview option, it has not made the attribute change. You must run the command without -preview to actually make the change.

Example:

```
runchannel http://mycompany:5282/SubscriptionManager  
-user john -password opensesame  
-configSet marimba.subscription.acl true
```

This example allows you to turn on the ACL feature by setting the value of the marimba.subscription.acl attribute to true.

-ldapservers <mapping\_file>

imports transmitter-to-directory service mappings from the specified file. These mappings are used to assign a list of directory services to each repeater.

**Note:** There is no GUI interface for this feature. The mapping file can be specified from the command line only.

*where*

<mapping\_file> is the full path and name (such as c:\ldap\map\_file.txt) of the file that lists the directory services. This file has the following format:

```
<transmitter_name>,server=<server_value>
<transmitter_name>,basedn=<basedn_value>
<transmitter_name>,binddn=<binddn_value>
<transmitter_name>,password=plain:<password_value>
<transmitter_name>,usessl=<ssl_value>
```

where

<transmitter\_name> is the machine name or IP address of the machine on which the transmitter is running.

<server\_value> is a comma-separated list of one or more <host>:<port> values (such as machine1:389,machine2:389). If you specify more than one <host>:<port> value, the list of servers is used for failover.

**Note:** Each server or repeater in the list must be configured to use the same <basedn\_value>, <binddn\_value>, <password\_value>, and <ssl\_value> settings.

<basedn\_value> is the distinguished name (DN) of a container in the directory service (such as dc=company,dc=com)

<binddn\_value> is the distinguished name (DN) of the user. This value is used by the Policy Service plug-in to connect to the directory service.

<password\_value> is the password in plain-text (unencoded).

**Update note:** The current version of Policy Manager recognizes the Base64-encoded password saved by Policy Management version 4.7.x in the LDAP server mapping file.

<ssl\_value> determines if the plug-in will connect to the directory server in SSL mode. If true, the plug-in will try to connect the directory server in secure (SSL) mode.

The following example shows the contents of a mapping file:

```
mytransmitter,server=myldap:389
mytransmitter,basedn=dc\=mycompany,dc\=com
mytransmitter,binddn=uid\=jouhn,ou=\people,dc\=mycompany,dc\=com
mytransmitter,password=plain:opensesame
mytransmitter,usessl=false
```

Directory services are typically replicated across an organization to improve response and minimize network traffic. You can take advantage of replicated directory services by configuring Policy Management so that each repeater contacts a nearby directory service, eliminating the need to contact the one assigned to the master transmitter. Moreover, you can assign a list of directory services to each repeater. If one directory service in the list fails, the repeater attempts to contact the next one, eliminating single point of failure problems. The mechanism for mapping directory services to repeaters is the LDAP mapping file. In the mapping file, you associate each repeater name with a list of directory services identified by host name, port number, base DN, bind DN, and password.

---

Note: For the changes you make with this command to take effect, you must publish them to the Policy Service plug-in. You can publish the plug-in changes in the same session in which you use the `-ldapservers` command. See the `-publish` command.

---

#### Example:

```
runchannel http://trans.mycompany.com:5282/SubscriptionManager  
-user john -password opensesame  
-ldapservers "c:\ldap\map_file.txt"  
-machines <machines_file>
```

imports the specified machines flat file into Sun ONE Directory. Machines information cannot be imported into Active Directory.

*where*

`<machines_file>` is the full path and name (such as `c:\ldap\mac_file.txt`) of the file that lists the machines and machine groups. This file has the following format:

```
<machine_name1>:<group_name1>,<group_name2>,...  
<machine_name2>:
```

*where*

`<machine_name1>` is a machine to be created in the directory service.

`<group_name1>` and `<group_name2>` are names of groups to which `<machine_name1>` belongs.

`<machine_name2>` is a machine that is not part of any group; no group names follow the colon (:).

If a machine name specified in the file already exists in the directory service, Policy Manager does not overwrite the existing machine object. Policy Manager adds the machine object to the specified groups (if any) if the machine is not already a member of that group.

Example:

```
runchannel http://trans.company.com:5282/SubscriptionManager  
-user john -password opensesame  
-machines "c:\ldap\mac_file.txt"
```

-publish <SubscriptionManager\_URL>

publishes the Policy Service plug-in to a transmitter.

where

<SubscriptionManager\_URL> specifies the subscription manager host name, port and Policy Service plug-in location where the plug-in will be published.

<SubscriptionManager\_URL> must end with the string

SubscriptionService, such as `http://trans.company.com:5282/SubscriptionService`. If you do not provide the SubscriptionService string, it will automatically be appended to the URL.

If access control is enabled on the transmitter, you can provide the required password (and user name, if necessary) using the -publishpw command. The -publishpw command is ignored if access control is not enabled on the transmitter.

The arguments to -publish and -publishpw are not stored by Policy Manager, so they must be specified each time a publish operation is performed.

Example:

```
runchannel http://trans.company.com:5282/SubscriptionManager  
-user john -password opensesame  
-publish http://trans.company.com:5282/SubscriptionService  
-publishpw <user_name> <password>
```

specifies the user name and password (if necessary) required for publishing the Policy Service plug-in to a transmitter. It is usually used with the -publish option. The -publishpw command is ignored if access control is not enabled on the transmitter.

where

<user\_name> specifies the name of the user allowed to publish to the transmitter. This argument is required only if the transmitter access control setting is based on a user name and a password. If the transmitter access control requires only the password, the -publishpw command will require only the <password> argument.

<password> specifies the password of the user allowed to publish to the transmitter, which is specified in plain text (unencoded).

---

Note: The user name and password that you specify with this option are the ones required when publishing to the transmitter. You still need to specify the user name and password required for authentication by the Policy Manager (see the -user and -password command-line options).

---

The following example shows how the -publishpw option can be used with the -publish option:

```
runchannel http://trans.company.com:5282/SubscriptionManager  
-user john -password opensesame  
-publish http://trans.company.com:5282/SubscriptionService  
-publishpw john txpublishpw  
  
-setpluginparam  
-pbinddn <bind_dn> -bindpasswd <password>  
-pbasedn <base_dn>  
-poolsize <pool_size>  
[-usessl]  
{-phost <host>:<port>}  
[-expirytime <expiry_time_for_last_successful_host_in_minutes>]
```

sets parameters used by the Policy Service plug-in to connect to the directory service. You must specify the -publish <PolicyService\_URL> command, along with the -setpluginparam command, to publish the Policy Service plug-in with the new parameters.

*where*

<bind\_dn> and <password> specifies the distinguished name (DN) and password of the user account used when the Policy Service plug-in establishes a connection to the directory service. This user account should have read permissions in the scope of the directory defined by the <base\_dn>, which typically maps to the entire directory tree, such as cn=john,ou=users,dc=company,dc=com. For Active Directory, the user ID can be entered using user principal name (UPN) format, such as john@mycompany.com.

*<base\_dn>* specifies the base distinguished name (DN) for the Policy Service plug-in's directory connection. The *<base\_dn>* determines the scope of the directory view, as seen by the plug-in. In most cases, the *<base\_dn>* will be equivalent to the Sun ONE Directory suffix or the Active Directory domain name, such as o=mycompany.com.

*<pool\_size>* sets the maximum number of connections established in the pool used by the Policy Service plug-in to establish and maintain connections to the directory service. Typically, it can be left set to the default value of 25.

*[ -usessl ]* specifies that an SSL (secure sockets layer) connection should be used when publishing the plug-in.

*<host>:<port>* specifies the host name and port number of a directory service. You can enter multiple servers in the form *<host>:<port>* to create a server list to provide failover protection, such as server1:389,server2:389.

The plug-in tries to connect to each directory service in the list, in succession. If it cannot connect to a particular directory service, it then attempts to connect to the next server in the list until it succeeds or exhausts the list. Server failover functions for both initial connections and also during normal plug-in operation. As long as there is another available directory service in the list, failover should be seamless.

---

Note: This functionality does not provide load balancing capabilities.

---

*<expiry\_time\_for\_last\_successful\_host\_in\_minutes>* specifies the expiration time for a directory service connection. If you specify a list of host names for directory services failover, Policy Manager will go down the list until it successfully connects to a host. It will use that successful host connection until the expiration time that you specify. Then, Policy Manager will attempt to make a new connection to the first host name in the list. Make sure this expiration time follows the host name and port number.

**Example:**

```
runchannel http://trans.company.com:5282/SubscriptionManager  
-user john -password opensesame  
-setpluginparam -pbinddn“cn=Directory Manager” -bindpasswd  
password -poolsize 25 -phost ldap_server1:389 -expirytime 10  
-txadminaccess <user_name> <password>
```

specifies the user name and password for transmitters with restricted access. The user name and password is used to authenticate against the transmitter. For example, if Policy Manager sources users and user groups from the transmitter, the user name and password are used when obtaining the user and user group lists from the transmitter.

*where*

<*user\_name*> specifies the name of the user who has access to the transmitter. If the transmitter is configured to use a password only for administration, enter “\*” for <*user\_name*>.

<*password*> specifies the password of the user with transmitter access, specified in plain text (unencoded). If you want to set the password to blank (no password), use quotation marks with nothing enclosed, such as -txadminaccess john “”. You should also use quotation marks if the password you want to specify contains spaces.

---

Note: The user name and password that you specify with this option are the ones required when subscribing to packages on the transmitter. You still need to specify the user name and password required for authentication by the Policy Manager. See the -user and -password command-line options.

---

The following example shows how you can use -txadminaccess to specify the user name and password for a restricted transmitter, so that you can subscribe the user john to the package MyPackage:

```
runchannel http://trans.company.com:5282/SubscriptionManager  
-user john -password opensesame  
-subscribe john user  
http://trans.company.com:5282/MyPackage=install  
-txadminaccess john subscribepw
```

## Policy options

This section includes the following options:

- “-changeorder {<target\_name> <target\_type> | -dn <distinguished\_name> } {<package\_url>=[package\_priority\_number]}” on page 401
- “-delete {<target\_name> <target\_type> | {<policy>} | -all | -cascade | -dn <dn>}” on page 402
- “-export <directory>” on page 403

- “-import {<directory> | {<file>} }” on page 403
  - “-list [<target\_name> <target\_type> | {<policy>} | -cascade | -channel <channel\_URL>| -dn <dn> ]” on page 404
  - “-namespace <child\_container>” on page 405
  - “-patchsubscribe [-modify] {<target\_name><target\_type> | -dn <dn>} {<patchgroup\_URL>=<assignment\_state>,[<exempt\_from\_blackout>]} [=wowdep <true or false>] [-schedpatch <date\_time\_range\_frequency>] [noalert|countdown=<countdown\_minutes>,[postpone=<postpone\_minutes>]]]” on page 406
  - “-remedysubscribe [-modify] {<target\_name><target\_type> | -dn <dn>} {<remedygroup\_URL>=<assignment\_state>,[<exempt\_from\_blackout>] [=wowdep <true or false>]}” on page 408
  - “-subscribe [-modify] {-remove} {<target\_name><target\_type> | -dn <dn>} | -targetSource <path\_to\_textfile\_containing\_target\_DNS> {<package\_URL>=<package\_state1>,[<package\_state2>],[<package\_priority\_number>],[<exempt\_from\_blackout>]} [-schedblackout <time\_range>] [-schedprimary {<package\_URL>=<date\_time\_range>}] [-schedsecondary {<package\_URL>=<date\_time\_range>}] [-schedupdate {<package\_URL>=<date\_time\_range\_frequency>}] [-schedverifyrepair {<package\_URL>=<date\_time\_range\_frequency>}] | -policySource <path\_to\_textfile\_containing\_policy\_info>” on page 409
  - “-tuner [-modify] {[-remove] <target\_name> <target\_type> | -dn <dn>} {<property\_name>[,<property\_type>]=<property\_value>} | -propertySource <path\_to\_textfile\_containing\_property\_list>” on page 412
- changeorder {<target\_name> <target\_type> | -dn <distinguished\_name> } {<package\_ur>}=[package\_priority\_number]}

modifies the install priority of packages in a Policy by changing numeric values associated with the packages, enabling you to override the original installation sequence the policy specified. If another package in the policy currently owns the package priority number you assign, the system increments the other package's number by one, renumbering subsequent packages on the priority list until the system has resolved numbering conflicts.

where

<*target\_name*> specifies the name of the target. Depending on the target type (value of <*target\_type*>), this argument refers to a machine name in the directory service, or to a user name or a group name obtained either from the directory service or from the transmitter.

<*target\_type*> specifies the type of the target. This argument refers to the type of the object for which the policy is being created. The following values are recognized target types:

- user
- usergroup
- machine
- machinegroup
- all

If <*target\_type*> is all, the system ignores <*target\_name*> and the target object is given the name all\_all.

---

Note: *Collection* is not a valid argument in the command-line interface. Use the machinegroup target type to designate a collection.

---

<*distinguished\_name*> specifies the distinguished name (DN) of the target in the directory service.

<*package\_url*> specifies the URL of the package whose install priority you want to change.

The following example demonstrates how to use the -changeorder flag:

```
C:\Program Files\Marimba\Tuner>runchannel http://  
10.10.137.137:5282/john/SubscriptionManager  
-user john  
-password opensesame  
-changeorder computers container http://alfonso:80/luser/  
Publisher=5
```

---

```
-delete {<target_name> <target_type>|
{<policy>} |
-all |
-cascade |
-dn <dn>}
```

deletes one or more policies. The command has five variations:

- -delete <target\_name><target\_type> deletes the policy specified by the target name and target type from the current child container in the directory service.

*where*

<target\_name> specifies the name of the target. Depending on the target type (value of <target\_type>), this argument refers to a machine name in the directory service or to a user name or a group name obtained either from the directory service or from the transmitter.

<target\_type> specifies the type of the target. This argument refers to the type of the object for which the policy is being created. The following values are recognized target types:

- user
- usergroup
- machine
- machinegroup
- all

---

Note: *Collection* is not a valid argument in the command-line interface. Use the machinegroup target type to designate a collection.

---

- -delete {<policy>} deletes one or more policy entries, specified using <policy> arguments from the current child container in the directory service.

The name of the policy object is obtained by concatenating the two strings <target\_name> and <target\_type>, separating them by an underline character (\_). For example, the following command identifies the policy for the user john:

-delete john\_user

To provide backward compatibility, the name of the policy can be a file name, such as:

-delete john\_user.sub smith\_machine.sub eng\_usergroup.sub

- -delete -all deletes all policies from the current child container in the directory service.
- -delete -cascade deletes all policies from the directory service.
- -delete -dn <dn> deletes the policy specified by the target's distinguished name (DN). This command is useful when two or more groups have the same common name under different organizational units in the directory service, such as cn=salesgroup,ou=newyork,dc=company,dc=com and cn=salesgroup,ou=sanfrancisco,dc=company,dc=com.

-export <directory>

exports all the policies as policy files (.sub files) to the specified directory.

*where*

<directory> specifies the full path and name of a directory (folder) on the local file system.

**Example:**

```
runchannel http://trans.company.com:5282/SubscriptionManager  
-user john -password opensesame  
-export "c:\subscription_files"
```

-import {<directory> | {<file>}}

imports one or more policy files (.sub files) that have been exported from Policy Manager.

*where*

<directory> specifies the full path and name of a directory (folder) on the local file system that contains one or more policy files to be imported. If some objects in the imported files already exist as objects in the directory service, the import will fail.

<file> specifies the full path and name of a file that contains policy information, with the format <target\_name>\_<target\_type>.sub. You can specify multiple policy files, separating them with a space. If some objects in the imported files already exist as objects in the directory service, the import will fail.

**Example:**

```
runchannel http://trans.company.com:5282/SubscriptionManager  
-user john -password opensesame  
-import  
"c:\subscription_files\cn=machines,dc=domain,dc=com_container.sub"
```

---

```
-list [<target_name> <target_type> |
{<policy>} |
-cascade |
-channel <channel_URL>|
-dn <dn> ]
```

lists information about one or more policies. The command has the following variations:

- -list (without any arguments) lists attributes of all policies in the current child container of the directory service.
- -list <target\_name> <target\_type> lists the policies specified by the target names and target types in the current child container of the directory service.

*where*

<target\_name> specifies the name of the target. Depending on the target type (value of <target\_type>), this argument refers to a machine name in the directory service or to a user name or a group name obtained either from the directory service or from the transmitter.

<target\_type> specifies the type of the target. This argument refers to the type of the object for which the policy is being created. The following values are recognized target types:

- user
- usergroup
- machine
- machinegroup
- all

If <target\_type> is all, <target\_name> is ignored and the target object is given the name all\_all.

---

Note: *Collection* is not a valid argument in the command line interface. Use the machinegroup target type to designate a collection.

---

- -list {<policy>} lists attributes of the specified policy objects. Only objects from the current child container (specified with -namespace) are listed.

*where*

*<policy>* specifies a policy object. The value of *<policy>* must be of the form *<target\_name>\_<target\_type>.sub* (such as *john\_user.sub*) for backward compatibility with earlier Policy Management releases.

- -list -cascade lists all policies present in the directory service.
- -list -channel *<channel\_URL>* lists the subscription targets for which a channel/package is currently assigned. For example:
  - `runchannel http://10.10.51.56:5282/rtgadmin/  
SubscriptionManager -user z -password z -list -channel http://  
10.10.51.20:80/CCMBDDM/712a_ga/BDDMService`
  - If the channel url contains spaces, you must specify the url within double quotes. For example:  
`runchannel http://10.10.51.56:5282/rtgadmin/  
SubscriptionManager -user z -password z -list -channel "http://  
10.10.51.20:80/CCMBDDM/712a_ga/BDDM Service"`
- -list -dn *<dn>* lists the policy specified by the target's distinguished name (DN). This command is useful when two or more groups have the same common name under different organizational units in the directory service, such as *cn=salesgroup,ou=newyork,dc=company,dc=com* and *cn=salesgroup,ou=sanfrancisco,dc=company,dc=com*.

-namespace *<child\_container>*

specifies the child container (previously called namespace) used for storing policies in the directory service. You use this command-line option with other options.

*where*

*<child\_container>* is the common name (CN) attribute of a container object used to store policy entries. This container will be used for listing, creating, and deleting policies. It is also used to set tuner properties for Policy Management entries residing directly under the *<child\_container>* container object.

If the -namespace option is not specified, Policy Manager uses the top-level container specified by the Subscription configuration object in the directory service.

---

Note: Policy Manager supports only one level of child containers under *ou=Subscriptions <Suffix>*.

---

In the following example, Policy Manager lists the policies in the child container `child1`:

```
runchannel http://trans.company.com:5282/SubscriptionManager
-user john -password opensesame
-namespace child1
-list

-patchsubscribe
[-modify]
{<target_name> <target_type> | -dn <dn>}
{<patchgroup_URL>=<assignment_state>,[<exempt_from_blackout>]}
[=wowdep <true or false>]
[-schedpatch <date_time_range_frequency>]
[noalert|countdown=<countdown_minutes>,[postpone=<postpone_minutes>]]]
```

subscribes a target, identified by `<target_name>` and `<target_type>` or by `<dn>`, to a patch group identified by `<patchgroup_URL>`.

where

`-modify` specifies that you want to edit an existing policy, but you do not want to overwrite it; changes and additions that you make will be appended to the policy. If you omit the `-modify` option, your changes will overwrite any previously assigned packages, patch groups, schedules, and settings in an existing policy.

`<target_name>` specifies the name of the target. Depending on the target type (value of `<target_type>`), this argument refers to a machine name in the directory service, or to a user name or a group name obtained either from the directory service or from the transmitter. Policy Manager will not create a policy object if the user or machine or group specified by `<target_name>` does not exist.

`<target_type>` specifies the type of the target. This argument refers to the type of the object for which the policy is being created. The following values are recognized target types:

- user
- usergroup
- machine
- machinegroup
- all

If `<target_type>` is `all`, `<target_name>` is ignored and the target object is given the name `all_all`.

---

Note: *Collection* is not a valid argument in the command-line interface. Use the `machinegroup` target type to designate a collection.

---

`-dn <dn>` specifies the distinguished name (DN) of the target. This option is useful when two or more groups have the same common name under different organizational units (OUs) in the directory service, such as `cn=salesgroup,ou=newyork,dc=company,dc=com` and `cn=salesgroup,ou=sanfrancisco,dc=company,dc=com`.

`<patchgroup_URL>` specifies the URL of the patch group that you want to distribute to targets. You can assign multiple patch groups to a target by specifying multiple patch group URLs.

`<assignment_state>` determines whether or not a patch group should be assigned to target machines. There are two patch group assignment states that you can assign to patch groups in Policy Manager: `assign` or `exclude`.

`<exempt_from_blackout>` specifies whether or not the patch group identified by `<patchgroup_URL>` should be exempt from the blackout period. `True` indicates that the patch group is exempt from the blackout period, while `false` indicates the blackout period applies to the patch group. If you assign multiple patch groups to a target, you can specify a blackout exemption option for each one.

`=wowdep <true or false>` specifies whether or not the target machines should be woken up for the deployment if any of the machines are powered off.

`-schedpatch <date_time_range_frequency>` specifies the schedule for updating Patch Service on a target. During an update, Patch Service scans the target for the list of required and already installed patches. It also installs a set of patches, taking into account the dependencies of each patch. You can specify that updates take place on a recurring schedule. This schedule applies to the entire target; you cannot specify a different schedule for each patch group.

For more information about the format for the schedule, see “Format for update or repair schedules” on page 422. For more information about the Patch Service schedule, see “Overriding the Patch Service update schedule for target machines” on page 304 and “Exempting Patch Service from the blackout period” on page 305.

---

```
-remedysubscribe
[-modify]
{<target_name> <target_type> | -dn <dn>}
{<remedygroup_URL>=<assignment_state>,[<exempt_from_blackout>]
[=wowdep <true or false>]}
```

subscribes a target, identified by `<target_name>` and `<target_type>` or by `<dn>`, to a remediation group identified by `<patchgroup_URL>`.

*where*

`-modify` specifies that you want to edit an existing policy, but you do not want to overwrite it; changes and additions that you make will be appended to the policy. If you omit the `-modify` option, your changes will overwrite any previously assigned packages, remediation groups, schedules, and settings in an existing policy.

`<target_name>` specifies the name of the target. Depending on the target type (value of `<target_type>`), this argument refers to a machine name in the directory service, or to a user name or a group name obtained either from the directory service or from the transmitter. Policy Manager will not create a policy object if the user or machine or group specified by `<target_name>` does not exist.

`<target_type>` specifies the type of the target. This argument refers to the type of the object for which the policy is being created. The following values are recognized target types:

- user
- usergroup
- machine
- machinegroup
- all

If `<target_type>` is `all`, `<target_name>` is ignored and the target object is given the name `all_all`.

---

Note: *Collection* is not a valid argument in the command-line interface. Use the `machinegroup` target type to designate a collection.

---

`-dn <dn>` specifies the distinguished name (DN) of the target. This option is useful when two or more groups have the same common name under different organizational units (OUs) in the directory service, such as `cn=salesgroup,ou=newyork,dc=company,dc=com` and `cn=salesgroup,ou=sanfrancisco,dc=company,dc=com`.

`<remedygroup_URL>` specifies the URL of the remediation group that you want to distribute to targets. You can assign multiple remediation groups to a target by specifying multiple remediation group URLs.

`<assignment_state>` determines whether or not a remediation group should be assigned to target machines. There are two remediation group assignment states that you can assign to remediation groups in Policy Manager: assign or exclude.

`<exempt_from_blackout>` specifies whether or not the remediation group identified by `<remedygroup_URL>` should be exempt from the blackout period. True indicates that the remediation group is exempt from the blackout period, while false indicates the blackout period applies to the remediation group. If you assign multiple remediation groups to a target, you can specify a blackout exemption option for each one.

`=wowdep <true or false>` specifies whether or not the target machines should be woken up for the deployment if any of the machines are powered off.

```
-subscribe
[-modify]
[-remove]
{<target_name> <target_type> | -dn <dn>} |
- targetSource <path_to_textfile_containing_target_DNS>
{<package_URL>=<package_state1>,[<package_state2>],[<package_priority_number>],[<exempt_from_blackout>]}
[-schedblackout <time_range>]
[-schedprimary [<package_URL>=<date_time_range>}]
[-schedsecondary [<package_URL>=<date_time_range>}]
[-schedupdate [<package_URL>=<date_time_range_frequency>}]
[-schedverifyrepair [<package_URL>=<date_time_range_frequency>]] |
-policySource <path_to_textfile_containing_policy_info>
```

subscribes a target, identified by `<target_name>` and `<target_type>` or by `<dn>`, to a package identified by `<package_URL>`.

*where*

`-modify` specifies that you want to edit an existing policy, but you do not want to overwrite it; changes and additions that you make will be appended to the policy. If you omit the `-modify` option, your changes will overwrite any previously assigned packages, schedules, and settings in an existing policy.

-remove specifies that you want to remove the specified channels from the specified targets. For example:

```
runchannel http://10.10.51.28:5282/ant/nrao/SubscriptionManager -  
user z -password z -subscribe -modify -remove -targetSource  
“c:\targets\desktopdns.txt” http://appupdate.web.gs.com:5282/D/  
DeviceLock_3_0_2 http://appupdate.web.gs.com:5282/N/  
NetFramework_3_5
```

*<target\_name>* specifies the name of the target. Depending on the target type (value of *<target\_type>*), this argument refers to a machine name in the directory service, or to a user name or a group name obtained either from the directory service or from the transmitter. Policy Manager will not create a policy object if the user or machine or group specified by *<target\_name>* does not exist.

*<target\_type>* specifies the type of the target. This argument refers to the type of the object for which the policy is being created. The following values are recognized target types:

- user
- usergroup
- machine
- machinegroup
- all

If *<target\_type>* is all, *<target\_name>* is ignored and the target object is given the name all\_all.

---

Note: *Collection* is not a valid argument in the command-line interface. Use the machinegroup target type to designate a collection.

---

-dn *<dn>* specifies the distinguished name (DN) of the target. This option is useful when two or more groups have the same common name under different organizational units (OUs) in the directory service, such as cn=salesgroup,ou=newyork,dc=company,dc=com and cn=salesgroup,ou=sanfrancisco,dc=company,dc=com.

-targetSource *<path\_to\_textfileContaining\_target\_DNS>* specifies the physical path of the file from which the target dns (FQDNs) can be loaded.

*<package\_URL>* specifies the URL of the package that you want to distribute to targets.

`<package_state1>,[<package_state2>]` specifies the primary and secondary states for the package identified by `<package_URL>`. The following states are available:

- advertise
- exclude
- install
- install-persist
- install-start
- install-start-persist
- primary
- stage
- uninstall

`<package_priority_number>` is an integer that specifies the installation priority for the package identified by `<package_URL>`.

`<exempt_from_blackout>` specifies whether or not the package identified by `<package_URL>` should be exempt from the blackout period. True indicates that the package is exempt from the blackout period, while false indicates the blackout period applies to the package.

`-schedblackout <time_range>` specifies the blackout period for the target. For more information about the format for the schedule, see “Specifying blackout schedules” on page 418.

`-schedprimary {<package_URL>=<date_time_range>}` specifies the schedule for enforcing the primary installation state of the package. For more information about the format for the schedule, see “Format for primary or secondary schedules” on page 422.

`-schedsecondary {<package_URL>=<date_time_range>}` specifies the schedule for enforcing the secondary installation state of the package. For more information about the format for the schedule, see “Format for primary or secondary schedules” on page 422.

`-schedupdate {<package_URL>=<date_time_range_frequency>}` specifies the update schedule for the package. For more information about the format for the schedule, see “Format for update or repair schedules” on page 422.

`-schedverifyrepair {<package_URL>=<date_time_range_frequency>}` specifies the repair schedule for the package. For more information about the format for the schedule, see “Format for update or repair schedules” on page 422.

The following example subscribes the user john to the package MyPackage with the primary state install and the priority 1. It sets a primary schedule that is active between January 1, 2002 at 4 AM and June 30, 2002 at 6 PM and an update schedule that updates every two weeks on Mondays and Wednesdays at 4 AM, active beginning at 5 AM on January 1, 2002:

```
runchannel http://trans.mycompany.com:5282/SubscriptionManager
-user john -password opensesame
-subscribe -modify john user
http://trans.mycompany.com:5282/MyPackage=install,1
-schedprimary http://trans.mycompany.com:5282/MyPackage="active
01/01/2002@4:00AM - 06/30/2002@6:00PM"
-schedupdate http://trans.mycompany.com:5282/MyPackage="every 2
weeks on mon+wed update at 4:00AM active 01/01/2002@5:00AM"
```

-policySource <path\_to\_textfile\_containing\_policy\_info> specifies a set of channels and their policy parameters such that policy details can be loaded from the file and applied to the targets mentioned in {<target\_name> <target\_type> | -dn <dn>} |

The policy file specified should have the policy information in the following format: http://appupdate.web.gs.com:5282/D/

DeviceLock\_3\_0\_2=install, 101 including other parameters in the format [-schedblackout <time\_range>] [-schedprimary

{<package\_URL>=<date\_time\_range>} [-schedsecondary  
{<package\_URL>=<date\_time\_range>} [-schedupdate

{<package\_URL>=<date\_time\_range\_frequency>} [-schedverifyrepair  
{<package\_URL>=<date\_time\_range\_frequency>}]

- targetSource <path\_to\_textfile\_containing\_target\_DNS>

-tuner [-modify] {[[-remove] <target\_name> <target\_type> | -dn <dn>] {<property\_name>[,<property\_type>]=<property\_value>} | -propertySource <path\_to\_textfile\_containing\_property\_list>}

sets one or more properties for a target tuner specified by <target\_name> and <target\_type> or by <dn>. This command-line option can also be used to set one or more properties for one or more packages on the target tuner.

where

[**-modify**] adds, appends, or removes the specified properties from the specified targets. For example:

```
runchannel http://10.10.51.28:5282/ant/nrao/SubscriptionManager  
-user z -password z -tuner -modify -targetSource  
“c:\targets\desktopdns.txt” marimba.subscription.retrycount=6  
marimba.subscription.nodelete=false
```

In this example, the properties `marimba.subscription.retrycount=6` and `marimba.subscription.nodelete=false` will be appended to each target DNS listed in the file specified by `-targetSource`.

[**-remove**] removes the specified properties from the specified targets. For example:

```
runchannel http://10.10.51.28:5282/ant/nrao/SubscriptionManager  
-user z -password z -tuner -modify -remove nrao machine  
-propertySource “c:\policy\desktopproperty.txt”
```

In this example, the properties listed in the file

`marimba.subscription.retrycount=6` will be removed from the target `nrao machine`.

`<target_name>` specifies the name of the target. Depending on the target type (value of `<target_type>`), this argument refers to a machine name in the directory service, or to a user name or a group name obtained either from the directory service or from the transmitter.

`<target_type>` specifies the type of the target. This argument refers to the type of the object for which the policy is being created. The following values are recognized target types:

- `user`
- `usergroup`
- `machine`
- `machinegroup`
- `all`

If `<target_type>` is `all`, `<target_name>` is ignored and the target object is given the name `all_all`.

---

Note: `Collection` is not a valid argument in the command-line interface. Use the `machinegroup` target type to designate a collection.

---

-dn <dn> specifies the distinguished name (DN) of the target. This option is useful when two or more groups have the same common name under different organizational units (OUs) in the directory service, such as cn=salesgroup,ou=newyork,dc=company,dc=com and cn=salesgroup,ou=sanfrancisco,dc=company,dc=com.

<property\_name> is the name of the property you want to set.

<property\_type> specifies the type of property you want to set: \* (all packages), subscribers (all subscribed packages), <package\_ur1> (a specific package), service (Policy Service channel). Omit <property\_type> for tuner properties.

<property\_value> is the value of the property you want to set.

For more information about the format for setting properties, see “Tuner and package properties format” on page 287. “Setting tuner and package properties for a target” on page 275 lists some tuner and package properties that you usually use with Policy Management. For more tuner and package properties, see the *BMC Marimba Client Automation Reference Guide*, available on the BMC Customer Support website.

The following example sets the following properties for the user john:

- The tuner property marimba.security.trusted.transmitters is set to the value trans.company.com.
- The package property reboot.showdialog is set to true for the package MyPackage.

```
runchannel http://trans.company.com:5282/SubscriptionManager
-user john -password opensesame
-tuner john user
marimba.security.trusted.transmitters=trans.company.com
reboot.showdialog=http://trans.company.com:5282/MyPackage=true
```

```
runchannel <Subscription Manager URL> -user <user> -password <password> -namespace
<Child Container Name> -export <dir_path>
```

exports the policies from Policy Manager.

where

-namespace

is optional, when that option is not given, policies from OU=Subscription and all other child containers will be exported. When namespace is given a value, then only the policies from that child container alone will be exported to disk.

Examples:

(OU=Child1 and OU=Child2 are child containers of OU=Subscription)

```
runchannel http://<URL of Subscription Manager>/SubscriptionManager -  
user binddn -password password -export c:\sub
```

This command exports all policies from OU=Subscription,  
OU=Child1,OU=Subscription and OU=Child2,OU=Subscription.

```
runchannel http://<URL of Subscription Manager>/SubscriptionManager -  
user binddn -password password-namespace cn=Child1 -export c:\sub
```

This command exports all the policies stored inside OU=Child1namespace.

## Policy Reporter options

This section includes the following option:

- “-reporter <parameter\_list>” on page 415

`-reporter <parameter_list>`

invokes the Policy Service in reporter mode and lists the policies assigned to an endpoint and user. For example, if you assign Channels A through C to an endpoint, then running this command lists Channel A, Channel B, and Channel C. If you run this command without using `-machinename` to identify an endpoint, the reporter lists the policies assigned to your local host.

*where*

`<parameter_list>` can include any combination of the following options:

`-v` indicates verbose mode.

`-machinename <machinename>` provides the machine name for which you want to list the policy.

`-userdn <userdn>` specifies the fully qualified distinguished name (DN) for a user target. Use this option only for Active Directory (Auto Discovery Mode).

`-machinedn <machinedn>` specifies the fully qualified DN for a machine target. DNs are useful for simulating plug-in and targets that are not in the same domain in an Active Directory forest environment. Use this option only for Active Directory (Auto Discovery Mode).

- d <name> specifies the name of the directory used for storing the data obtained from the plug-in.
- refreshconfig <refreshconfig> ensures that Policy Reporter does not cache the config.xml file and does not show inaccurate results if the plug-in cannot contact the directory service.

**Example:**

```
runchannel http://trans.company.com:5282/SubscriptionService -reporter -machinename TestMachine1
```

## ACL and permission options

This section includes the following options:

- aclCheck -target <target> -perm <permission\_name> <permission>  
checks whether the logged on user has the specified permission on the specified target. The resolution of permissions is done recursively.
- aclGet {[ -target <target> ]|[-admin <administrator\_name>]} [-perm <permission\_name>]  
returns an enumeration of permissions for the specified item. Specify either -target or -admin (but not both). If -target is specified, a list of administrators and their corresponding ACLs is displayed. If -admin is specified, a list of targets and the permissions on the targets assigned to that administrator is displayed.
- aclRemove -admin <administrator\_name> -target <target> -perm <permission\_name>  
removes the specified permission for the specified administrator on the target. To perform this operation, the logged on user must have permissions to change the permission for the administrator.
- aclSet -admin <administrator\_name> -target <target> -perm <permission\_name> <permission>  
sets the permission for the specified administrator on the target. To perform this operation, the logged on user must have permissions to change the permission for the administrator. This command-line option replaces any existing permissions with the ones that you specify. For example, if read permissions are already set for the target and you want to additionally set write permissions, you must specify both read and write permissions; otherwise, only write permissions will be set.

*where*

<administrator\_name> identifies the administrator or group of administrators being assigned permissions (such as a user, user group, OU, CN, or a domain). <administrator\_name> is the distinguished name (DN) of an object in the directory service.

<target> identifies the object in the directory service to which the permissions apply (such as a user, machine, group, Organizational Unit (OU), CN, or a domain). <target> is the distinguished name (DN) of an object in the directory service.

<permission\_name> is the type of permission being assigned:

SubscriptionPermission or AclPermission

<permission> is a comma-delimited list of the permissions being assigned:  
read or write

The following combination of permissions are available:

- SubscriptionPermission read allows the administrator to view a policy.
- SubscriptionPermission write allows the administrator to create, edit, or delete a policy.
- AclPermission read allows the administrator to view permissions.
- AclPermission write allows the administrator to set permissions.

## Deprecated options

This section includes the deprecated options:

- “-aclCheck -target <target> -perm <permission\_name> <permission>” on page 416
- “-aclGet { [-target <target>] | [-admin <administrator\_name>] } [-perm <permission\_name>]” on page 416
- “-aclRemove -admin <administrator\_name> -target <target> -perm <permission\_name>” on page 416
- “-aclSet -admin <administrator\_name> -target <target> -perm <permission\_name> <permission>” on page 416
- “-D {<dn> | <uid>}” on page 417
- “-w <bind\_password>” on page 418

-D {<dn> | <uid>}

(deprecated; see “`-user <dn> | <cn> | <uid> | <sAMAccountName> | <upn>`” on page 392) specifies the name of the user logging in to use Policy Manager. You can use either the distinguished name (DN) or user ID. Together with the password, the user name is used to authenticate users before executing Policy Manager commands.

*where*

`<dn>` specifies the name of a directory service user in the distinguished name (DN) format. For example, you can specify an Active Directory user as `cn=john,cn=users,dc=company,dc=com` or a Sun ONE Directory user as `uid=john,ou=People,dc=company,dc=com`.

`<uid>` specifies the user ID or login name of a directory service user. For example, you can specify an Active Directory user in the following formats:

- The user principal name (UPN) such as `john@company.com`
  - The logon name (sAMAccountName) such as `john`
- or a Sun ONE Directory user with the user ID such as `john`.

`-w <bind_password>`

(deprecated; see `-password`) specifies the directory service bind password, specified in clear text without encoding. Together with the user name, the password is used to authenticate users before executing Policy Manager commands.

## Specifying schedules

This section describes how to specify schedules using the `-subscribe` command.

### Specifying blackout schedules

You use the following command-line option to specify a blackout schedule:

`-schedblackout <time_range>`

*where*

`<time_range>` is a range of time in the form

`HH:MM{AM|PM} - HH:MM{AM|PM}`

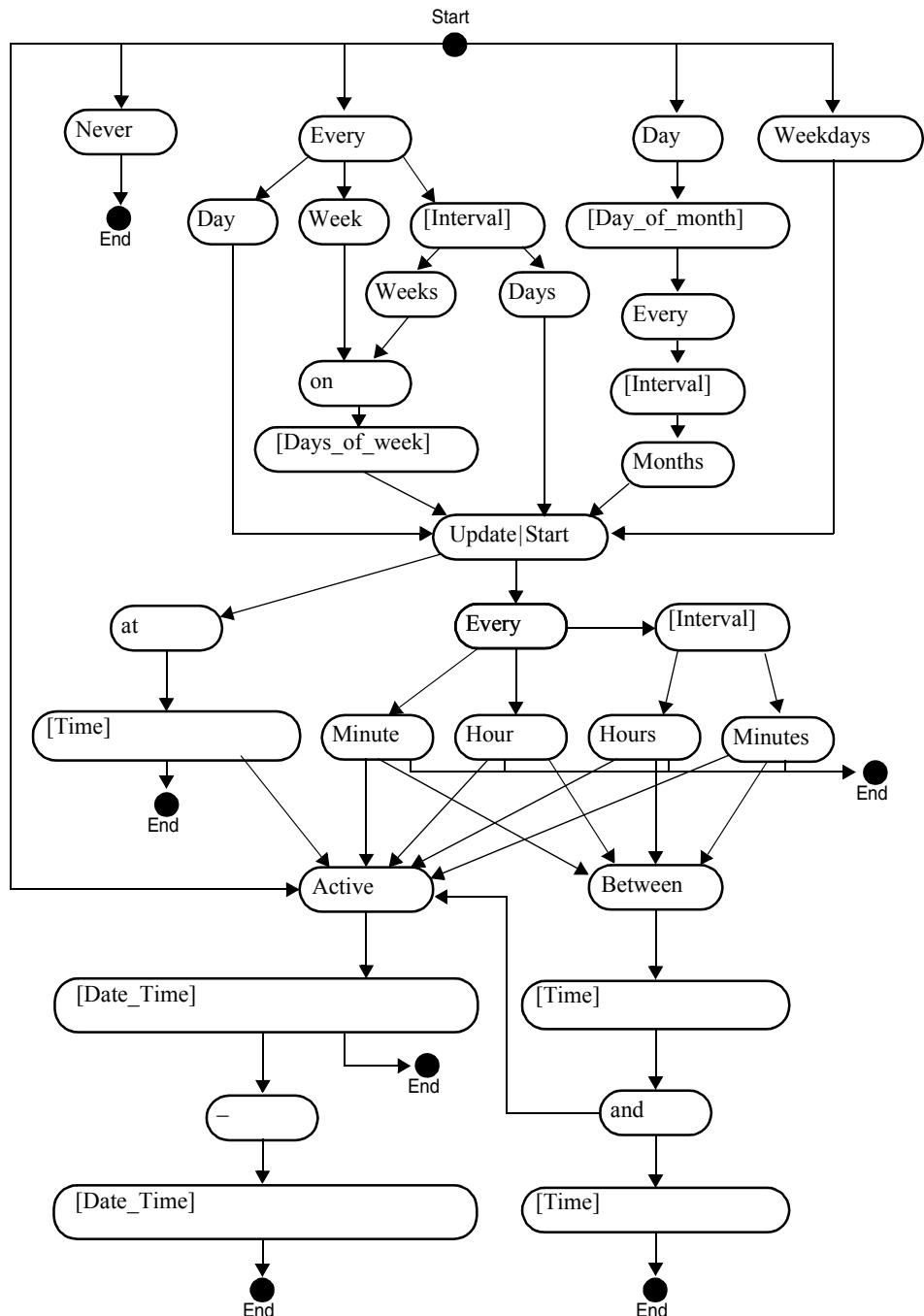
For example:

`-schedblackout "9:00AM - 5:00PM"`

## Specifying primary, secondary, update, and repair schedules

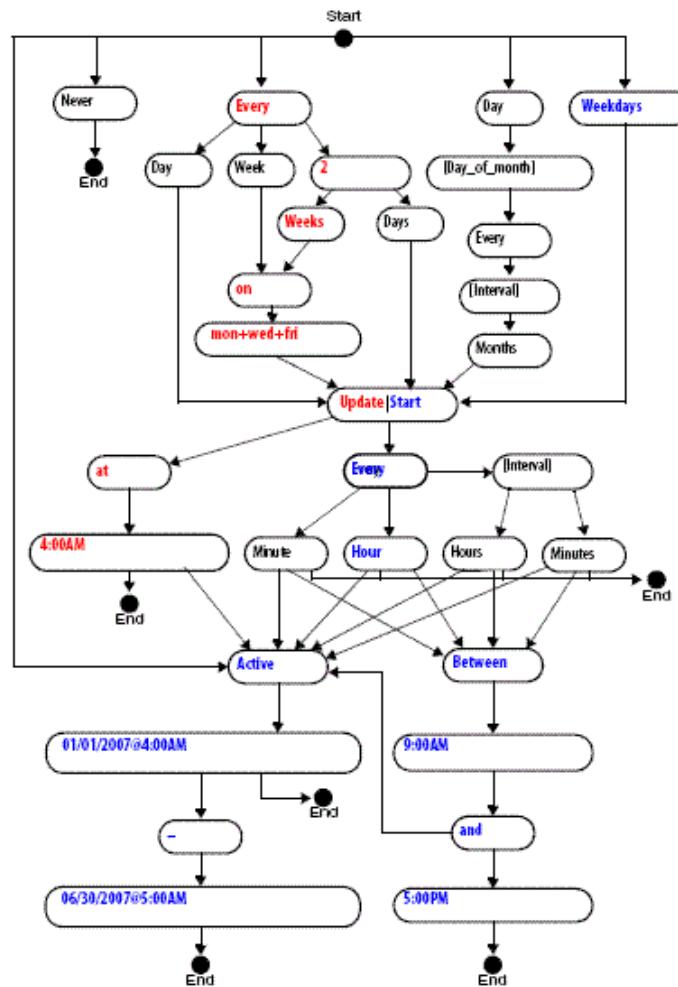
The format for the other types of schedule is less straightforward and is best described in the diagram shown in Figure A-1 on page 420. A valid package schedule is any string that can be traced from the *start* state in the diagram, through intermediate states, and terminating in an *end* state.

Figure A-1: Package schedule syntax diagram



The schedule string is not case sensitive. The following examples of package schedules are composed from the diagram. Two are detailed by color:

- never
- every 2 days update at 4:00AM
- every day update every 30 minutes between 9:00AM and 5:00PM
- weekdays start every hour between 9:00AM and 5:00PM active 01/01/2002@4:00AM - 06/30/2002@6:00PM
- active 01/01/2002@4:00AM - 06/30/2002@6:00PM
- every 2 weeks on mon+wed+fri update at 4:00AM



## Format for primary or secondary schedules

For primary and secondary schedules, specify a date and time when a state change occurs. Unlike in update or repair schedules, you do not need to specify a recurrence frequency. Use the following command-line options to specify a primary or secondary schedule:

```
-schedprimary {<package_URL>=<date_time_range>}
```

or

```
-schedsecondary {<package_URL>=<date_time_range>}
```

For example, the following command specifies a primary schedule that is active between January 1, 2002 at 4 AM and June 30, 2002 at 6 PM:

```
-schedprimary http://trans.company.com:5282/MyPackage="active 01/01/2002@4:00AM - 06/30/2002@6:00PM"
```

## Format for update or repair schedules

For update and repair schedules, you can specify a recurrence frequency. Use the following command-line options to specify an update or repair schedule:

```
-schedupdate {<package_URL>=<date_time_range_frequency>}
```

or

```
-schedverifyrepair {<package_URL>=<date_time_range_frequency>}
```

For example, the following command specifies an update schedule that updates every two weeks on Mondays and Wednesdays at 4 AM, active beginning at 5 AM on January 1, 2002:

```
-schedupdate http://trans.company.com:5282/MyPackage="every 2 weeks on mon+wed update at 4:00AM active 01/01/2002@5:00AM"
```



# B Improving Sun ONE directory LDAP performance

This appendix describes ways to improve LDAP performance by creating indexes for LDAP entries and setting LDAP parameter values affecting caching.

The following topics are provided:

- Creating indexes for LDAP entries used by BMC Configuration Automation for Clients (page 424)
- Tuning database performance (page 425)

# Creating indexes for LDAP entries used by BMC Configuration Automation for Clients

Sun Java System Directory Server (Sun One) Directory uses index files to aid in searching the directory. Indexes greatly improve the performance of searches in the directory databases. However, this improved performance comes at the price of slower database modification and creation operations. Indexes also are expensive in terms of system resources, especially disk space.

To search for entries, BMC Configuration Automation for Clients components use class attribute names corresponding to entries. During the search, the search filter uses the equality condition to compare an entry to the attributes. The attributes are shown in Table B-1 on page 424 All of these attributes can benefit from indexing to improve performance.

Table B-1: Attribute names and their object classes

Attribute name	Object class
uid	inetorgperson
cn	mrbamachine
cn member uniqueMember	groupofnames, groupofuniquenames, mrbacollection
cn mrbatargetdn mrbatargettxgroup mrbatargettxuser	mrbaSubscription
objectclass	inetorgperson, mrbamachine, groupofnames, groupofuniquenames, mrbacollection, mrbaset, mrbasubscription

Of these attributes, the `objectclass` attribute cannot be deleted. It is part of the system indexes defined by the directory service. Other attributes, such as `uid`, `cn`, `member` and `uniqueMember`, are default indexes created by the directory service. Although it is possible to delete these attributes, BMC recommends that you do not change them.

To improve Policy Service plug-in performance, you can create indexes for the following additional attributes:

- `mrbatargetdn`

- mrbatargetxgroup
- mrbatargetxuser
- mrbatargetall

► **To create indexes for these attributes, use the following procedure for each of the attributes**

- 1 In the Sun ONE Directory server console, select the Configuration tab.
- 2 In the left panel, expand the Data node and locate the suffix representing the database that you want to index (for example, dc=company, dc=com).
- 3 Expand the suffix of the database, and select the entry under it (for example, userRoot).
- 4 Click the Indexes tab on the right panel.
- 5 Click the Add attribute button at the bottom of the right panel and select the attribute that you want to index.
- 6 Check only the Equality checkbox. Verify that all others are unchecked.
- 7 Click the Save button at the bottom of the dialog box.

## Tuning database performance

The most straightforward way to influence server performance is by setting the amount of memory available to the server. Larger memory settings improve search performance.

► **To configure database parameters for increased performance**

- 1 On the directory service console, click the Configuration tab.
- 2 Click Database in the left panel to display database tabs in the right panel.
- 3 Click the Performance tab in the right panel to display the current database performance settings.
- 4 Enter the number of entries you want the server to keep in memory in the Maximum Entries in Cache text box.

**Maximum Entries in Cache** specifies the number of entries the directory service maintains in cache. Increasing this number uses more memory, but can substantially improve search performance. The actual amount of memory required per additional entry depends on the nature of the data stored in the directory service. However, as a general guideline, you can estimate that each user or machine entry maintained in cache requires approximately 1 KB (1024 B) of memory. See “Calculating maximum cache size and maximum entries in cache” on page 427 for an example of calculating the Maximum Entries in Cache setting.

- 5 Enter the amount of memory you want to make available for open index files in the Maximum Cache Size text box.

**Maximum Cache Size** specifies the size, in bytes, of the in-memory cache. Increasing this number uses more memory, but can substantially improve server performance. The improvement is especially marked during modifications or when the indexes are being built. Do not increase this number beyond the available resources for your machine. To estimate the largest cache size your system can support, see “Calculating maximum supportable cache size” on page 426. For an example of calculating desirable maximum cache size, see “Calculating maximum cache size and maximum entries in cache” on page 427.

## Calculating maximum supportable cache size

If you are creating a very large database from LDIF, set the Maximum Cache Size value as large as possible. The larger this value, the faster your database will be created. The following procedure can help you estimate the maximum cache size your system can support.

### ► To estimate the maximum cache size your system can support

- 1 Determine how much free memory you have on your system.
- 2 Divide the amount of free memory by two.
- 3 Subtract 1 MB from the result. The resulting value is a good estimate for the value of maximum cache size your system can support.

To summarize:

$$\begin{aligned} \text{(Maximum supportable cache size, in MB)} &= \\ \text{((System free memory, in MB) / 2) - (1 MB)} \end{aligned}$$

For example, if your system has 50 MB of free memory, the maximum cache size your system can support is 24 MB, calculated as follows:

$$\begin{aligned} \text{(Maximum supportable cache size, in MB)} &= \\ ((50 \text{ MB}) / 2) - (1 \text{ MB}) &= 24 \text{ MB} \end{aligned}$$

---

Note: Increasing maximum cache size improves performance when you create a large database, but may waste memory resources at other times. Therefore, when you are done creating your database, set this parameter back to a lower value before running your server in a production environment.

---

## Calculating maximum cache size and maximum entries in cache

Consider a BMC Policy Management Directory Enabled mode deployment with entries as summarized in Table B-2 on page 427 To calculate the maximum number of entries in the cache, sum the number of entries. To calculate the total required cache size, sum the cache requirements for all entry types.

Table B-2: Example entries and cache requirements

Entry type	Number of entries	Cache required per entry type	Total cache required
user	10,000	1 KB	10,000 KB = 10 MB
machine	10,000	1 KB	10,000 KB = 10 MB
group	1,000	10 KB <sup>1</sup>	10,000 KB = 10 MB
policy	1,000	10 KB <sup>1</sup>	10,000 KB = 10 MB
<b>Totals:</b>	<b>22,000</b>		<b>40 MB</b>

<sup>1</sup> The cache required per group entry assumes 100 user or machine members per group and 100 bytes per member attribute, for a total of 10 KB per group. The same assumptions apply to policy entries.

As summarized in the table, the maximum number of entries in the cache for this example totals 22,000. The desirable maximum cache size setting is the sum of cache requirements for all cache entries: 40 MB.

---

Note: Make sure the maximum cache size value you set does not exceed the maximum cache size your system can support. Calculate the maximum cache size your system can support as described in “Calculating maximum supportable cache size” on page 426.

---

## Monitoring database caching performance

If you want to monitor the database to see how efficiently it is using the cached information, you must examine the entry cache hit ratio. The entry cache hit ratio is the ratio of database cache hits to database cache tries. The closer this value is to 100%, the better.

Whenever a directory operation attempts to find a portion of the database that is not resident in the database cache, the directory service has to perform a disk access to obtain the necessary database page. Thus, a low ratio indicates that directory service performance is low because the number of disk accesses is high.

To improve this ratio, you can increase the value of the Maximum Cache Size parameter in order to increase the amount of data that the directory service maintains in the database cache. The maximum value that you can set on this parameter depends on the amount of real memory on your machine as well as the value set for the Maximum Entries in Cache entry, as described in previous sections.

### ► To see entry cache hit ratio

- 1 Click the Status tab in the directory service.
- 2 Click the Performance Counters entry in the left panel.
- 3 Click the Database tab in the right panel and observe the Summary Information.

# Appendix C

# Troubleshooting

This appendix provides suggestions for troubleshooting common problems with Policy Manager.

The following topics are provided:

- Troubleshooting Policy Manager (page 430)
- Troubleshooting system settings (page 434)

# Troubleshooting Policy Manager

This section provides suggestions for troubleshooting common problems with Policy Manager.

**Problem** Is there an easy way to copy subscription policies between different groups? For example, I have a group called `Endpoints_A` that has a load of channels targeted to it, as well as tuner properties. I want to copy this information to additional groups such as `Endpoints_B` and `Endpoints_C`.

The recommended maximum number of computers in an Active Directory group is about 3,500. I need to upgrade over 40,000 endpoints, so I want to create multiple groups to perform this upgrade. Can I copy a policy from one group to another to avoid creating multiple identical policies from scratch?

**Solution A** In the lastest versions of Policy Manager, you can copy a policy from one target to another using the copy icon. See “Copying policies” on page 253 for more information about this procedure.

**Solution B** If your version does not include the copy option, you can get around the 3500 object limitation by migrating the policy to a master group of groups, then making each group a member of the master group. See “Creating a group of groups” on page 63 for more information about this procedure.

---

**WARNING:** Before you add subgroups to the master group, make sure to clear out any channel update schedules in the sub-groups to avoid canceling out. This is especially true of service channels such as policy service and inventory service.

---

**Solution C** You can also use Report Center to locate machines in specific locations based on IP address ranges or subnets. Then use Policy Management to create collections from these reports, and target the collections.

**Problem** I am troubleshooting a problem with an endpoint user’s machine. However, when I log in as an administrator to the user’s machine, the user’s channels get deleted. What can I do to prevent the user’s channels from getting deleted?

Possible solution	If the endpoint user's machine has Policy Service 5.0.1 or higher, you can set a tuner property to specify the administrators who you want to be able to log in to a user's machines temporarily when using user-based targeting (possibly for troubleshooting). The property prevents the user's channels from getting deleted when you log in as an administrator. You can use Policy Manager to set the following property:  <code>marimba.subscription.adminusers=&lt;user1&gt;,&lt;user2&gt;,...</code> where <code>&lt;user1&gt;,&lt;user2&gt;,...</code> specifies a comma-delimited list of administrators. The list of administrators can contain the name of any user who can log in at the endpoint machines.  When the administrator logs in, the following happens: <ul style="list-style-type: none"><li>■ Channels subscribed for another user (the end users) will not be removed.</li><li>■ If there are channels that have been assigned to an administrator, these channels will be delivered to the endpoint.</li><li>■ The properties that have been set for the administrator will be set. Possibly, the properties that have been set for the end users will be overwritten by those set for the administrator. However, when the end users log back in and the Policy Service updates, the end users will get their properties back.</li></ul>
Problem	When I browse an organizational unit or container with many members, nothing seems to happen, or I receive an error message.
Possible solution	When you browse an organizational unit or container, Policy Manager sends a search request to the directory service and waits a maximum of one minute limit for a reply. The wait time is not configurable. Therefore, if the organizational unit or container you are browsing contains many members, your directory service may be exceeding the timeout value.  To work around the problem, use the search box to narrow your directory service search to some subset of the members.
Problem	Policy Manager doesn't show me all of my targets in the Target View page. I'm using Sun ONE Directory.
Possible solution	You may need to increase the <i>look-through limit</i> in Sun ONE Directory, or set up the Virtual List View. See the <i>BMC Marimba Client Automation Installation Guide</i> , available on the BMC Customer Support website.
Problem	I can't expand some of my target groups in Policy Manager Target View. I'm using Active Directory.

Possible solution	If you're using Active Directory in a multidomain configuration, the groups you're trying to expand are probably in a domain different from the one in which Policy Manager is running. You cannot expand groups in other domains. Only groups that are in the same domain as the one you specified in the System Settings are expandable.
<b>Problem</b>	I want to target a container (organizational unit) with Policy Manager, but am not allowed to do so.
Possible solution	Starting in Policy Management 5.1, you can target containers and organizational units.
<b>Problem</b>	I switched from using a directory service to sourcing users and user groups from the transmitter. However, in the GUI, I still see the policies that I created when I was sourcing user information from the directory service. Shouldn't those policies disappear?
Possible solution	Most users don't switch configurations this way. What you've done is pretty rare and usually happens only if you changed your mind about how you want to obtain user information after you installed the product.  To fix the problem, delete the old policies from the command line. For more information about using the <code>-delete</code> command, see "Command-line basics" on page 386.
<b>Problem</b>	Sometimes when I expand a group in Policy Manager, I don't see any members, even when I know the group isn't empty.
Possible solution	Groups are represented by containers in the directory service. If you don't have permission to read the container in the directory service, no members of the group will be displayed in Policy Manager.  Make sure you (or the account of the user you logged in as) have the correct permissions in the directory service for the groups you want to expand. See "Setting up user accounts" on page 65.
<b>Problem</b>	When I remove a tuner or package property from the list in the Tuner and Package Properties page, then save the policy, the property is not removed from the endpoint.
Possible solution	Deleting a property from the Properties text box does not remove it from the target endpoints. This causes the property to remain as is on the target tuner (not managed by Policy Manager). You remove a tuner or package property from the target endpoints by setting its value to <code>&lt;null&gt;</code> , including the angle brackets ( <code>&lt;&gt;</code> ), or leaving the value blank, as in <code>&lt;no value&gt;</code> .

When you delete a tuner property using Policy Manager, it is deleted from the prefs.txt file in the tuner's workspace directory and usually it will no longer take effect. However, if you previously had set the property when packaging the tuner, it cannot be deleted through Policy Manager. This is because setting the property during packaging sets the property in the properties.txt file located in the tuner's workspace directory (typically, this is C:\Program Files\Marimba\Tuner\.marimba\Marimba\properties.txt on Windows), and Policy Manager will not remove the property from the properties.txt file. Usually, the tuner properties in the prefs.txt file override the default tuner properties in the properties.txt file. In this case, however, the tuner property only exists in the properties.txt file, so it will take effect.

**Problem** I'm not always getting the results I expect when I set tuner or package properties with Policy Manager.

**Possible solution** You might experience conflicts with other administrators if you edit policies at the same time. In that case:

Assume that UserA and UserB independently edit properties for a policy called Policy ManagementX. Before UserA commits his changes to the properties, UserB edits and saves properties for the same target. In this scenario, UserB saves her edits to Policy ManagementX before UserA does—UserA is the last to save.

**Case 1:** UserA adds a new property that was not present in Policy ManagementX and is not added by UserB.

Result: The property is added to Policy ManagementX.

**Case 2:** UserA adds a new property that was not present in Policy ManagementX but is also added, with a different value, by UserB.

Result: The value assigned by UserA is saved, overwriting the value assigned by UserB.

**Case 3:** UserA removes a property from Policy ManagementX, and the property is not modified by UserB.

Result: The property is removed from Policy ManagementX.

**Case 4:** UserA removes a property from Policy ManagementX, but the property is modified by UserB.

Result: The property is removed from Policy ManagementX.

**Case 5:** UserA does not modify a property, but the property is modified or removed by UserB.

Result: The property is modified or removed as specified by UserB.

**Case 6:** UserA modifies a property that is also modified or removed by UserB.

Result: UserA's modified property is saved and UserB's modification is lost.

**Problem** I am having a problem connecting to ADAM / AD LDS. When I try to add it as a data source, I get an error message saying that the directory service might be down or the credentials are invalid, even though I am sure the password I entered is correct. What is wrong?

**Possible solution** The password set up for the user account might not be valid. The domain where ADAM / AD LDS is running might have certain password restrictions that the password being used does not meet. For example, on Windows 2003 Server, the complex password restriction—passwords must include six or more characters and at least one punctuation symbol—is enabled by default. If the password being used is not valid (that is, it does not meet the restriction), then the user account that you create will be disabled.

You can solve this problem by using one of these options:

- Take the machine off the domain and reset the `msDS-UserAccountDisabled` to false.
- Reset the password to a valid password and reset the `msDS-UserAccountDisabled` to false.

**Note:** You must reset the `msDS-UserAccountDisabled` attribute to false. Just taking the machine off the domain will not solve the problem.

## Troubleshooting system settings

This section provides suggestions for troubleshooting common problems with the system settings.

**Problem** I can't log in to Report Center. I get an error message that says, "Unable to log in. Please contact your primary administrator, or check the following...."

Possible solutions The most common problem when logging in is incorrectly typing the user name, the password, or both. Try entering them again. If you have made sure that you are entering the correct user name and password but still cannot log in, check the items described in this section.

**Note:** Some solutions might require access to pages that are reserved for primary administrators. If you don't have access to some pages, contact your primary administrator.

You may also need to log in with the emergency user name, which is `admin`, and its password. By default, no password is set, but it is recommended that you do set an emergency password.

If you have problems logging in, first, verify that the user name is either in the local user database or in the directory service, whichever you are using to authenticate users:

- a Find out which data source you are using by going to the User Authentication Type page (System Settings > User Authentication > User Authentication Type).

Either the Directory Service option or the Local User Database option is selected.

- b Do one of the following:

- If the Local User Database option is selected: Verify that the user is in the local user database by going to the Local User Database page (System Settings > User Authentication > Local User Database).
- If the Directory Service option is selected: Verify that the user is in the directory service by using your directory service administration tool.

**Note:** Directory service users must belong to a group, and the group must be mapped to a user role in the User Roles page (System Settings > User Authentication > User Roles).

If a directory service is being used for user authentication, you might also want to check the following possible causes:

- No directory service has been specified or your directory service settings are incorrect.

Check the directory service settings by going to the Directory Services page (System Settings > Data Source > Directory Services). Make sure that all the fields are filled in with the correct information.

If you change the directory service settings, any users who are currently logged in will be asked to log in again. You might want to warn users who are currently using the system before you change the directory service settings.

- The directory service is offline or not available.

Check that the directory service is running and that Common Management Services (CMS) is able to connect to it.

- Directory service groups have not been mapped to user roles.

Check that directory service groups have been mapped to user roles by going to the User Roles page (System Settings > User Authentication > User Roles). Make sure that the directory service groups you want to have access are specified for one of the user roles. To specify multiple groups, enter a comma-separated list of groups for each role.

<b>Problem</b>	When I try to enter the URL for accessing Report Center ( <code>http://&lt;machine_name&gt;:8888</code> ), another browser application appears. If another application is already using the default port 8888, how do I change the port number for Report Center?
----------------	---

<b>Possible solution</b>	The Common Management Services (CMS) channel, which in turn runs Report Center, first attempts to use port 8888 by default. If CMS finds that port is already in use, it will use port 8889. If that port is in use, CMS will use port 8890, and so on. Keeping this behavior in mind, you can use the following solutions:
--------------------------	---

- If you are attempting to access Report Center remotely, from another machine, you can try using port 8889.
- If you are on the machine that hosts Report Center, you can use the Common Management Services command-line option called `-getPort` to find out which port number is being used, such as `runchannel http://trans.company.com/cms -getPort`, and then you can enter the URL with that port number.
- If you are on the machine that hosts Report Center, you can use the Common Management Services command-line option called `-setPort` to change the port number, such as `runchannel http://trans.company.com/cms -setPort 8880`, and then you can enter the URL with that port number.

<b>Problem</b>	I stopped working in Report Center to do something else. When I returned and clicked something, the login page appeared. What happened?
----------------	---

---

Possible solution	You were automatically logged out. Users who have been idle or inactive for a specified number of minutes are automatically logged out to help to prevent unauthorized access to the applications. You can change this setting on the User Timeout page (System Settings > General > User Timeout). The default user timeout is 60 minutes. You can disable the user timeout by setting it to 0 minutes.
<b>Problem</b>	I changed the number of minutes specified on the User Timeout page, but the new user timeout I specified doesn't seem to be taking effect.
Possible solution	If you change the User Timeout setting, the new timeout will not apply to users who are already logged in. The new timeout will take effect the next time users log in. You must log out and log in again for the new setting to take effect.
<b>Problem</b>	I can't find the log files. Where are they?
Possible solution	Log files for applications are stored in the tuner's workspace directory. You can find out the exact location of the log files by looking in the Log Files page (System Settings > General > Log Files).



# D Removing Policy Management entries from the directory service

This appendix describes how to remove Policy Management entries from the directory service. You might want to do this if you no longer want to store policies in the directory service. You will need to remove the group of containers that Policy Manager added to the directory service when you ran the installation script.

The following topics are provided:

- [Removing Policy Management entries from directory services \(page 440\)](#)

# Removing Policy Management entries from directory services

Follow the instructions in this section to remove Policy Management entries from Active Directory, ADAM / AD LDS, or Sun ONE Directory. You can use a graphical user interface, such as the MMC snap-in called Active Directory Users and Groups, ADAM ADSI Edit, or the Sun ONE Directory server console, to delete the Policy Manager entries.

Be sure to stop Policy Manager before you start removing entries from the directory service.

---

**WARNING:** Only remove entries created for use by Policy Manager and Report Center if you no longer want to use the directory service with Policy Manager. Any policies that you previously created will be deleted when you remove these entries.

---

You will remove the following containers:

■ **ACL container**

Example:

`ou=Acl`

This container stores any ACLs and permissions that you assigned using Policy Manager.

■ **Collections container**

Examples:

`ou=Collections`

This container stores any collections that you created using Report Center. You specify the path using the BMC Marimba Client Automation configuration object property `marimba.ldap/browse.collectionbase`.

■ **BMC CM container**

Examples:

Active Directory: `cn=BMC CM,ou=BMC Software,dc=company,dc=com`

Sun ONE Directory: `ou=BMC CM,ou=BMC Software,dc=company,dc=com`

The child containers of the BMC CM container store collections, computers, subscriptions (policies), ACL objects, and configuration objects. The Subscription configuration object is located in the ConfigObjects container.

**■ Subscriptions container**

Example:

ou=Subscriptions,ou=ConfigObjects,ou=BMC CM,ou=BMC Software,dc=company,dc=com

This container stores the policies you have saved. You specify the path using the Suscription configuration object property marimba.subscriptionplugin.subscriptionbase.

**► To remove Policy Management entries from active directory**

- 1 Open the Windows Start menu and choose Programs > Administrative Tools > Active Directory Users and Computers.
- 2 Select and delete the following containers:
  - Acl
  - Collections
  - BMC CM
  - Subscriptions
- 3 Save your work and close the console.

---

Note: Active Directory does not allow the removal of schema definitions. As a result, schema items added by the Policy Manager installation script cannot be removed from the Active Directory schema.

---

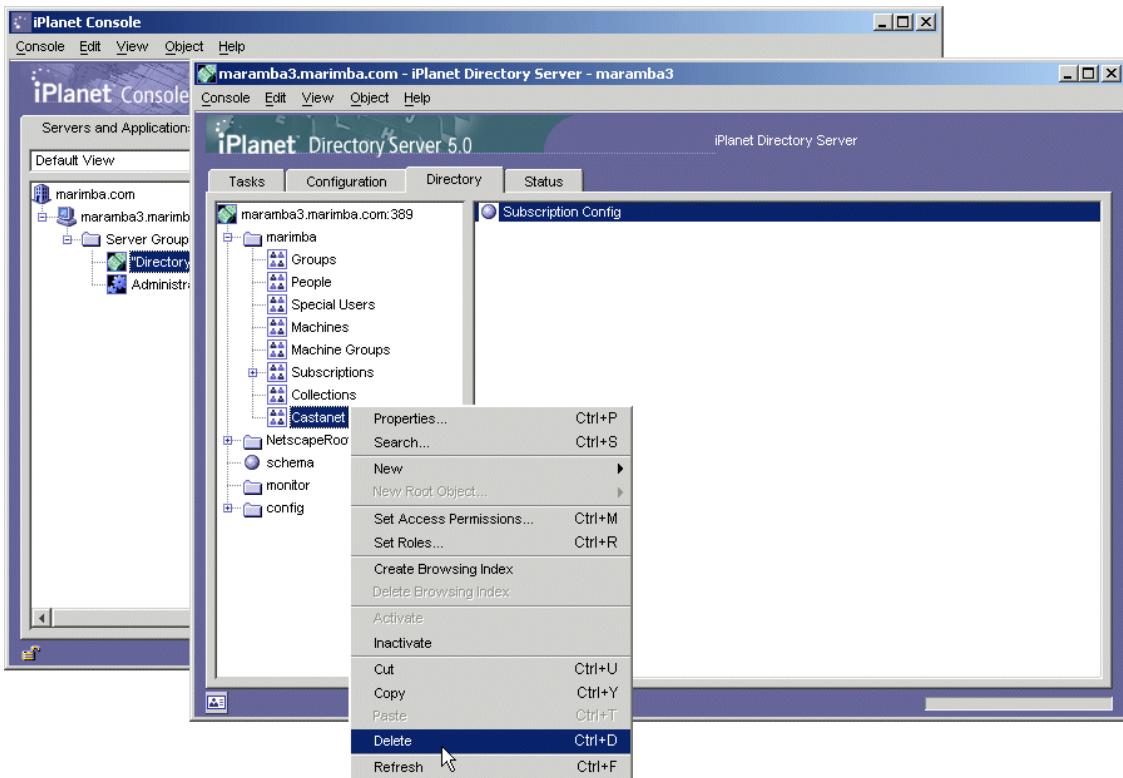
**► To remove Policy Management entries from ADAM**

- 1 Open the Windows Start menu and choose Programs > ADAM > ADAM ADSI Edit for Windows 2003 and choose Administrator Tools > ADSI Edit for Windows 2008.
- 2 Select and delete the following containers:
  - Acl
  - Collections
  - BMC CM
  - Subscriptions
- 3 Save your work and close the console.

► To remove Policy Management entries from Sun ONE directory

- 1 Start the Sun ONE Directory console and navigate to the directory service that you have configured Policy Manager to use.
- 2 Click Open, and select the Directory tab. Expand the directory tree to list the directory service groups.
- 3 Select and delete the following containers:
  - Acl
  - Collections
  - BMC CM
  - Subscriptions

Figure D-1: Deleting subscription entries from Sun ONE directory



**Note:** If you used the command-line option `-machines` to import machine names into Sun ONE Directory, you can remove the machines as well. The machines are stored in the path you specify using the attribute `marimba.subscriptionplugin.machineimportbase` in the Subscription configuration object (example path: `ou=Machine Groups,dc=company,dc=com`).

**Important:** Do not delete the Machines container unless you are sure that only Policy Manager-created machine entries are inside.

- 4 Save your work and close the console.



Appendix

# E State mappings for policy compliance

This appendix shows state mappings that determine the policy compliance category for packages.

# Mapping of package states for policy compliance

The following table shows how policy compliance assigns a category to packages, based on:

- The package state the policy specifies
- The package state scanner service reports at the endpoint

Table E-1: Mapping of package states for policy compliance

Policy package state	Endpoint package state	Policy compliance category
available	aborted	non-compliant
	available	compliant
	checking	compliant
	failed	non-compliant
	install-pending	compliant
	installed	compliant
	removed	non-compliant
	rolled back	non-compliant
	running	compliant
	subscribed	compliant
	subscribing	non-compliant
	uninstalled	non-compliant
	unsubscribed	non-compliant
	updating	compliant
delete	aborted	non-compliant
	available	non-compliant
	checking	non-compliant
	failed	non-compliant
	install-pending	non-compliant
	installed	non-compliant
	removed	compliant
	rolled back	non-compliant

Table E-1: Mapping of package states for policy compliance (Continued)

Policy package state	Endpoint package state	Policy compliance category
	running	non-compliant
	subscribed	non-compliant
	subscribing	non-compliant
	uninstalled	compliant
	unsubscribed	non-compliant
	updating	non-compliant
exclude	aborted	non-compliant
	available	non-compliant
	checking	non-compliant
	failed	non-compliant
	install-pending	non-compliant
	installed	non-compliant
	removed	compliant
	rolled back	non-compliant
	running	non-compliant
	subscribed	non-compliant
	subscribing	non-compliant
	uninstalled	compliant
	unsubscribed	compliant
	updating	non-compliant
inactive	aborted	non-compliant
	available	compliant
	checking	non-compliant
	failed	non-compliant
	install-pending	non-compliant
	installed	non-compliant
inactive (Continued)	removed	compliant
	rolled back	non-compliant

Table E-1: Mapping of package states for policy compliance (Continued)

Policy package state	Endpoint package state	Policy compliance category
	running	non-compliant
	subscribed	non-compliant
	subscribing	non-compliant
	uninstalled	compliant
	unsubscribed	non-compliant
	updating	non-compliant
none	aborted	non-compliant
	available	non-compliant
	checking	non-compliant
	failed	non-compliant
	install-pending	non-compliant
	installed	non-compliant
	removed	non-compliant
	rolled back	non-compliant
	running	non-compliant
	subscribed	non-compliant
	subscribing	non-compliant
	uninstalled	non-compliant
	unsubscribed	non-compliant
	updating	non-compliant
primary	aborted	non-compliant
	available	non-compliant
	checking	non-compliant
	failed	non-compliant
	install-pending	non-compliant
	installed	compliant
	removed	non-compliant
	rolled back	non-compliant

Table E-1: Mapping of package states for policy compliance (Continued)

Policy package state	Endpoint package state	Policy compliance category
	running	compliant
	subscribed	compliant
	subscribing	non-compliant
primary (Continued)	uninstalled	non-compliant
	unsubscribed	non-compliant
	updating	compliant
start_persist	aborted	non-compliant
	available	non-compliant
	checking	compliant
	failed	non-compliant
	install-pending	compliant
	installed	compliant
	removed	non-compliant
	rolled back	non-compliant
	running	compliant
	subscribed	compliant
	subscribing	non-compliant
	uninstalled	non-compliant
	unsubscribed	non-compliant
	updating	compliant
subscribe	aborted	non-compliant
	available	non-compliant
	checking	compliant
	failed	non-compliant
	install-pending	non-compliant
	installed	compliant
	removed	non-compliant
	rolled back	non-compliant

Table E-1: Mapping of package states for policy compliance (Continued)

Policy package state	Endpoint package state	Policy compliance category
	running	compliant
	subscribed	compliant
	subscribing	non-compliant
	uninstalled	non-compliant
	unsubscribed	non-compliant
	updating	compliant
subscribe_noinstall	aborted	non-compliant
	available	non-compliant
	checking	compliant
subscribe_noinstall <i>(Continued)</i>	failed	non-compliant
	install-pending	compliant
	installed	non-compliant
	removed	non-compliant
	rolled back	non-compliant
	running	compliant
	subscribed	compliant
	subscribing	non-compliant
	uninstalled	non-compliant
	unsubscribed	non-compliant
	updating	compliant
subscribe_persist	aborted	non-compliant
	available	non-compliant
	checking	compliant
	failed	non-compliant
	install-pending	compliant
	installed	compliant
	removed	non-compliant

Table E-1: Mapping of package states for policy compliance (Continued)

Policy package state	Endpoint package state	Policy compliance category
	rolled back	non-compliant
	running	compliant
	subscribed	compliant
	subscribing	non-compliant
	uninstalled	non-compliant
	unsubscribed	non-compliant
	updating	compliant
subscribe_start	aborted	non-compliant
	available	non-compliant
	checking	non-compliant
	failed	non-compliant
	install-pending	non-compliant
	installed	compliant
	removed	non-compliant
	rolled back	non-compliant
subscribe_start <i>(Continued)</i>	running	compliant
	subscribed	compliant
	subscribing	non-compliant
	uninstalled	non-compliant
	unsubscribed	non-compliant
	updating	compliant
update	aborted	non-compliant
	available	non-compliant
	checking	non-compliant
	failed	non-compliant
	install-pending	non-compliant

Table E-1: Mapping of package states for policy compliance (Continued)

Policy package state	Endpoint package state	Policy compliance category
	installed	compliant
	removed	non-compliant
	rolled back	non-compliant
	running	compliant
	subscribed	compliant
	subscribing	non-compliant
	uninstalled	non-compliant
	unsubscribed	non-compliant
	updating	compliant
verify_repair	aborted	non-compliant
	available	non-compliant
	checking	non-compliant
	failed	non-compliant
	install-pending	non-compliant
	installed	compliant
	removed	non-compliant
	rolled back	non-compliant
	running	compliant
	subscribed	compliant
	subscribing	non-compliant
verify_repair (Continued)	uninstalled	non-compliant
	unsubscribed	non-compliant
	updating	compliant

The aborted state includes the following sub-states:

- aborted (install)
- aborted (post-install)
- aborted (post-repair)

- aborted (post-rollback)
- aborted (post-uninstall)
- aborted (post-update)
- aborted (post-verify)
- aborted (pre-install)
- aborted (pre-repair)
- aborted (pre-rollback)
- aborted (pre-uninstall)
- aborted (pre-update)
- aborted (pre-verify)
- aborted (repair)
- aborted (rollback)
- aborted (uninstall)
- aborted (verify)

The failed state includes the following sub-states:

- failed (install)
- failed (post-install)
- failed (post-launch)
- failed (post-minorupdate)
- failed (post-repair)
- failed (post-uninstall)
- failed (post-update)
- failed (post-verify)
- failed (pre-install)
- failed (pre-launch)
- failed (pre-minorupdate)
- failed (pre-repair)
- failed (pre-uninstall)

- failed (pre-update)
- failed (pre-verify)
- failed (repair)
- failed (uninstall)
- failed (verify)

# Glossary

## ADAM

Active Directory Application Mode, a lightweight directory service supported by BMC Marimba Client Automation .

## Application Installer

A component that appears when a user subscribes to a channel created using Application Packager. It installs the application or content files comprising the channel.

## Application Packager

A BMC Marimba Client Automation application used to package software application (and subsequent updates) into a format that you can then distribute to target endpoints, such as desktops or servers. It provides an interface to a family of packaging components, including Packager for Shrinkwrap Windows Applications, Windows Installer Packager, Custom Application Packager, File Packager, and Java Packager.

## assembly

The primary building block of .NET framework applications; a collection of functionality that is built, versioned, and deployed as a single implementation unit. Assembly files usually have the file extension .netmodule, and can be either static or dynamic.

## byte-level differencing

A BMC Marimba Client Automation feature that allows updates to be performed easily, efficiently, and even automatically. During updates, byte-level differencing means that you don't need to transfer entire files. Only the new or changed bytes are downloaded during updates.

## certificate

See *security certificate*.

## certificate authority

An agency from which security certificates can be obtained, either directly or through Certificate Manager.

**Certificate Manager**

A BMC Marimba Client Automation application that can be used to obtain security certificates from a certificate authority and to install and manage those certificates.

**channel**

A format used to publish content to a transmitter and download by a tuner to an endpoint. Channels contain all the information necessary for installation on endpoints. A channel can be:

- An application of any type (Windows, Java, and so on) and optionally related data files
- One or more content files, containing HTML or any data
- A combination of the above

**channel category**

Any of a number of named sets into which channels are organized in a tuner's channel list. For each category, the channel list shows a colored bar containing the category title.

**Channel Copier**

A BMC Marimba Client Automation application that can be used to copy channels (and information about them, such as channel properties) from a transmitter, CAR file, or package directory to a transmitter or CAR file. Using Channel Copier to copy a channel to a transmitter constitutes publishing the channel.

**channel directory**

See *package directory*.

**channel index**

A representation of a channel's file structure, including each file's contents along with other information, such as the file's checksum. The transmitter caches channel indexes for all channels published to it, to optimize its handling of requests for channel updates.

**Channel Manager**

The channel that provides the user interface to the standard tuner; the tuner's default primary channel.

**channel parameter**

Information for installing and launching a packaged application. The information in this file is used by the tuner when you run a channel to install and launch the packaged application. See also *parameters.txt*.

**channel property**

One of many characteristics of a channel that provide information about the channel, such as who created or published it, what its URL is, when it was last updated, and when the next update is scheduled. Through the tuner, a user can view channel properties and set some of them; additional channel properties can be set by an administrator or a channel publisher. See also *channel parameter* and *tuner property*.

**channel segment**

See *segment (of a channel)*.

**channel-signing certificate**

A security certificate, indicating who published a particular channel, that is installed in the tuner from which the channel is being published and is used to digitally sign the channel when it is published.

**channel update**

The synchronization of a tuner's subscribed channel with new channel data from the transmitter, made by the tuner either automatically according to an update schedule or specifically at the user's request.

**channel URL**

A channel's uniform resource locator, which is assigned when the channel is published and provides a unique identifier for that channel on the transmitter. By specifying the channel URL, a user can subscribe to the channel.

**checkpoint restart**

A BMC Marimba Client Automation feature that enables the downloading of a channel (or channel update) to continue seamlessly when the tuner restarts after an interruption, as might be caused by a power outage or the loss of a network connection.

**child containers**

Organizational objects in the CMS like folders or directories, that can be created on multiple levels in tree format to hold data about organizations and organizational units such as individual users, machines, groups of users or machines.

**CMDB**

Configuration Management Database, both an ITIL standard protocol and a BMC solution.

**CMS**

Common Management Services, the central user interface for BMC Marimba Client Automation applications and solutions.

**collections**

Dynamic groups of machines that result from posing Report Center queries, and used as targets of iterative policies. See “What is a collection?” on page 161.

**command line**

An interface provided by most products that enables commands to be entered, either at a command prompt or in a script or batch file; useful for automating the entry of complex commands.

**containers**

Organizational objects in the CMS like folders or directories, that can be created on multiple levels in tree format to hold data about organizations and organizational units such as individual users, machines, groups of users or machines.

**Content Replicator**

Content Replicator is a BMC Marimba Client Automation component that does the work of taking data files from a source system, sending the content to a data server (that is, a transmitter), and then previewing, staging, installing, or rolling back the content on destination machines. Content Replicator has the following capabilities, among others:

- Takes directories from a source system on one platform and publishes them in a format that allows for deployment on Windows NT, Windows 2000, and Linux operating systems
- Downloads and activates content and applications on one or more target servers

- Can install applications and data files regardless of whether some files are locked on the target server; can also unlock shared network resources
- Allows you to control the rate at which content is deployed, to accommodate bandwidth connections that vary, from high-speed backbone access to fractional T1 speeds

#### Custom Application Packager

An Application Packager component that allows you to package any application, especially ones developed by companies for internal use.

#### dependency

A requirement that you want to check at the endpoint before downloading and installing a channel. For example, you may want to require that a certain file or a certain channel exists at the endpoint. If dependencies cannot be validated, the channel is not downloaded, installation of the channel fails, and, depending on how the channel is configured, the user at the endpoint is informed of the failure.

#### directory services

Repositories for data about network entities such as applications, files, users, printers, and so on. Policy Manager supports directory services such as ADAM / AD LDS, Active Directory, and SunONE.

#### endpoints

The objects of a service such as policy or inventory, including tuners that receive distributions, or from which inventory data is collected. See “What is the All Endpoints target?” on page 160.

#### File Packager

An Application Packager component that allows you to package a collection of files (for example, spreadsheets, HTML files, or templates).

#### file reference

An Application Packager feature that allows you to publish the contents of a file directly to the transmitter instead of adding it to the package directory first. Using file references allows you to save disk storage space and to update files without having to replace them manually.

#### forms

Pre-packaged queries that can include variable parameters.

#### global assembly cache (GAC)

For .NET framework applications, a machine-wide code cache located on each computer where the common language runtime is installed. The global assembly cache stores assemblies specifically designated to be shared by several applications on the computer. See also *assembly*.

#### history entry

An annotation in the package that allows you to keep a record or history of the changes to a channel. A history entry might be especially useful if multiple users are editing a package. The following fields are available for each history entry:

- Timestamp
- Author
- E-mail
- Phone
- Subject
- Notes

**ITIL**

IT Infrastructure Library, the authority on best practices for quality IT service, available in the public domain.

**Java Packager**

An Application Packager component that allows you to package a Java application and set options, such as using a different Java virtual machine (JVM).

**just-in-time (JIT) application deployment**

An operation for ensuring that an application is updated or repaired before it is used. Using the Package Editor, you can configure the Windows shortcut used to start an application, so that before the application starts, one of the following operations takes place:

- The application gets updated.
- The application gets verified, and if necessary, repaired.
- The application gets both updated and repaired.

These operations are performed before the application is started, whenever the shortcut is used. Using this feature allows you to seamlessly update and repair an application, without requiring users at the endpoints to manually perform any operations, other than double-clicking the shortcut.

**macro**

a variable that is resolved at the end point

An Application Packager feature that allows you to customize and replace items in a channel, such as a file name or directory path. There are some predefined macros included with Application Packager, and you can also create user-defined macros.

**major update**

An update where the contents of the channel are changed. A major update means files, registries, environment variables, or text modifiers have been added, changed, or removed from the channel. For files, content changes would include any changes in file attributes also.

**manifest file**

A file that contains the code and instructions required to install and update an application packaged as a channel.

**master transmitter**

The transmitter whose channels a repeater or mirror transmitter replicates.

**merge modules**

Pre-compiled bundles of components (files, registry entries, and other modules) that enable developers to easily add third-party features to a Microsoft installation (MSI). See also *MSI package*.

**metabase**

A structure similar to the registry that Microsoft's Internet Information Server (IIS) uses to store information. Although it resembles the registry, the metabase is more sophisticated and has optimized information retrieval capabilities. Like the registry, the IIS metabase contains key nodes and value nodes.

**Microsoft installation (MSI) package**

This consists of at least one file with the .msi extension, which contains an installation database, a summary information stream, and data streams for various parts of the installation. An MSI package can also contain one or more transforms, internal source files, and external source files or cabinet files required by the installation. See also *Windows Installer*.

**minor update**

An update where the contents of the channel are not changed. A minor update means that the channel changed, but there are no new, deleted, or changed files, registry values, or environment variables.

**mirror**

To replicate channels from a master transmitter to another transmitter for the purpose of balancing the tuner request load; see *mirror transmitter*.

**mirror transmitter**

A transmitter that replicates channels from a master transmitter for the purpose of balancing the tuner request load, but that (unlike a repeater) doesn't have tuner requests redirected to it from the master transmitter. Users' tuners subscribe to channels directly on the mirror transmitter.

**MSI package**

See *Microsoft installation (MSI) package*.

**.NET Packager**

A component of Application Packager that allows you to package applications that have been created using Microsoft's .NET framework. The .NET framework is Microsoft's platform for building, deploying, and running Web services and applications.

**organizations**

Organizational objects in the CMS like folders or directories that can be created on multiple levels in tree format to hold data about organizations such as companies or enterprises, or departments within an enterprise that you want to separate for organizational purposes. See also *containers*, *child containers*, and *organizational units*.

**organizational units**

Organizational objects in the CMS like folders or directories that can be created on multiple levels in tree format to hold data about parts of an organization such as departments, individual users, machines, groups of users or machines. See also *containers*, *child containers*, and *organizations*.

**package**

(n.) Any data to distribute; typically an application to install on endpoints.

(v.) To prepare some kinds of applications and content for publishing as a channel, as required by the transmitter. Java application or applet channels and HTML content don't require packaging before they can be published.

See also *Application Packager* and *channel*.

**package directory**

The directory in which Application Packager stores the files for a packaged application. A package directory is created when you use any of the packager components of Application Packager to package a software application. It is also sometimes referred to as the channel directory or publish directory.

Channel directory is also sometimes used to refer to the subdirectory of the tuner's workspace directory in which the files for a particular channel are stored.

**Package Editor**

A component of Application Packager that allows you to add, edit, and remove files, directories, and registry entries in packaged applications. You can also use the Package Editor to set macros, customize environment settings, add installation scripts, and so on.

**Packager for Shrinkwrap Windows Applications**

An Application Packager component that allows you to package any commercial 32-bit Windows software application.

**parameters.txt**

A property file that contains channel parameters and is located in the package directory for that channel. It stores information for installing and launching a packaged application. The information in this file is used by the tuner when you run a channel to install and launch the packaged application.

**patch files**

Storage files for MSI packages that contain at least one database transform that adds patching information to the database of its target installation. They typically have the .msp file extension. Patches can be applied to more than one application or can upgrade an application into another application or version.

See also *Microsoft installation (MSI) package*.

**PDA Packager**

An Application Packager component that allows you to package either directories of files or PDA (personal digital assistant) applications in the form of CAB files.

policy (for installation, update, uninstallation, verify, and repair)

An Application Packager feature that determines whether files and the other contents of a channel are installed, updated, uninstalled, verified, or repaired. You can set policies at two levels:

- You can set policies as the default for all the contents of a channel.
- You can also override the default policies and set policies for individual items in a channel.

**Policy Management**

Previously called Subscription, a set of components that enables channels to be assigned to users or machines through a centrally located policy. See *Policy Manager*.

**Policy Manager**

Previously called Subscription Policy Manager, an application found in the CMS console that an administrator runs to assign channels to users, machines, or groups of users or machines as part of a policy.

**Policy Plug-In**

A middle-tier component hosted by the Transmitter and used by Policy Manager/Service.

**Policy Service**

A middle-tier component hosted by the Transmitter and used by Policy Manager/Service.

**prefs.txt**

A property file in the tuner's workspace directory in which tuner properties can be set when the tuner is installed, or in some cases after installation.

**properties.txt**

A property file in any of several locations applying to channels, tuners, or transmitters (for example, in a package directory, the tuner's workspace directory, or the transmitter's workspace directory).

**property**

See *channel property* or *tuner property*.

**property file**

A file containing key-value pairs that define characteristics of a channel, tuner, or transmitter, depending on the name of the file and where it is stored. Channel properties and parameters, tuner properties, and transmitter properties are stored in property files.

**publish (a channel)**

To copy a channel (or channel update) to a transmitter in a way that enables it to be downloaded by a tuner. This operation (which includes copying information about the channel, such as channel properties) can be performed with Channel Copier or, as an alternative that users of earlier versions of BMC Marimba Client Automation components might favor, with Publisher.

**publish directory**

See *package directory*.

**Publisher**

A BMC Marimba Client Automation application that can be used to publish channels to a transmitter, as an alternative to Channel Copier; sometimes favored by those who have experience with earlier versions of BMC Marimba Client Automation components.

**repair**

An action performed on a channel to check and fix problems. When you repair a channel, the files and other objects in the channel are first verified. After verification is complete, one of the following occurs:

- There are no object mismatches. No repairs are needed.
- There are object mismatches. The tuner re-installs the affected files or registry entries if they are available from the tuner storage. If not, the tuner contacts the transmitter to download the files or registry entries needed to complete the repair.

See also *verify*.

**repeater**

A transmitter on which channels from a master transmitter have been replicated, and to which requests made by tuners can be redirected from the master transmitter.

**replication**

The transfer of channels from a master transmitter to a repeater or mirror transmitter. This is just a duplication of the channels and not the same operation performed by Channel Copier, which publishes channels when it copies them to a transmitter.

**roll (a log file)**

To clear a log file before filling it with new data.

**rollback install**

An action triggered when an installation fails, causing the channel to be in a failed state. As a result, the channel ends up in an uninstalled state. All items are removed or restored to their previous states regardless of uninstallation policies and properties (such as nobackup). However, no scripts are executed and no dependencies are checked.

**rollback update**

An action triggered when a *major update* fails causing the channel to be in failed state. As a result, the channel ends up in a rollback state. This state indicates that the channel update has failed and has been brought back to its previous state. However, this state cannot successfully verify, repair, or install. It can only be “updated” with the correct version of the channel to bring it back into an “installed” state. During the rollback of an update, all files overwritten are temporarily backed up so they can be restored. Like rollback install, no scripts are executed and no dependencies are checked during this phase.

**script**

A file in the form of batch files, executables, or Java classes that you can use to customize the behavior of your channel on the endpoint. Scripts are invoked at key times in your channel’s lifecycle (install, uninstall, update, verify, repair, execute, or a combination of these phases). There can be any number of scripts in a channel.

**security certificate**

A mechanism for ensuring that a channel being subscribed to comes from the proper source (*channel-signing certificate*).

**segment (of a channel)**

A platform-specific or localized version of a channel. Every channel is made up of one or more segments; the transmitter determines which one to send based on feedback it received from the tuner.

**segment ID**

A string that identifies the specific platform and locale for which a channel segment is intended—for example, Windows NT,x86/en\_US.

**semisilent installation**

A mode of installation (of a packaged channel) that provides user feedback in the form of progress bars but doesn’t offer any installation options.

**services**

Programs or processes that perform a specific system function, usually as background tasks, to support other programs on Windows NT, Windows 2000, and Windows XP. Typically, they do not require user interaction and do not have a user interface. Some Windows applications include and install services.

**signed channel**

A channel (or channel update) that has been digitally signed by means of a channel-signing certificate, guaranteeing subscribers that it comes from a reliable source and is not an unauthorized, possibly corrupted, copy. When the user subscribes to a signed channel, a dialog box appears that requests permission to download the channel.

**silent installation**

A mode of installation (of a packaged channel) that provides no user feedback and offers no installation options. It displays no dialog boxes or progress bars to the end user. The installer uses all of the default channel settings during installation.

**subscribe**

To request a channel from a transmitter via a tuner, causing the channel's files (and subsequent channel updates) to be downloaded to the workstation on which the tuner is located.

**Subscription Management**

See *Policy Manager*.

**targets**

The selected recipients of distributions, including individual users, computers, or groups of users or computers. See "Types of targets" on page 160.

**template file**

See *XML template file*.

**text modifier group**

Contains all the text changes that you want to make to one ASCII text file. The text modifier group specifies two things:

- The name of the text modifier group
- The path of the ASCII text file to be modified

See also *text modifier*.

**text modifier**

An Application Packager feature that allows you to insert and replace text in ASCII text files. The files to be modified can be part of your channel or they can be any files on the endpoint.

**transform files**

Files for MSI packages that the Windows Installer uses to modify the installation database at installation time and dynamically affect the installation behavior. Transform files typically have a .mst file name extension. Transform files are applied at initial installation; they cannot be applied to an already installed application.

See also *Microsoft installation (MSI) package*.

**transmitter**

A server component that delivers applications and content in the form of channels. Transmitters serve channels, similar to Web servers serving Web pages. Channels (and updates) are published to a transmitter and downloaded from there to its clients (either tuners or other transmitters running as repeaters or mirror transmitters). The transmitter is itself a channel that runs on the tuner or some other tuner under the control of an administrator.

**Transmitter Administrator**

A channel that an administrator uses to configure one or more transmitters.

**trusted channel**

A channel that has been given permission to read or write data anywhere on the user's system and to call any code, including operating system code. Channels must be signed to be designated as trusted.

**tuner**

A client component, it is the interface through which BMC Marimba Client Automation components interact. The tuner is the application through which users subscribe to channels that have been published on a transmitter. The tuner downloads the channel files (or updates to them) from the transmitter to the user's workstation.

**Tuner Administrator**

An application that allows you manage tuners remotely, controlling which channels appear on tuners or changing their configuration in other ways.

**tuner property**

One of many characteristics of a tuner, including the URL of its primary channel, the URL for tuner updates, and the update schedule for its channels, as well as licenses and proxy details. Tuner properties can be set when the tuner is configured with Tuner Packager, and some properties can also be set during or after tuner installation.

**update schedule**

The periodic timing with which updates are automatically delivered to channels that have been subscribed to and downloaded. This schedule is set up either when the channel is published or later at the user's request.

**verify**

An action performed on a channel to check if there are any problems. When you verify a channel, the files and other objects in the channel are compared with the information for the channel when it was originally installed. Any object mismatches found are recorded in the log file.

**WFP**

See *Windows File Protection (WFP)*.

**Windows File Protection (WFP)**

A feature in the Windows 2000 product family for protecting system files to ensure that only the owners or vendors of those files may modify them.

**Windows Installer**

An installation and configuration service that ships as part of the Microsoft Windows 2000 operating system; it can also be installed on Windows 95/98 and Windows NT 4.0. See also *Microsoft installation (MSI) package*.

**Windows Installer Packager**

An Application Packager component that enables you package Microsoft installations (MSI) that were created for the Microsoft Windows Installer.

**Windows Terminal Services (WTS)**

Services that provide clients remote access to applications that run on a server. Applications are installed on the server only, and clients access the applications through WTS client software. WTS transmits only the user interface of the application to the client. The client then returns keyboard and mouse clicks to be processed by the server.

**workspace directory**

The directory on the user's workstation that is, by default, where the tuner stores channel files (in called channel directories) and that is normally the only place where files accessed by channels can be located. For a transmitter, this term refers to the directory where the transmitter stores data and channels; by default it is a subdirectory of the transmitter's channel directory, but it is typically configured to be a different directory. Similarly, a proxy has a workspace directory where it stores data, including its channel cache.

**WTS**

See *Windows Terminal Services (WTS)*.

**XML template file**

A file that allows you to save, modify, and apply default settings and configurations for Application Packager and channels created using Application Packager. These settings include installation policies, macros, scripts, installation modes, and other settings that can apply to multiple applications.

# Index

## A

access control lists (ACLs)  
    command-line options 373  
    enabling and disabling 156  
    prerequisites 156  
    primary administrators 73  
    roles 73  
    setting up 155  
access to Policy Manager features, limiting 72  
-aclCheck command-line option 373  
-aclGet command-line option 373  
-aclRemove command-line option 374  
-aclSet command-line option 374  
activation for packages in a policy 197  
Active Directory  
    automatic discovery of the Global Catalog 53  
    browsing and targeting in a forest  
        environment 50  
    changing MaxPageSize 54  
    classes and attributes added by Policy  
        Management 42  
    collections 50  
    containers 46  
    distribution, domain local groups 50  
    Global Catalog 52  
    group types 49  
    installation scripts 48  
    integration with Policy Management 37, 38  
    large numbers of groups 389  
    LDIF scripts 48

object attributes 43, 59  
object classes 43  
overview of 38  
permissions for a multidomain forest  
    environment 78  
requirements for a forest environment 50  
schema changes 42  
schema modifications 42  
setting permissions in 75  
specifying the bind DN for configuration 154  
UPN format for the bind DN 154  
using automatic discovery 40  
using groups as targets 49  
viewing a large number of groups 54  
Active Directory to ADAM Synchronizer 47  
ADAM  
    classes and attributes added by Policy  
        Management 42  
    integration with Policy Management 37  
    overview of 38  
    schema modifications 42  
    setting permissions in 78  
    specifying the base DN for configuration 153  
    specifying the bind DN for configuration 154  
    troubleshooting 392  
ADAM Synchronizer 47  
adamsync 47  
admin user name and password 393  
advertise state 191  
All Endpoints  
    description 162

- All Endpoints target
    - in Active Directory 52
  - All Endpoints target type 162
  - Application Packager
    - used to create packages 175
    - using to configure reboot behavior 249
  - assemblies
    - .netmodule file extension 413
  - assign state for patch group assignments 271
  - assign state for remediation group
    - assignments 285
  - assignment states for a patch group 271, 365, 367
  - authentication
    - command-line options 350
    - required command-line options 346
  - automatic discovery, using for Active Directory 40
- B**
- base DN
    - configuring for ADAM 153
    - configuring for Sun ONE Directory 153
    - specifying for configuration 153
  - bind DN
    - configuring for Active Directory 154
    - configuring for ADAM 154
    - configuring for Java System Directory Server 154
    - configuring for Sun ONE Directory 154
    - specifying for configuration 154
    - specifying the password for configuration 154
    - using the UPN format for configuration 154
  - blackout period
    - defined 229
    - exempting packages 233
    - exempting patch groups 280
    - exempting Patch Service 279
    - exempting Policy Service 230
    - setting for a target 230
    - setting globally 230
  - blackout periods
    - using with policy updates 324
  - blackout schedules
    - format for command line 376
    - overview 140
  - BMC CM configuration object
    - attributes 119
    - BMC CM configuration object, overview 118
    - BMC Software, contacting 2
    - browser access port, changing 394
    - browsing packages 176
    - browsing targets 164

## C

- cache
  - calculating size in iPlanet 384
  - monitoring performance in iPlanet 386
- case sensitivity
  - command-line options 346
- centralized mode for collections 97
- changeorder command-line option 359
- channel parameters, defined 248
- channel properties. <Emphasis>See package properties.
- child containers
  - command for specifying 363
  - clientcertpw command-line option 351
- cn attribute 382
- collectionmachinebase property
  - security issues 80
- collections 50
  - based on a database query 163
  - based on an LDAP query 163
  - centralized mode 97
  - creating new members in 104
  - defined 86, 163
  - displaying large machine groups 103
  - distributed mode 98
  - installation issues 100, 101
  - LDAP-query based 86
  - list of members 103
  - modes 97
  - security issues 79
  - target type 162
  - using a directory service 86
  - using LDAP 86
  - using Report Center 86, 163
  - using the Inventory module 86, 163
- collections, moving in LDAP 95
- command-line options
  - ACLs and permissions 373

- authentication 350
- case sensitivity 346
- configuration 351
  - guidelines for using 344
  - permissions 71
  - policies 358, 372
    - processing order of commands 347
    - required authentication 346
    - return codes 346
    - session information 345
    - specifying child containers 363
    - tuner and package properties 346
  - common schedules, setting for packages 206
  - compliance, policy
    - definition 292
    - prerequisites 295
    - viewing for packages 311
    - viewing for targets 302
  - configSet command-line option 351
  - configuration
    - command-line options 351
    - limiting access to configuration features 72
    - policy compliance 157
    - Policy Manager 147
    - policy updates 319
    - setting properties in iPlanet 64
    - Subscription plug-in 148
  - configuration object
    - attributes 117
    - editing attributes for 120
    - overview 115, 118
  - configuration objects
    - editing attributes of 32
  - configuring LDAP query collections 88
  - conflict resolution
    - for package states and schedules 207
    - property values 257
    - when multiple packages have the same install priority 215
    - when multiple users edit properties 258
    - when multiple users edit the same policy 216
  - connection expiration time, directory service 154
  - connection pool size, directory service 154
  - console server, prerequisites for policy
    - compliance 295
  - containers, targeting 390
  - copying policies 217
  - creating LDAP query collections 89
  - creating policies 184
  - customer support 2

## D

  - D command-line option 375
  - database
    - using for collections 86, 163
    - database query collections 163
    - Daylight Savings Time changeover 140
  - delete command-line option 360
  - deleting LDAP query collections 95
  - deleting packages from targets 189
  - deleting policies 218
  - deleting policies, using the command line 360
  - deploying patches 271
  - deploying remediations groups
    - deploying to targets 285
  - Deployment Manager 315, 325
    - prerequisites for using with Policy Manager 316
    - quorum 322
    - specifying the tuner port 316
    - specifying the user name and password 316
  - differences in states and schedules, resolving for policies 209
  - directly assigned policy icon 164, 219
  - directly assigned target 164
  - Directory Information Tree (DIT) 56
  - directory service
    - using for collections 86
  - directory services
    - assigning directory service-to-repeater mappings 352
    - connection pool size 154
    - creating a mapping file for repeaters 33
    - definition 116
    - expiration time for the last successful host connection 154
    - index files for Sun ONE Directory 382
    - mapping file 33
    - mapping file example 35
    - permissions required for creating policies 75

removing Policy Management entries 397  
schema mapping example 123  
schema mapping parameters 122  
specifying the base DN for configuration 153  
specifying the bind DN for configuration 154  
UPN format for the bind DN 154  
using automatic discovery 40  
using SSL when connection with the transmitter 153  
using with repeaters 32  
disable the plug-in 152  
distinguished name 57, 115  
distributed mode for collections 98  
DIT (Directory Information Tree) 56  
documentation 20  
domain local groups, using for Policy Manager 75  
domains, using as targets in Active Directory 52

## E

Edit All button 187  
editing policies 184  
emergency user name 393  
enable the plug-in 152  
enabling and disabling policy updates 319  
endpoints  
    All Endpoints target type 162  
    defined 161  
    listing policy assignments 373  
    management concepts 220  
    prerequisites for policy compliance 296  
    types of targets 162  
entry points  
    for policy objects 121  
    when using Sun ONE Directory 57  
exclude state  
    for packages 164, 192  
    for patch group assignments 271  
    for remediation group assignments 285  
excluded target 164  
expiration for packages in a policy 197  
expiration time 139  
expiration time, directory service connection 154  
-export command-line option 361  
exporting policies from the command line 361

## F

forest environment  
    browsing and targeting 50  
    permissions required 78  
    requirements for 50

## G

getHostName() method 131  
Global Catalog  
    automatic discovery of 53  
    overview 52  
global groups  
    using for Policy Manager 74  
globalcatalogbase property 126  
group display limit, changing 55  
groupclass property 64, 122, 124  
groupmemberattr property 65, 123  
groupnameattr property 65, 122  
groupofnames object class 382  
groupofuniqueNames object class 382  
groups, trouble expanding 390

## I

immediate policy updates, performing 319  
immediate vs. scheduled events 133  
-import command-line option 361  
importing policies from the command line 361  
importmachinebase property  
    security issues 80  
indirectly assigned target 164  
inetorgperson object class 382  
install priority  
    conflict resolution 215  
    considerations 214  
    defined 212  
    packages 223  
    setting for packages 214  
install state 191  
installation  
    issues with collections 100  
    obtaining user information from the transmitter 112  
    scripts for Active Directory 48  
installation order 212  
installation schedules

defined 196  
 setting 200  
 installation states. *See* states.  
 install-persist state 192  
 install-start state 192  
 install-start-persist state 192  
 integration with Deployment Manager 315, 325  
**Inventory module**  
 integration with  
 using for collections 86, 163  
**iPlanet Directory (now called Sun ONE Directory)**  
 cache size 384  
 namespaces 56  
 object classes 63  
 trouble viewing targets 389

**L**

**LDAP**  
 cache hit ratio 386  
 database parameters, configuring 383  
 indexes, creating 383  
 multiple enterprises on one server 57  
 namespaces 56  
 using for collections 86  
**LDAP query collection**  
 overview 87  
**LDAP query collections** 163  
 configuring 88  
 creating 89  
 deleting 95  
 modifying 94  
 overview 86  
 prerequisites 88  
 previewing results for 93  
 refreshing 96  
 viewing 91  
**LDAP, moving collections** 95  
**-ldapservers command-line option** 352  
**LDIF**  
 for creating group entries 67  
 for creating group of groups entries 67  
 for creating machine entries 66  
 for creating machine group entries 66  
 for creating user entries 67  
 usage examples 66

**LDIF scripts**  
 Active Directory 48  
**-list command-line option** 362  
 listing policies from the command line 362  
**log files**  
 properties for rolling 126  
 troubleshooting 395  
**login problems**  
 troubleshooting 392  
**logs.roll.policy property** 126  
**logs.roll.size property** 127  
**logs.roll.versions property** 126  
**M**  
**machine name** 131  
**machineclass property** 123  
**machineimportbase property** 65  
**machinename property** 131  
**machinenameattr property** 65, 123  
**-machines command-line option** 354  
**machines flat file, importing** 354  
**management**  
 endpoints 220  
 mapping directory services to repeaters 352  
**mapping file** 352  
**marimba.ldap.browse.collectionbase property of**  
 the BMC CM configuration object 119  
**marimba.ldap.browse.collectionmachinebase**  
 property of the configuration object 119  
**marimba.ldap.browse.collectionmode property of**  
 the BMC CM configuration object 119  
**marimba.ldap.browse.hideentries property of the**  
 BMC CM configuration object 119  
**marimba.ldap.browse.machineclass property of**  
 the BMC CM configuration object 119  
**marimba.schemapatchversion property of the**  
 BMC CM configuration object 119  
**marimba.subscription.acl property of the**  
 Subscription configuration object 117  
**marimba.subscription.adminusers** 245, 389  
**marimba.subscription.autostart** 246  
**marimba.subscription.installmode** 244  
**marimba.subscription.machinename** 246

marimba.subscription.nodelete 243  
marimba.subscription.reapplyconfigonfail 244  
marimba.subscription.retrycount 243  
marimba.subscription.retryintervalsec 246  
marimba.subscription.retrytime 243  
marimba.subscription.snmpintervalsec  
    property 151  
marimba.subscription.timeout 247  
marimba.subscription.useshortcuts 243  
marimba.subscription.varytime 247  
marimba.subscriptionplugin.collectionbase  
    property of the Subscription configuration  
    object 125  
marimba.subscriptionplugin.collectionmachineba  
    se property of the Subscription configuration  
    object 125  
marimba.subscriptionplugin.collectionmode  
    property of the Subscription configuration  
    object 125  
marimba.subscriptionplugin.ldapcollectionbase  
    attribute 88  
marimba.subscriptionplugin.ldapcollectionsched  
    attribute 88  
marimba.subscriptionplugin.mode property of the  
    Subscription configuration object 117  
marimba.subscriptionplugin.overrideclientdn  
    property of the Subscription configuration  
    object 117  
marimba.subscriptionplugin.requireentriesinldap  
    property of the Subscription configuration  
    object 117  
marimba.subscriptionplugin.resolvetype property  
    of the Subscription configuration object 117  
marimba.subscriptionplugin.subscriptionbase  
    property of the Subscription configuration  
    object 118, 121  
marimba.subscriptionplugin.usednfromclientonly  
    property of the Subscription configuration  
    object 118, 125  
marimba.tuner.display.noerrors 247  
marimba.tuner.display.noprogress 247  
marimba.tuner.update.profile property 265  
MaxPageSize, changing for Active Directory 54  
member attribute 382  
mode property 65, 117

modes  
    multiple selection 174, 179  
    single selection 173, 178  
modifying LDAP query collections 94  
monitoring policy updates 321  
mrbaACL object class 63  
mrbaACLEntry attribute 62  
mrbaACLPermission attribute 62  
mrbaACLPrincipal attribute 62  
mrbaACLPrincipalDN attribute 62  
mrbaACLPrincipalType attribute 62  
mrbaACLResource attribute 62  
mrbaACLResourceDN attribute 62  
mrbaACLVVersion attribute 62  
mrbaARReferenceTag attribute 62  
mrbaBlackOutSched attribute 62  
mrbaChannel attribute 60  
mrbaChannelExemptBlackout attribute 61  
mrbaChannelInitSched attribute 61  
mrbaChannelOrder attribute 61  
mrbaChannelSecondary attribute 60  
mrbaChannelSecSched attribute 61  
mrbaChannelTitle attribute 61  
mrbaChannelUpdateSched attribute 61  
mrbaChannelVerRepairSched attribute 61  
mrbaCollection object class 63  
mrbaCollection object class 382  
mrbaConfig attribute 59  
mrbaLastUpdated attribute 59  
mrbaMachine object class 63  
mrbamachine object class 382  
mrbaPackageGroupDN attribute 62  
mrbaPackageGroupMemberOf attribute 62  
mrbaProperties object class 63  
mrbaSet object class 382  
mrbaSm attribute 61  
mrbaSQL attribute 59  
mrbaSQLCondition attribute 59  
mrbaSubscription object 68  
mrbaSubscription object class 63, 68, 382  
mrbasubscription object class 382  
mrbaTargetAll attribute 60  
mrbaTargetDN attribute 60  
mrbatargetdn attribute 382  
mrbaTargetTxGroup attribute 60

`mrbatargetxgroup` attribute 382  
`mrbaTargetTxUser` attribute 60  
`mrbatartetxuser` attribute 382  
.mst file name extension 422  
multidomain forest environment, permissions required 78  
multiple selection mode 174, 179

## N

-namespace command-line option 363  
.NET framework 418  
.netmodule file extension 413  
Network Operating System (NOS) Directory 39  
new policies, creating 184  
NOS (Network Operating System) Directory 39

## O

object identification number (OID) for schema changes 42  
`objectclass` attribute 382  
obtaining users from transmitters 127  
OID for schema changes 42  
OS provisioning 155

## P

package properties  
conflict resolution 257  
conflicts 391  
defined 248  
deleting 254, 255  
format 253  
`reboot.allow` 250  
`reboot.allowcancel` 250  
`reboot.force` 250  
`reboot.showdialog` 250  
setting 252  
setting from the command line 346, 370  
using spaces in 254  
package states  
verification of 130  
packages  
adding to policies 189  
browsing 176  
compliance 292  
created using Application Packager 175

defined 175  
deleting from targets 189  
exempting from the blackout period 233  
icon 175  
install priority 212, 223  
installation priority considerations 214  
removing from policies 189  
scheduling repair 203  
searching for 177  
setting common schedules 206  
setting install priority 214  
sorting 172  
specifying primary and secondary states 193  
state precedence 193  
states  
updating staged 142  
viewing details 180  
viewing policy compliance 311  
parameters, channel 248  
password  
client certificate 351  
specifying for command-line publish operations 355  
patch groups  
adding and removing from a policy 273  
assigning to targets 271  
assignment states 271, 365, 367  
defined 270  
deploying to targets 271  
exempting from the blackout period 280  
icon 270  
policy compliance 281  
simulating the installation 276  
uninstallation 275  
Patch Management  
integration with Policy Manager 269  
prerequisites for using with Policy Manager 270  
Patch Service  
defined 271  
exempting from the blackout period 279  
overriding the update schedule 278  
PatchManagement/PatchGroups folder on the transmitter 273  
-patchsubscribe command-line option 364

performing policy updates 319  
permissions  
    command-line options 373  
    defining for policies 79  
    enabling and disabling access control lists (ACLs) 156  
    for a multidomain forest environment 78  
    for command-line options 71  
    for Policy Manager 72  
    for the different user roles 73  
    for the Subscription plug-in 81  
    for the Subscriptions container 79  
    primary administrators 73  
    security issues 69  
permissions for transmitters  
    adding 260  
    defined 260  
    deleting 261  
    editing 260  
Plug-in Status 152  
plug-in, Subscription  
    configuring 148  
    defined 148  
    permissions for Subscription 81  
    previewing changes 151  
    publishing 71, 354  
    using with Sun ONE Directory 56  
policies  
    adding and removing patch groups 273  
    adding and removing remediation groups 287  
    adding packages 189  
    assigning patch groups to target machines 271  
    assigning remediation groups to target machines 285  
    command-line options 358, 372  
    conflict resolution 216  
    copying 217  
    creating 184  
    defined 184  
    defining read write permissions 79  
    deleting 218  
    deleting from the command line 360  
    deleting old 390  
    directory service permissions required for creating 75  
    editing 184  
    icon for directly assigned policies 164  
    listing endpoint assignments 373  
    listing from the command line 362  
    previewing changes 187  
    removing packages 189  
    resolving differences in states and schedules 209  
    retrying a failed policy 130  
    specifying primary and secondary states 193  
    used as models for provisioning 155  
    viewing compliance information 302, 311  
policy compliance  
    configuration 157  
    defined 281, 292  
    for patch groups 281  
    prerequisites 295  
    prerequisites for the console server 295  
    prerequisites for the endpoints 296  
    viewing for packages 311  
    viewing for targets 302  
policy icon 164, 219  
Policy Management  
    attributes, Sun ONE Directory 59  
    configuring 31  
    deleting policies from the command line 360  
    exporting policies from the command line 361  
    importing policies from the command line 361  
    security issues 69  
Policy Management collections. *See* collections.  
Policy Management entries, removing from the directory service 397  
Policy Manager  
    changing the group display limit 55  
    configuring 147  
    naming conventions 68  
    permissions 72  
    troubleshooting 388  
    user roles for 72  
policy objects, entry point for 121  
Policy Service  
    defined 235  
    exempting from the blackout period 230

- schedules 132
- scheduling updates 236
- using with a newer plug-in 36
- policy updates
  - enabling and disabling 319
  - how it works 323
  - monitoring and viewing status 321
  - performing 319
  - prerequisites for using 316
  - retrying 322
  - stopping and retrying 322
  - using with blackout periods 324
- port number for browser access, changing 394
- Power Options for Windows targets 239
- precedence, subscription state 193
- prerequisites
  - LDAP query collections 88
  - prerequisites for access control lists (ACLs) 156
- previewing policies 187
- previewing results for LDAP query collections 93
- primary channel state 192
- primary schedule
  - defined 196
  - example 138
  - format for command line 379
  - setting 200
- primary states
  - list of 191
  - specifying for a package 193
  - specifying for policies 193
- priority for properties 248, 252, 253
- processing order, command line 347
- product support 2
- profiles
  - loading 264
  - types of 262
- profiles for Windows Power Options 158
- properties
  - conflicts 391
  - conflicts for tuner properties 258
  - deleting package and tuner properties 254, 255
  - for package reboot settings 249
  - for packages 248
  - for tuners 242
- format for package and tuner properties 253
- list of commonly used tuner properties 242
- setting package properties 252
- setting tuner properties 251
- using spaces in 254
- property priority 248, 252, 253
- property values
  - conflict resolution 257
- provision by policy 155
- provisioning
  - using policies as models 155
- publish command-line option 354
- publishing the Subscription plug-in 354
- publishpw command-line option 355

## Q

- quorum
  - Deployment Manager 322

## R

- reboot settings
  - for packages, properties that control 249
  - for Windows targets 239
- reboot.allow property 250
- reboot.allowcancel property 250
- reboot.force property 250
- reboot.showdialog property 250
- recurrence for scheduled activities 199
- refreshing LDAP query collections 96
- remediation group
  - defined 284
- remediation groups
  - adding and removing from a policy 287
  - assigning to targets 285
- remedysubscribe command-line option 366
- removing
  - LDAP query collections 95
  - removing Policy Management entries from the directory service 397
- repair schedule
  - defined 198
  - format for the command line 379
  - setting 203
- repeaters
  - directory service mapping file 33

using with directory services 32  
when obtaining users from Transmitter 113

Report Center  
prerequisites for using with Policy Manager 316  
using for collections 86, 163

reporter  
listing policies assigned to endpoints 373  
-reporter command-line option 373  
retrying a failed policy 130  
return codes, command line 346  
roles, for users in Policy Manager 72  
runchannel  
command syntax 344  
running LDAP query collections 96

**S**

saving policies 187

schedules  
activation for packages in a policy 197  
as enforced by Policy Service 132  
blackout format for command line 376  
blackout period 140, 229  
conflict resolution 207  
example of primary 138  
example of secondary 138  
expiration for packages in a policy 197  
expiration time 139  
format for command line, primary or secondary schedule 379  
format for specifying 376  
format for the command line, repair schedule 379  
format for the command line, update schedule 379  
life cycle of 137  
overview of policy assignment schedules 195  
Patch Service updates 278  
Policy Service updates 236  
primary schedule 196, 200  
recurrence for activities 199  
repair 198  
secondary schedule 196, 200  
setting common schedules for multiple packages 206

setting for packages 190  
types of 136  
update 198  
varying time 141

scheduling  
updates 221

schema  
Active Directory 42  
ADAM 42

script inserts  
using on different platforms 265

scripts  
for Active Directory installation 48

search  
for packages 177  
for targets 167

secondary schedule  
example 138  
format for command line 379

secondary schedules  
defined 196  
setting 200

secondary states  
list of 191  
specifying for a package 193  
specifying for policies 193

security issues 69

service records (SRV) 53

session for the command line 345  
-setpluginparam command-line option 356

setting up access control lists (ACLs) 155

simulation  
installation of patch groups 276  
viewing details about the installation of patch groups 277

single selection mode 173, 178

sorting the list of packages 172

sorting the list of targets 180

sourcing users from transmitters 127

special characters in search strings 170

SRV records 53

SSL, using between the transmitter and directory service 153

stage state 191  
description 191

- updating packages 142
- staging
  - updates 221, 222
- starting the plug-in 152
- state precedence 193
- states
  - advertise 191
  - assign 271, 285
  - assigned vs. endpoint 135
  - conflict resolution 207
  - defined 191
  - endpoint transitions 135
  - exclude 164, 192, 271, 285
  - for patch groups 271, 365, 367
  - install 191
  - install-persist 192
  - install-start 192
  - install-start-persist 192
  - list of primary states 191
  - list of secondary states 191
  - package installation retry 130
  - primary channel 192
  - stage 191
  - uninstall 192
  - verification 130
- Status icon 74
- stopping the plug-in 152
- subscribe command-line option 367
- subscribing targets to packages from the command line 367
- subscribing targets to patch groups from the command line 364
- subscribing targets to remediation groups from the command line 366
- Subscription configuration object
  - attributes 117
  - editing attributes for 120
  - location of 116
  - overview 115
- Subscription plug-in
  - configuring 148
  - defined 148
  - permissions 81
  - previewing changes 151
  - publishing 71, 354
- using with Sun ONE Directory 56
- subscription policies. *See* policies.
- Subscription Policy Manager*See* Policy Manager
- subscriptionbase property 65, 121
- Subscriptions container, permissions for 79
- Sun Java System Directory Server
  - setting permissions in 77
  - specifying the bind DN for configuration 154
- Sun Java System directory Server
  - configuring replicants 33
- Sun ONE Directory
  - configuring replicants 33
  - optimizing performance 382
  - performance tuning 381
  - setting permissions in 77
  - specifying the base DN for configuration 153
  - specifying the bind DN for configuration 154
- support, customer 2
- synchronizing Active Directory to ADAM 47
- syntax for runchannel commands 344
- system reboots, properties that control 249
- system settings, troubleshooting 392

## T

Target Details page, using to delete packages from targets 218

Target View page, using to delete packages from targets 218

### targets

- All Endpoints 162
- browsing 164
- collection 162
- defined 161
- directly assigned 164
- excluded targets 164
- icons for different types 162
- indirectly assigned 164
- resolving differences in policies 209
- searching for 167
- searching using special characters 170
- sorting 180
- targeting containers 390
- trouble expanding groups 389
- trouble viewing 389
- types of targets 162

- viewing details 171
  - viewing policy compliance 302
  - technical support 2
  - transmitter permissions
    - adding 260
    - defined 260
    - deleting 261
    - editing 260
  - transmitters
    - sourcing users from 127
    - using as the source of users and groups 109
    - using SSL when connecting to the directory service 153
  - troubleshooting
    - ADAM 392
    - finding log files 395
    - logging in as an administrator on endpoint users' machines 388
    - login problems 392
    - Policy Manager 388
    - system settings 392
    - user timeout 394, 395
  - tuner command-line option 370
  - tuner properties
    - conflict resolution 257
    - conflicts 258, 391
    - defined 242
    - deleting 254, 255
    - format 253
    - list of commonly used tuner properties 242
    - marimba.subscription.adminusers 245, 389
    - marimba.subscription.autostart 246
    - marimba.subscription.installmode 244
    - marimba.subscription.machinename 246
    - marimba.subscription.nodelete 243
    - marimba.subscription.reapplyconfigonfail 24 4
    - marimba.subscription.retrycount 243
    - marimba.subscription.retryintervalsec 246
    - marimba.subscription.retrytime 243
    - marimba.subscription.timeout 247
    - marimba.subscription.useshortcuts 243
    - marimba.subscription.varytime 247
    - marimba.tuner.display.noerrors 247
    - marimba.tuner.display.noprogress 247
  - setting for targets 251
  - setting from the command line 346, 370
  - setting from the GUI 251
  - using spaces in 254
  - txadminaccess command 112
  - txadminaccess command-line option 357
- U**
- uid attribute 382
  - uninstall state 192
  - uninstallation of patch groups 275
  - uniqueMember attribute 382
  - universal groups, using for Policy Manager 74
  - update schedule
    - defined 198
    - format for the command line 379
    - Patch Service 278
    - Policy Service 236
    - setting 201
  - updates
    - scheduling 221
    - staging 221, 222
  - stopping and retrying for policies 322
- updating endpoints, overview 28
- UPN (user principal name) format 154
  - usecomputername property 131
  - usednfromclientonly property 125, 127
  - useglobalcatalog property 126
  - password command-line option 350
  - user command-line option 350
  - user principal name (UPN) format 154
  - user roles
    - in Policy Manager 72
  - user timeout, troubleshooting 394, 395
  - userclass property 122
  - useridattr property 122
  - users and groups, using transmitters as the source 109
  - usetransmitterusers property 110, 126
- V**
- varytime property 141
  - viewing LDAP query collections 91

## W

-w command-line option 375

wake up machines 194

Wake-on-Wan (WoW) 194

WOW deployments 186

WoW deployments 194

## A

access control lists (ACLs)

command-line options 416

enabling and disabling 154

prerequisites 154

primary administrators 69

roles 69

setting up 153

access to Policy Manager features, limiting 68

-aclCheck command-line option 416

-aclGet command-line option 416

-aclRemove command-line option 416

-aclSet command-line option 416

activation for packages in a policy 233

Active Directory

automatic discovery of the Global Catalog 49

browsing and targeting in a forest environment 46

changing MaxPageSize 50

classes and attributes added by Policy Management 39

collections 46

containers 42

distribution, domain local groups 46

Global Catalog 48

group types 45

installation scripts 44

integration with Policy Management 33, 34

large numbers of groups 431

LDIF scripts 44

object attributes 39, 55

object classes 39

overview of 34

permissions for a multidomain forest environment 74

requirements for a forest environment 46

schema changes 38

schema modifications 38

setting permissions in 71

specifying the bind DN for configuration 152

UPN format for the bind DN 152

using automatic discovery 37

using groups as targets 45

viewing a large number of groups 50

Active Directory to ADAM Synchronizer 43

## ADAM

classes and attributes added by Policy Management 39

integration with Policy Management 33

overview of 34

schema modifications 38

setting permissions in 74

specifying the base DN for configuration 151

specifying the bind DN for configuration 152

troubleshooting 434

ADAM Synchronizer 43

adamsync 43

admin user name and password 435

advertise state 215, 216

All Endpoints

description 160

All Endpoints target

in Active Directory 48  
All Endpoints target type 160  
Application Packager  
    used to create packages 175  
    using to configure reboot behavior 283  
assemblies  
    .netmodule file extension 455  
assign state for patch group assignments 297  
assignment states for a patch group 297, 407, 409  
authentication  
    command-line options 392  
    required command-line options 388  
automatic discovery, using for Active Directory 37  
**B**  
base DN  
    configuring for ADAM 151  
    configuring for Sun ONE Directory 151  
    specifying for configuration 151  
bind DN  
    configuring for Active Directory 152  
    configuring for ADAM 152  
    configuring for Java System Directory Server 152  
    configuring for Sun ONE Directory 152  
    specifying for configuration 152  
    specifying the password for configuration 152  
    using the UPN format for configuration 152  
blackout period  
    defined 200  
    exempting packages 204  
    exempting patch groups 306  
exempting Patch Service 305  
exempting Policy Service 201  
setting for a target 201  
    setting globally 201  
blackout periods  
    using with policy updates 350  
blackout schedules  
    format for command line 418  
    overview 136  
BMC CM configuration object  
    attributes 115  
BMC CM configuration object, overview 114  
BMC Software, contacting 2  
browser access port, changing 436  
browsing packages 176  
browsing targets 165  
**C**  
cache  
    calculating size in iPlanet 426  
    monitoring performance in iPlanet 428  
case sensitivity  
    command-line options 388  
centralized mode for collections 93  
-changeorder command-line option 401  
channel parameters, defined 282  
channel properties. <Emphasis>See package properties.  
child containers  
    command for specifying 405  
-clientcertpw command-line option 393  
cn attribute 424  
collectionmachinebase property  
    security issues 76  
collections 46  
    based on a database query 161  
    based on an LDAP query 161  
    centralized mode 93

- creating new members in 100
- defined 82, 161
- displaying large machine groups 99
- distributed mode 93
- installation issues 96, 97
- LDAP-query based 82
- list of members 99
- modes 93
- security issues 75
- target type 160
- using a directory service 82
- using LDAP 82
- using Report Center 82, 161
- using the Inventory module 82, 161
- collections**, moving in LDAP 91
- command-line options**
  - ACLs and permissions 416
  - authentication 392
  - case sensitivity 388
  - configuration 393
  - guidelines for using 386
  - permissions 67
  - policies 400, 415
  - processing order of commands 389
  - required authentication 388
  - return codes 388
  - session information 387
  - specifying child containers 405
  - tuner and package properties 388
- common schedules**, setting for packages 243
- compliance, policy**
  - definition 318
  - prerequisites 321
  - viewing for packages 337
  - viewing for targets 328
- configSet command-line option** 393
- configuration**
  - command-line options 393
  - limiting access to configuration features 68
  - policy compliance 155
  - Policy Manager 145
  - policy updates 345
  - setting properties in iPlanet 60
  - Subscription plug-in 146
- configuration object**
  - attributes 113
  - editing attributes for 116
  - overview 111, 114
- configuration objects**
  - editing attributes of 28
- configuring LDAP query collections** 84
- conflict resolution**
  - for package states and schedules 244
  - property values 291
  - when multiple packages have the same install priority 252
  - when multiple users edit properties 292
  - when multiple users edit the same policy 252
- connection expiration time, directory service** 152
- connection pool size, directory service** 152
- console server, prerequisites for policy compliance** 321
- containers, targeting** 432
- copying policies** 253
- creating LDAP query collections** 85
- creating policies** 209
- customer support** 3
- D**
- D command-line option** 417
- database**
  - using for collections 82, 161

- database query collections 161
- Daylight Savings Time changeover 136
- delete command-line option 402
- deleting LDAP query collections 91
- deleting packages from targets 214
- deleting policies 254
- deleting policies, using the command line 402
- deploying patches 297
- Deployment Manager 341, 351, 371
  - prerequisites for using with Policy Manager 342
  - quorum 348
  - specifying the tuner port 342
  - specifying the user name and password 342
- differences in states and schedules, resolving for policies 246
- directly assigned policy icon 162, 256
- directly assigned target 162
- Directory Information Tree (DIT) 52
- directory service
  - using for collections 82
- directory services
  - assigning directory service-to-repeater mappings 394
  - connection pool size 152
  - creating a mapping file for repeaters 29
  - definition 112
  - expiration time for the last successful host connection 152
  - index files for Sun ONE Directory 424
  - mapping file 29
  - mapping file example 31
  - permissions required for creating policies 71
  - removing Policy Management entries
- 439
- schema mapping example 119
- schema mapping parameters 118
- specifying the base DN for configuration 151
- specifying the bind DN for configuration 152
- UPN format for the bind DN 152
- using automatic discovery 37
- using SSL when connection with the transmitter 151
- using with repeaters 28
- disable the plug-in 150
- distinguished name 53, 111
- distributed mode for collections 93
- DIT (Directory Information Tree) 52
- documentation 16
- domain local groups, using for Policy Manager 71
- domains, using as targets in Active Directory 48
- E
- Edit All button 211
- editing policies 209
- emergency user name 435
- enable the plug-in 150
- enabling and disabling policy updates 345
- endpoints
  - All Endpoints target type 160
  - defined 159
  - listing policy assignments 415
  - management concepts 260
  - prerequisites for policy compliance 322
  - types of targets 160
- entry points
  - for policy objects 117
  - when using Sun ONE Directory 53

exclude state  
  for packages 162, 217  
  for patch group assignments 297  
excluded target 162  
expiration for packages in a policy 233  
expiration time 135  
expiration time, directory service connection 152  
-export command-line option 403  
exporting policies from the command line 403

F

forest environment  
  browsing and targeting 46  
  permissions required 74  
  requirements for 46

G

getHostName() method 127

Global Catalog  
  automatic discovery of 49  
  overview 48

global groups  
  using for Policy Manager 70

globalcatalogbase property 122

group display limit, changing 51

groupclass property 60, 118, 120

groupmemberattr property 61, 119

groupnameattr property 61, 118

groupofnames object class 424

groupofuniquenames object class 424

groups, trouble expanding 432

I

immediate policy updates, performing 345

immediate vs. scheduled events 129

-import command-line option 403

importing policies from the command line 403

importmachinebase property

  security issues 76

indirectly assigned target 162

inetorgperson object class 424

install priority  
  conflict resolution 252  
  considerations 250  
  defined 249  
  packages 263  
  setting for packages 251

install state 216

installation  
  issues with collections 96  
  obtaining user information from the transmitter 108  
  scripts for Active Directory 44

installation order 249

installation schedules  
  defined 232  
  setting 237

installation states. *See* states.

install-persist state 216

install-start state 216

install-start-persist state 216

integration with Deployment Manager 341, 351, 371

Inventory module  
  integration with  
  using for collections 82, 161

iPlanet Directory (now called Sun ONE Directory)  
  cache size 426  
  namespaces 52  
  object classes 59  
  trouble viewing targets 431

L

LDAP  
  cache hit ratio 428  
  database parameters, configuring 425

indexes, creating 425  
multiple enterprises on one server 53  
namespaces 52  
using for collections 82

LDAP query collection  
overview 83

LDAP query collections 161  
configuring 84  
creating 85  
deleting 91  
modifying 90  
overview 82  
prerequisites 84  
previewing results for 89  
refreshing 92  
viewing 87

LDAP, moving collections 91

-ldapservers command-line option 394

LDIF  
for creating group entries 63  
for creating group of groups entries 63  
for creating machine entries 62  
for creating machine group entries 62  
for creating user entries 63  
usage examples 62

LDIF scripts  
Active Directory 44

-list command-line option 404

listing policies from the command line 404

log files  
properties for rolling 122  
troubleshooting 437

login problems  
troubleshooting 434

logs.roll.policy property 122

logs.roll.size property 123

logs.roll.versions property 122

M  
machine name 127  
machineclass property 119  
machineimportbase property 61  
machinename property 127  
machinenameattr property 61, 119  
-machines command-line option 396  
machines flat file, importing 396  
management  
endpoints 260  
mapping directory services to repeaters 394  
mapping file 394  
marimba.ldap.browse.collectionbase  
property of the BMC CM configuration object 115  
marimba.ldap.browse.collectionmachinebase  
property of the configuration object 115  
marimba.ldap.browse.collectionmode  
property of the BMC CM configuration object 115  
marimba.ldap.browse.hideentries property  
of the BMC CM configuration object 115  
marimba.ldap.browse.machineclass prop-  
erty of the BMC CM configuration object  
115  
marimba.schemapatchversion property of  
the BMC CM configuration object 115  
marimba.schemaversion property of the  
BMC CM configuration object 115  
marimba.subscription.acl property of the  
Subscription configuration object 113  
marimba.subscription.adminusers 279,  
431  
marimba.subscription.autostart 280  
marimba.subscription.installmode 278  
marimba.subscription.machinename 280

marimba.subscription.nodelete 277  
marimba.subscription.reapplyconfigonfail 278  
marimba.subscription.retrycount 277  
marimba.subscription.retryintervalsec 280  
marimba.subscription.retrytime 277  
marimba.subscription.snmpintervalsec property 149  
marimba.subscription.timeout 281  
marimba.subscription.useshortcuts 277  
marimba.subscription.varytime 281  
marimba.subscriptionplugin.collectionbase property of the Subscription configuration object 121  
marimba.subscriptionplugin.collectionmachinebase property of the Subscription configuration object 121  
marimba.subscriptionplugin.collectionmode property of the Subscription configuration object 121  
marimba.subscriptionplugin.ldapcollectionbase attribute 84  
marimba.subscriptionplugin.ldapcollectionsched attribute 84  
marimba.subscriptionplugin.mode property of the Subscription configuration object 113  
marimba.subscriptionplugin.overrideclientdn property of the Subscription configuration object 113  
marimba.subscriptionplugin.requireentriesinldap property of the Subscription configuration object 113  
marimba.subscriptionplugin.resolvetype property of the Subscription configuration object 113  
marimba.subscriptionplugin.subscriptionbase property of the Subscription configu ration object 114, 117  
marimba.subscriptionplugin.usednfromclientonly property of the Subscription configuration object 114, 121  
marimba.tuner.display.noerrors 281  
marimba.tuner.display.noprogress 281  
marimba.tuner.update.profile property 313  
MaxPageSize, changing for Active Directory 50  
member attribute 424  
mode property 61, 113  
modes  
    multiple selection 174, 179  
    single selection 173, 178  
modifying LDAP query collections 90  
monitoring policy updates 347  
mrbaACL object class 59  
mrbaACLEntry attribute 58  
mrbaACLPPermission attribute 58  
mrbaACLPrincipal attribute 58  
mrbaACLPrincipalDN attribute 58  
mrbaACLPrincipalType attribute 58  
mrbaACLResource attribute 58  
mrbaACLResourceDN attribute 58  
mrbaACLVersion attribute 58  
mrbaARReferenceTag attribute 58  
mrbaBlackOutSched attribute 58  
mrbaChannel attribute 56  
mrbaChannelExemptBlackout attribute 57  
mrbaChannelInitSched attribute 57  
mrbaChannelOrder attribute 57  
mrbaChannelSecondary attribute 56  
mrbaChannelSecSched attribute 57  
mrbaChannelTitle attribute 57  
mrbaChannelUpdateSched attribute 57  
mrbaChannelVerRepairSched attribute 57  
mrbaCollection object class 59

mrbacollection object class 424  
mrbaConfig attribute 55  
mrbaLastUpdated attribute 55  
mrbaMachine object class 59  
mrbamachine object class 424  
mrbaPackageGroupDN attribute 58  
mrbaPackageGroupMemberOf attribute 58  
mrbaProperties object class 59  
mrbase object class 424  
mrbsm attribute 57  
mrbaSQL attribute 55  
mrbaSQLCondition attribute 55  
mrbaSubscription object 64  
mrbaSubscription object class 59, 64, 424  
mrbasubscription object class 424  
mrbaTargetAll attribute 56  
mrbaTargetDN attribute 56  
mrbatargetdn attribute 424  
mrbaTargetTxGroup attribute 56  
mrbatargettxgroup attribute 424  
mrbaTargetTxUser attribute 56  
mrbatartetxuser attribute 424  
.mst file name extension 464  
multidomain forest environment, permissions required 74  
multiple selection mode 174, 179

N

-namespace command-line option 405  
.NET framework 460  
.netmodule file extension 455  
Network Operating System (NOS) Directory 35  
new policies, creating 209  
NOS (Network Operating System) Directory 35

O

object identification number (OID) for schema changes 38  
objectclass attribute 424  
obtaining users from transmitters 123  
OID for schema changes 38  
OS provisioning 153

P

package properties

- conflict resolution 291
- conflicts 433
- defined 282
- deleting 288, 289
- format 287
- reboot.allow 284
- reboot.allowcancel 284
- reboot.force 284
- reboot.showdialog 284
- setting 286
- setting from the command line 388, 412
- using spaces in 288

package states

- verification of 126

packages

- adding to policies 213
- browsing 176
- compliance 318
- created using Application Packager 175
- defined 175
- deleting from targets 214
- exempting from the blackout period 204
- icon 175
- install priority 249, 263
- installation priority considerations 250
- removing from policies 214
- scheduling repair 241
- searching for 177

- setting common schedules 243
- setting install priority 251
- sorting 172
- specifying primary and secondary states 218
- state precedence 217
- states
  - updating staged 138
  - viewing details 180
  - viewing policy compliance 337
- parameters, channel 282
- password
  - client certificate 393
  - specifying for command-line publish operations 397
- patch groups
  - adding and removing from a policy 299
  - assigning to targets 297
  - assignment states 297, 407, 409
  - defined 296
  - deploying to targets 297
  - exempting from the blackout period 306
  - icon 296
  - policy compliance 307
  - simulating the installation 302
  - uninstallation 301
- Patch Management
  - integration with Policy Manager 295
  - prerequisites for using with Policy Manager 296
- Patch Service
  - defined 297
  - exempting from the blackout period 305
  - overriding the update schedule 304
- PatchManagement/PatchGroups folder on the transmitter 299
- patchsubscribe command-line option 406
- performing policy updates 345
- permissions
  - command-line options 416
  - defining for policies 75
  - enabling and disabling access control lists (ACLs) 154
  - for a multidomain forest environment 74
  - for command-line options 67
  - for Policy Manager 68
  - for the different user roles 69
  - for the Subscription plug-in 77
  - for the Subscriptions container 75
  - primary administrators 69
  - security issues 65
- permissions for transmitters
  - adding 308
  - defined 308
  - deleting 309
  - editing 308
- Plug-in Status 150
- plug-in, Subscription
  - configuring 146
  - defined 146
  - permissions for Subscription 77
  - previewing changes 149
  - publishing 67, 396
  - using with Sun ONE Directory 52
- policies
  - adding and removing patch groups 299
  - adding packages 213
  - assigning patch groups to target machines 297
  - command-line options 400, 415
  - conflict resolution 252
  - copying 253

- creating 209
- defined 208
- defining read write permissions 75
- deleting 254
- deleting from the command line 402
- deleting old 432
- directory service permissions required
  - for creating 71
- editing 209
- icon for directly assigned policies 162
- listing endpoint assignments 415
- listing from the command line 404
- previewing changes 211
- removing packages 214
- resolving differences in states and schedules 246
- retrying a failed policy 126
- specifying primary and secondary states 218
- used as models for provisioning 153
- viewing compliance information 328, 337
- policy compliance
  - configuration 155
  - defined 307, 318
  - for patch groups 307
  - prerequisites 321
  - prerequisites for the console server 321
  - prerequisites for the endpoints 322
  - viewing for packages 337
  - viewing for targets 328
- policy icon 162, 256
- Policy Management
  - attributes, Sun ONE Directory 55
  - configuring 27
  - deleting policies from the command line 402
  - exporting policies from the command line 403
  - importing policies from the command line 403
  - security issues 65
  - Policy Management collections. *See* collections.
  - Policy Management entries, removing from the directory service 439
  - Policy Manager
    - changing the group display limit 51
    - configuring 145
    - naming conventions 64
    - permissions 68
    - troubleshooting 430
    - user roles for 68
  - policy objects, entry point for 117
- Policy Service
  - defined 270
  - exempting from the blackout period 201
  - schedules 128
  - scheduling updates 271
  - using with a newer plug-in 32
- policy updates
  - enabling and disabling 345
  - how it works 349
  - monitoring and viewing status 347
  - performing 345
  - prerequisites for using 342
  - retrying 348
  - stopping and retrying 348
  - using with blackout periods 350
- port number for browser access, changing 436
- Power Options for Windows targets 273
- precedence, subscription state 217
- prerequisites

- LDAP query collections 84
- prerequisites for access control lists (ACLs) 154
- previewing policies 211
- previewing results for LDAP query collections 89
- primary channel state 217
- primary schedule
  - defined 232
  - example 134
  - format for command line 422
  - setting 237
- primary states
  - list of 216
  - specifying for a package 218
  - specifying for policies 218
- priority for properties 282, 286, 287
- processing order, command line 389
- product support 3
- profiles
  - loading 312
  - types of 310
- profiles for Windows Power Options 156
- properties
  - conflicts 433
  - conflicts for tuner properties 292
  - deleting package and tuner properties 288, 289
  - for package reboot settings 283
  - for packages 282
  - for tuners 276
  - format for package and tuner properties 287
  - list of commonly used tuner properties 276
  - setting package properties 286
  - setting tuner properties 285
  - using spaces in 288
- property priority 282, 286, 287
- property values
  - conflict resolution 291
- provision by policy 153
- provisioning
  - using policies as models 153
- publish command-line option 396
- publishing the Subscription plug-in 396
- publishpw command-line option 397
- Q
- quorum
  - Deployment Manager 348
- R
- reboot settings
  - for packages, properties that control 283
  - for Windows targets 273
- reboot.allow property 284
- reboot.allowcancel property 284
- reboot.force property 284
- reboot.showdialog property 284
- recurrence for scheduled activities 236
- refreshing LDAP query collections 92
- remedysubscribe command-line option 408
- removing
  - LDAP query collections 91
- removing Policy Management entries from the directory service 439
- repair schedule
  - defined 235
  - format for the command line 422
  - setting 241
- repeaters
  - directory service mapping file 29
  - using with directory services 28
- when obtaining users from Transmitter 109

Report Center  
prerequisites for using with Policy Manager 342  
using for collections 82, 161

reporter  
listing policies assigned to endpoints 415

-reporter command-line option 415

retrying a failed policy 126

return codes, command line 388

roles, for users in Policy Manager 68

runchannel  
command syntax 386

running LDAP query collections 92

S

saving policies 211

schedules  
activation for packages in a policy 233  
as enforced by Policy Service 128  
blackout format for command line 418  
blackout period 136, 200  
conflict resolution 244  
example of primary 134  
example of secondary 134  
expiration for packages in a policy 233  
expiration time 135  
format for command line, primary or secondary schedule 422  
format for specifying 418  
format for the command line, repair schedule 422  
format for the command line, update schedule 422  
life cycle of 133  
overview of policy assignment schedules 231

Patch Service updates 304

Policy Service updates 271

primary schedule 232, 237  
recurrence for activities 236  
repair 235

secondary schedule 232, 237

setting common schedules for multiple packages 243

setting for packages 215

types of 132

update 234  
varying time 137

scheduling  
updates 261

schema  
Active Directory 38  
ADAM 38

script inserts  
using on different platforms 313

scripts  
for Active Directory installation 44

search  
for packages 177  
for targets 167

secondary schedule  
example 134  
format for command line 422

secondary schedules  
defined 232  
setting 237

secondary states  
list of 216  
specifying for a package 218  
specifying for policies 218

security issues 65

service records (SRV) 49

session for the command line 387

-setpluginparam command-line option 398

setting up access control lists (ACLs) 153

- simulation
  - installation of patch groups 302
  - viewing details about the installation of patch groups 303
- single selection mode 173, 178
- sorting the list of packages 172
- sorting the list of targets 180
- sourcing users from transmitters 123
- special characters in search strings 171
- SRV records 49
- SSL, using between the transmitter and directory service 151
- stage state 216
  - description 215
  - updating packages 138
- staging
  - updates 261, 262
- starting the plug-in 150
- state precedence 217
- states
  - advertise 215, 216
  - assign 297
  - assigned vs. endpoint 131
  - conflict resolution 244
  - defined 215
  - endpoint transitions 131
  - exclude 162, 217, 297
  - for patch groups 297, 407, 409
  - install 216
  - install-persist 216
  - install-start 216
  - install-start-persist 216
  - list of primary states 216
  - list of secondary states 216
  - package installation retry 126
  - primary channel 217
  - stage 215, 216
  - uninstall 217
- verification 126
- Status icon 70
- stopping the plug-in 150
- subscribe command-line option 409
- subscribing targets to packages from the command line 409
- subscribing targets to patch groups from the command line 406
- subscribing targets to remediation groups from the command line 408
- Subscription configuration object
  - attributes 113
  - editing attributes for 116
  - location of 112
  - overview 111
- Subscription plug-in
  - configuring 146
  - defined 146
  - permissions 77
  - previewing changes 149
  - publishing 67, 396
  - using with Sun ONE Directory 52
- subscription policies. *See* policies.
- Subscription Policy Manager*See* Policy Manager
- subscriptionbase property 61, 117
- Subscriptions container, permissions for 75
- Sun Java System Directory Server
  - setting permissions in 73
  - specifying the bind DN for configuration 152
- Sun Java System directory Server
  - configuring replicants 29
- Sun ONE Directory
  - configuring replicants 29
  - optimizing performance 424
  - performance tuning 423

setting permissions in 73  
specifying the base DN for configuration 151  
specifying the bind DN for configuration 152  
support, customer 3  
synchronizing Active Directory to ADAM 43  
syntax for runchannel commands 386  
system reboots, properties that control 283  
system settings, troubleshooting 434

**T**

Target Details page, using to delete packages from targets 255  
Target View page, using to delete packages from targets 255  
targets

- All Endpoints 160
- browsing 165
- collection 160
- defined 159
- directly assigned 162
- excluded targets 162
- icons for different types 160
- indirectly assigned 162
- resolving differences in policies 246
- searching for 167
- searching using special characters 171
- sorting 180
- targeting containers 432
- trouble expanding groups 431
- trouble viewing 431
- types of targets 160
- viewing details 171
- viewing policy compliance 328

technical support 3  
transmitter permissions

- adding 308

defined 308  
deleting 309  
editing 308

transmitters

- sourcing users from 123
- using as the source of users and groups 105
- using SSL when connecting to the directory service 151

troubleshooting

- ADAM 434
- finding log files 437
- logging in as an administrator on endpoint users' machines 430
- login problems 434
- Policy Manager 430
- system settings 434
- user timeout 436, 437

-tuner command-line option 412  
tuner properties

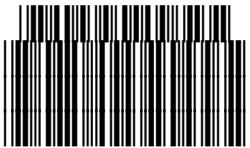
- conflict resolution 291
- conflicts 292, 433
- defined 276
- deleting 288, 289
- format 287
- list of commonly used tuner properties 276

marimba.subscription.adminusers 279, 431  
marimba.subscription.autostart 280  
marimba.subscription.installmode 278  
marimba.subscription.machinename 280  
marimba.subscription.nodelete 277  
marimba.subscription.reapplyconfigonfail 278  
marimba.subscription.retrycount 277  
marimba.subscription.retryintervalsec

280  
marimba.subscription.retrytime 277  
marimba.subscription.timeout 281  
marimba.subscription.useshortcuts  
    277  
marimba.subscription.varytime 281  
marimba.tuner.display.noerrors 281  
marimba.tuner.display.noprogress 281  
    setting for targets 285  
    setting from the command line 388,  
        412  
    setting from the GUI 285  
    using spaces in 288  
-txadminaccess command 108  
-txadminaccess command-line option 399  
**U**  
uid attribute 424  
uninstall state 217  
uninstallation of patch groups 301  
uniqueMember attribute 424  
universal groups, using for Policy Manager 70  
update schedule  
    defined 234  
    format for the command line 422  
    Patch Service 304  
    Policy Service 271  
    setting 238  
updates  
    scheduling 261  
    staging 261, 262  
    stopping and retrying for policies 348  
updating endpoints, overview 24  
UPN (user principal name) format 152  
usecomputername property 127  
usednfromclientonly property 121, 123  
useglobalcatalog property 122  
-password command-line option 392  
-user command-line option 392  
user principal name (UPN) format 152  
user roles  
    in Policy Manager 68  
user timeout, troubleshooting 436, 437  
userclass property 118  
useridattr property 118  
users and groups, using transmitters as the source 105  
usetransmitterusers property 106, 122  
**V**  
varytime property 137  
viewing LDAP query collections 87  
**W**  
-w command-line option 418  
wake up machines 219  
Wake-on-Wan (WoW) 219  
WOW deployments 210  
WoW deployments 219







\*439158\*